DOCUMENTS DE TREBALL
DE LA FACULTAT DE CIÈNCIES ECONÒMIQUES I EMPRESARIAL

Col·lecció d'Economia

DESIGN OF HOMOGENOUS TERRITORIAL UNITS: A METHODOLOGICAL PROPOSAL *

Juan Carlos Duque
Raúl Ramos
Jordi Suriñach

Address for correspondence

Grup d'Anàlisi Quantitativa Regional
Espai of Recerca en Economia
Facultat of Ciències Econòmiques i Empresarials, Universitat of Barcelona
Avda. Diagonal 690 - 08034 Barcelona, Espanya
Tel: 934037043 Fax: 934021821
E-mail: jduque@eco.ub.es, rramos@ub.edu, jsurinach@ub.edu

Received: December 2003

* Authors wish to thank an anonimous referee, E. López-Bazo and E. Pons for helpful comments and suggestions to previous versions of the paper. The usual disclaimer applies. Financial support is gratefully acknowledged from CICYT SEC 2002-00165 project. Juan Carlos Duque also thanks the support of the Generalitat de Catalunya through the grant 2001FI 00296.

ABSTRACT:

One of the main questions to solve when analysing geographically added information consists of the design of territorial units adjusted to the objectives of the study. This is related with the reduction of the effects of the Modifiable Areal Unit Problem (MAUP).

In this paper an optimisation model to solve regionalisation problems is proposed. This model seeks to reduce some disadvantages found in previous works about automated regionalisation tools.

Key words: Zone design, Modifiable Areal Unit Problem, Optimisation, Contiguity constraint.

JEL codes: R22, R12, C61

RESUMEN:

Uno de los principales inconvenientes al realizar estudios que impliquen la utilización de información agregada geográficamente consiste en la sensibilidad de los resultados a la forma cómo se ha configurado las unidades territoriales, unidades que en muchas ocasiones no se relacionan con los objetivos del estudio sino con la disponibilidad de información estadística. Dicho problema ha sido estudiado en la literatura cómo el Problema de la Unidad Espacial Modificable (PUEM).

En este estudio se presenta un modelo de optimización y un algoritmo para el diseño de unidades territoriales con los cuales se busca cubrir algunos vacíos encontrados en otras formulaciones propuestas en la literatura.

Palabras clave: Zonificación, unidad espacial modificable, optimización, restricción de continuidad.

Códigos JEL: R22, R12, C61

DESIGN OF HOMOGENOUS TERRITORIAL UNITS: A METHODOLOGICAL PROPOSAL

1. INTRODUCTION

The interest for geographical information technologies has considerably increased during the last three decades. Nowadays, geographical information is no more exclusive of government and public administrations (in the areas of planning, demography and topography) thanks to the development of computer tools (in software and hardware) that have made possible to use this information in firms and in academic areas.

This kind of statistical information is usually published at different territorial levels with the aim of providing information of interest for all the potential users. When using this information, they have two different choices: first, to use the officially established territorial units (towns, provinces, etc.) or, second, to design territorial units directly related with the analysed phenomena aggregating territorial units of small size[1], but without arriving at the upper level, or combining information from different levels[2].

[1] Apart from aspects such as the statistical secret or other legislation about the treatment of statistical data, according to Wise *et al*, (1997), this kind of territorial units are designed in such a way as to be above minimum population or household thresholds, to reduce the effect of outliers when aggregating data or to reduce possible inexactities in the data, and to simplify information requirements for calculations or to facilitate its visualisation and interpretations in maps.

[2] See, for example, Albert *et al*, (2003), who analyze the spatial distribution of economic activity using information with different levels of regional aggregation, NUTS III for Spain and France and NUTS II for the rest of countries, with the objective "using similar territorial units". López-Bazo *et al*. (1999) analyze inequalities and regional convergence

In most cases, the aggregation of territorial information is usually done using "*ad-hoc*" criteria due to the lack of regionalisation methods with enough flexibility. In fact, most of these methods have been developed to deal with very particular regionalisation problems, so when applied in other contexts the results could be very restrictive or inappropriate for the considered problem. However, and with independence of the applied territorial aggregation method, there is an implicit risk, known in the literature as "Modifiable Areal Unit Problem" (Openshaw, 1984), and related with the sensitivity of the results to the aggregation of geographical data and its consequences on the analysis.

This paper formulates a new methodology of aggregation based on an optimisation model that tries to overcome some of the disadvantages of available methodologies. Among the main characteristics of the proposed model, it is worth mentioning the following:

a) Automated regionalisation model in order to design a given number of homogeneous geographical units from aggregate small areas subject to contiguity requirements.

b) The aggregation process takes into account not only characteristics of each area[3] but also the relationships among them (symmetric and not necessarily metric).

---

at the European level in terms of GDP per capita using a database for 143 regions using NUTS II data for Belgium, Denmark, Germany, Greece, Spain, France, Italy, Netherlands and Portugal, and NUTS-I for the United Kingdom, Ireland and Luxemburg with the objective of ensuring the comparability of geographical units.

[3] In this paper, we will use the term "area" to denote the smallest territorial unit. The aggregation of areas will form a "region" and the aggregation of regions will cover the whole considered territory.

c) In this paper, the regionalisation problem is posed as a lineal optimisation problem. This ensures the possibility of finding, among all feasible solutions, the global optimum.

d) More coherent solutions can be easily obtained introducing additional constraints about other specific requirements relevant for the regionalisation process.

e) There is more freedom than in other methodologies regarding the shapes of the regions, which only depend on data characteristics and are not imposed by the considered methodology.

The paper is organised in the following sections: in section 2 the literature about the different regionalisation methods are briefly summarised; in section 3 the proposed lineal optimisation model for automated regionalisation is described; section 4 introduces an algorithm to deal with more complex regionalisation problems, and, last, the most relevant conclusions of the paper are presented in section 5.

2. REVISION OF THE LITERATURE

In this section, we briefly summarised the most relevant methodologies for territorial aggregation. We have only focused on those methodologies with a higher impact in the specialised literature and on those ones that have been tested satisfactorily in real problems.

Some of these methodologies use techniques based on cluster analysis[4]. In this context, the problem of aggregation of spatial data is considered as a particular case of clustering where geographical contiguity among the elements to be grouped should be considered. This particular case of clustering methods is usually known as contiguity-constrained clustering or simply regionalisation problem. A detailed summary of these aggregation methodologies can be found in Gordon (1999) and for the case of constrained clustering in Fisher (1980), Murtagh (1985) and Gordon (1996).

Regionalisation algorithms can be categorized under three methodological strategies: two-stages aggregation; the inclusion of geographical information in the set of classification variables; and, the use of additional instruments to control for the geographical contiguity constraint.

2.1. Two stages aggregation

This strategy consists of splitting the aggregation process in two stages. The first stage consists of applying a conventional clustering model without take into account the contiguity constraint, and, in a second stage, the clusters

[4] Multivariate statistical tool widely used to classify elements in terms of their similarities or dissimilarities (Jobson, 1991).

are revised in terms of geographical contiguity. With this methodology, if the areas included in the same cluster are geographically disconnected, those areas are defined as different regions (Ohsumi, 1984).

Two conventional clustering algorithms can be used in this context: hierarchical or partitioning.

2.1.1. Hierarchical algorithms.

They are usually applied when the researcher is interested in obtain a hierarchical and nested classification (for every scale levels), that is usually summarised using dendograms[5]. The main disadvantage of using hierarchical clustering algorithms, without considering the high computational requirements (Wise *et al.*, 1997), is the high probability of obtaining local optimum due to the fact that once two elements have been grouped in an aggregation level, they would not return to be evaluated independently in higher aggregation levels (Semple and Green, 1984). On the other hand, the main advantage that should be highlighted is that there is no need to specify initial partitions to apply the algorithm (Macmillan and Pierce, 1994).

2.1.2. Partitioning algorithms.

More used in regionalisation processes is the K-means clustering procedure, which belongs to partitioning clustering category, this iterative technique consists of selecting from elements to be grouped, a predetermined number of *k* elements that will act as centroids (the same number as groups to

[5] Graphical representation of the solutions of hierarchical cluster (Gordon, 1996).

be formed). Then, each of the other elements is assigned to the closest centroid.

The aggregation process is based on minimizing some measure of dissimilarity among elements to aggregate in each cluster. This dissimilarity measure is usually calculated as the squared Euclidean distance from the centroid of the cluster[6], see equation 2.1.

$$\sum_{m \in c} \sum_{i=1}^{N} (X_{im} - \overline{X}_{ic})^2 \qquad (2\cdot1)$$

Where X_{im} denotes the value of variable i (i=1..N) for observation m (m=1..M), and $\overline{X}_{ic}$ is the centroid of the cluster c to which observation m is assigned or the average X_i for all the observations in cluster c.

K-means algorithm is based on an iterative process where initial centroids are explicitly or randomly assigned and the other elements are assigned to the nearest centroid. After this initial assignation, initial centroids are reassigned in order to minimize the squared Euclidean distance. The iterative process is terminated if there is not any change that would improve the actual solution.

It is important to note that the final solutions obtained by applying K-means algorithm depend on the starting point (the initial centroids designation). This fact makes quite difficult to obtain a global optimum solution.

[6] A detailed summary of these aggregation methodologies can be found in Gordon (1999) and for the case of constrained clustering in Fisher (1980), Murtagh (1985) and Gordon (1996).

Finally, when K-means algorithm is applied in a two stages regionalisation process, it will be possible that the required number of regions to design will be not necessarily equal to the value given to parameter *k* as areas belonging to the same cluster have to be counted as different regions if they are not contiguous. So, different proofs have to be done with different values of *k* (lower than the number of desired regions), until contiguous regions are obtained. In some cases could be impossible to obtain the desired number of contiguous regions.

Among the advantages of two stages aggregation methodology, Openshaw and Wymer (1995) highlight that the homogeneity of the defined regions is guaranteed by the first stage. Moreover, this methodology can also be useful as a way to obtain evidence of spatial dependence among the elements. However, taking into account the objectives of the regionalisation process, the fact that the number of groups depends on the degree of spatial dependence[7] and not on the researcher can be an important problem.

2.2. Inclusion of geographical information as classification variables.

The second strategy consists of including as classification variables the geographical coordinates of centroids representing the areas to be grouped (Perruchet, 1983, Webster and Burrough, 1972). In this strategy, as a way to force the geographical contiguity, the geographical coordinates are included in the calculation of dissimilarities between areas and, next, conventional classification algorithms are applied.

---

[7] When the spatial dependence is higher (lower) there will be a trend towards the creation of less (more) regions.

This kind of approach has been implemented in the SAGE system (*Spatial Analysis in a GIS Environment*) (Haining *et al.*, 1996). In its regionalisation algorithm, this system uses an objective function formed by three components, the first controls the intra-group variance taking into account the non spatial attributes, the second, as geographical component, includes the sum of the distances from areal centroids to the cluster centroids in order to force geographical contiguity, and the third component is a deviation measure between the regional value of an attribute and its average value. A different weight is assigned to each of these components in the objective function in order to obtain a unique value to minimise. The regionalisation procedure is based on a partitioning algorithm K-means (Andemberg, 1973).

Calciu (1996) uses the same territorial aggregation strategy, referring to it as "*contrainte spatiale implicite*" (implicit spatial constraint), which incorporates as geographical variables the Cartesian coordinates, conveniently transformed, of the points representing each area. This author is in favour of applying a hierarchical classification algorithm, where the inclusion of the coordinates permits to obtain an improved geographical continuity, although it implies some lost in terms of intragroups homogeneity in relation to the case where the hierarchical algorithm is applied without considering these geographical variables.

The main inconvenient associated to this methodology are the difficulty of treating simultaneously variables expressed in different measure units and the definition of objective weights for each of the variables, specially the geographical ones as the weights should be strong enough to guarantee that geographical contiguous regions are formed (Wise *et al.*, 1997).

Another disadvantage is that the final solution can change depending on the applied method to localise the centroid that represents each of the areas to be grouped, especially in those cases where the areas are considerably big (Horn, 1995, Martin *et al.*, 2001).

2.3. Additional instruments to control for the continuity restriction.

The last, but perhaps the most used strategy to solve territorial aggregation problems, consists of controlling the geographical contiguity constraint using additional instruments as the contact matrix or its corresponding contiguity graph. Contact matrix is a binary matrix with elements c_{ij}, where c_{ij} takes value 1 if areas i and j share a border; and 0 otherwise. In the contiguity graph the areas to be grouped are represented as nodes and arcs represent the adjacency relationship between them[8].

The elements above are used to adapting conventional clustering algorithms, hierarchical or partitioning, with the objective of respecting the continuity constraint.

The main problem with adapted hierarchical algorithms in the context of regionalisation processes is that there can be breaks in monotonicity among elements. This problem is known as reversals: the distance between two objects can be higher than the distance between the union of this object with a third one (Calciu, 1996, Gordon 1996, Ferligoj and Batagelj, 1982). It makes difficult the interpretation of classification.

[8] For a more detailed description of the methods for the elaboration of this kind of graphs, see Gordon (1996, 1999).

In adapted partitioning algorithms, contact matrices or contiguity graphs have mainly been applied into two different methodologies: mathematical programming and iterative algorithms.

Regarding to mathematical programming, Macmillan and Pierce (1994) define the regionalisation problem as an optimisation problem where, given a predetermined number of groups to form, the solution will define the optimum territorial aggregation. The proposed solution by these authors to ensure the geographical continuity consists of exponenciating the contact matrix, taking into account that for the formation of a region with n continuous areas is necessary that the $(n\text{-}1)^{\text{th}}$ power of the contact matrix does not contain null elements. This solution implies that the feasible space defined by the constraints is non-convex and, as a result, the objective function is likely to get trapped in a local optimal solution.

Cutting algorithms for graph partitionig are another way to see the regionalisation problem from a mathematical programming point of view. In these models, the contiguity graph has associated in their arcs a value of dissimilarity between areas, i.e. $G=(V,E)$, with a weight function $w : E \rightarrow N$.

The cutting algorithms looks for a partition of the node set V into k disjoint sets $F=\{C_1, C_2,..., C_k\}$ where k is integer and $k \in [2..|V|]$. Thus, in a regionalization process, the idea could be to maximice the isolation between groups, so the objective in a "maximum k-cut" is to maximice the sum of the weight of the edges between the disjoint sets, i.e.:

$$\sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \sum_{\substack{v_1 \in C_i \\ v_2 \in C_j}} w(\{v_1, v_2\}) \qquad (2\cdot2)$$

Where v_1 and v_2 are the endpoints of an arc[9].

Another method, cited by Neves *et al.* (2001), consists of the reduction of the contiguity graph (G=(V,E)) where each arc has associated a value of dissimilarity between areas (weight function $w : E \rightarrow N$). The reduction makes a progressive elimination of arcs until a minimum spanning tree is obtained. The main point of this representation is that the elimination of one arc at a time implies the partition of the graph in intraconnected, but not interconnected, subgroups (Ahuja *et al.*, 1993).

One disadvantage of the regionalisation methodologies modelling the dissimilarity relationships using the arcs of the contiguity graph is related with the fact that an important number of dissimilarity relationships between areas that are not contiguous are not being considered.

Taking into account that the resolution of this kind of problems using conventional optimisation methods is extremely complex[10], other methodologies have been developed in the field of regionalisation that have been very effective in those cases where the number of elements to group is very high. Among these different solutions, the algorithms known as *Iterative Relocation Algorithms* have been widely analysed. These methods try to find the best regional configuration using as a starting point a non-optimal configuration[11] and, next, different movements of areas between regions are done with the objective of improving the objective function. Ferligoj and

---

[9] A compendium of models related to network design can be found in Crescenzi and Kann (2004).

[10] Openshaw (1984) calculated that to aggregate 1,000 areas in 20 regions there are 101,260 different solutions. For more information about combinatorial problems, see Aarts and Lenstra (1997).

[11] Different alternatives to determine the initial solution can be found in Wise et al. (1997).

Batagelj (1982) provide different iterative reallocation algorithms that allow moving an area to a different region only if contiguity constrains are satisfied.

Algorithms such as the *Automatic Zoning Procedure* (AZP) (Openshaw, 1977), the *Land Allocation Problem* (Benabdallah and Wright, 1992), the *Redistricting Problem* (Macmillan and Pierce 1994) and the *Regional Partitioning Problem* (Horn, 1995) have been used in the literature related with the particular case of splitting a country in administrative areas or electoral districts such that the final regionalisation minimises the effects of the Modifiable Areal Unit Problem (MAUP)[12].

Iterative Relocation Algorithms have been improved using heuristics that permit a better search among the different feasible solutions and to avoid the risk of getting trapped into a local optimum. The most used heuristics in this context are the *Simulated Annealing* (AZP-SA) and the *Tabu Search Algorithm*[13,14] (AZP-TABU), proposed by Openshaw and Rao (1995), and the *Anneal Redistricting Algorithm* proposed by Macmillan and Pierce (1994).

The methodologies of constrained clustering where additional instruments, such as distance or contact matrix, are included, have as a common characteristic that the relationships between the areas to group are

---

[12] Openshaw defined the problem of the Modifiable Areal Unit Problem (MAUP) as a potential source of error that can affect the results of those studies based in geographical aggregated information as these results could vary in function of the configuration of this aggregation. The MAUP is related with two different problems regarding the analysis of spatial data: the problem of scale, related with the desired number of regions, and the problem of aggregation, related with the configuration of small areas inside bigger areas. For more information, see Openshaw (1977), Openshaw and Taylor (1981), and in an econometric context, see Fotheringham and Wong (1991) and Amrhein and Flowerdew (1992).

[13] The *Simulated Annealing* was proposed as an optimisation procedure by Kirkpatrick *et al.* (1983) and first time applied in the *Redistricting Problem* by Browdy (1990).

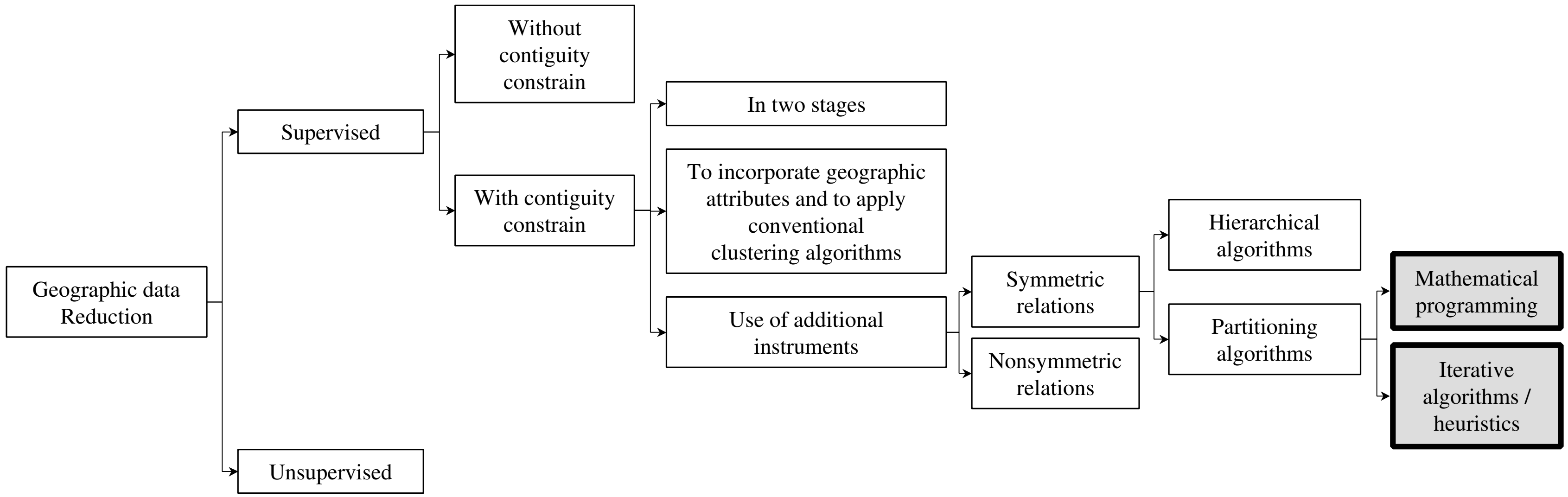
[14] For more information about the *Tabu Search Algorithm*, see Glover (1977, 1989, 1990).

symmetric. In this sense, Ferligoj and Batagelj (1983) have developed agglomerative algorithms where asymmetric relationships can be considered.

All the methods presented above are “supervised” models, which means that the researcher knows *a priori* the data structure of the analysed phenomenon. But there are other unsupervised models that can be useful when the researcher wants to analyse a big amount of data and there is not enough information of the factors that can affect the system. In these cases, one possibility consists in applying a non-parametric analysis of data that will permit to find the patterns and relationships among the considered elements. One of the most known applications of these methods in the field of regionalisation is *Self Organization Maps* (SOM) proposed by Kohonen (1984). There is no consensus among researchers about the validity of this methodology, originally developed in the field of artificial intelligence, due to the lack of a theoretical basis that difficult the interpretation of the results (Openshaw, 1992).

A summary of the different methodologies in this section can be found in table 1.

Table 1. Summary of the different available methodologies for the reduction of geographical data



Source: Own elaboration.

3. A LINEAL OPTIMISATION MODEL FOR THE CONFIGURATION OF REGIONS

3.1. Model description

In this section, we formulate the regionalisation problem as a linear optimisation model that permits the design of regions taking into account the characteristics of the areas but also their relationships. In this model, the restrictions of geographical contiguity will not conditionate, more than necessary, the final result. Before introducing the mathematical formalisation of the model, its main characteristics and assumptions used will be mentioned.

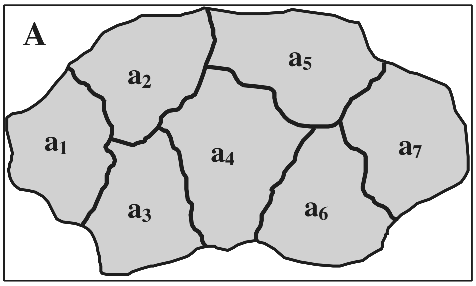
3.1.1. Representation of the geographical set

The starting point of any regionalisation process consists in the identification of the territory to regionalise. As an example, Figure 1 shows a territory that could be regionalised. It is composed by a finite number (*n*) of geographical areas of smaller size that form a geographical contiguous **A** = $\{a_1, a_2, a_3, \ldots , a_n\}$.

Once the territory of interest has been defined, the next step consists in simplifying the previously defined geographical set in a way that each of the considered elements (*n* areas) and their neighbourhood relationships could be easily represented. This simplification can be done using a graph formed by *n* nodes, each of them representing one of the considered areas, and arcs that represent the geographical contiguity among them.

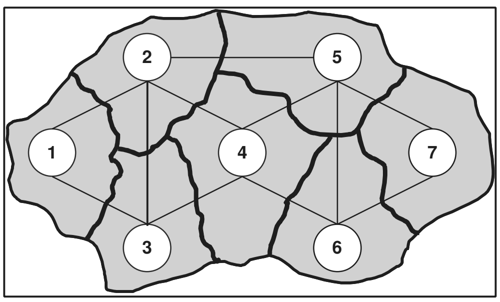
There are different methods in order to make this kind of simplification. We have selected the most general one, the *Delaunay Triangulation* (DT) (Aurenhammer, 1991). With this method, each arc relates those areas with a common border. One of the main advantages of this method is that the localisation of the point representing each of the areas does not affect the result of the graph. Other methods, such as the *Gabriel Graph* (Matula and Sokal, 1980), the *Relative Neighbourhood Graph* (Toussaint, 1980) or the *Minimum Spanning Tree* (Graham and Hell, 1985) are particular cases of DT and results can be different depending on the localisation of the areal centroids. Figure 2 combines the DT graph with the one representing the territory considered in the example.

Figure 1. Group of areas that form the territory to regionalise



Source: Own elaboration.

Figure 2. Delaunay Triangulation (DT)



Source: Own elaboration.

3.1.2. Relationships between the elements to be grouped

The next step consists in the consideration of the relationships between the different areas (or nodes of the graph). The consideration of these relationships is one of the more relevant elements in the regionalisation process proposed in this paper, as its consideration permits to consider interactions between areas in order to obtain more homogeneous regions. For example, if the objective of the study is to build regions with a similar population in order to establish proper comparisons, it will be helpful to consider also information on dissimilarities regarding other socio-economic variables in order to obtain more homogenous regions.

These relationships are incorporated in the model through a squared and symmetric matrix D_{ij} (i = 1, 2, ..., n and j = 1, 2, ... , n) where d_{ij} contains a dissimilarity measure between every couple of areas i, j.

The selected function to calculate dissimilarities between couples of areas should satisfy the following properties:

$$d_{ij} = d_{ji} \qquad \forall i, \forall j = 1,...,n \tag{1}$$

$$d_{ij} \geq 0, \; \left(d_{ij} = 0 \,\text{if}\; i = j\right) \qquad \forall i, \forall j = 1,...,n \tag{2}$$

These properties imply that the function should not be metric (it does not have to satisfy the triangular inequality[15]):

$$d_{ij} \leq d_{ik} + d_{kj} \qquad \forall i, \forall j, \forall k = 1,...,n \tag{3}$$

[15] For more information, see Gower and Legendre (1986).

The possibility of using distance functions that should not be necessarily metric can be understood as a relaxation of the hypothesis used in the regionalisation models based in centroids where the rest of areas are assigned to each region depending on their proximity. When metric distance functions are used, the centroid-based approach ensures that the final solution will satisfy the geographical continuity constrain.

3.1.3. Strategy for the configuration of regions

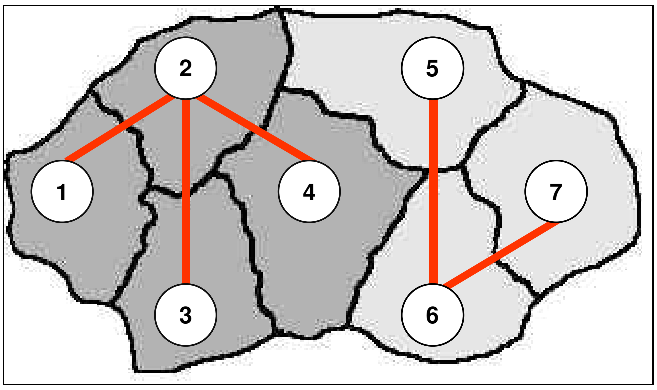
Once we have information about the territorial configuration and the relationships between the different areas, the next step consists in grouping the n areas $\{a_1, a_2, \dots, a_n\}$ in m non-empty sets or regions $\{1, 2, \dots, m\}$ in a way that the areas belonging to each region form a geographical contiguity.

To define these regions it will be necessary to select n-m arcs from the global set of arcs that define the contiguity graph. These n-m arcs can be understood as a necessary but not sufficient condition to form m regions in a way that areas belonging to each region are totally interconnected but disconnected from the areas belonging to other regions. This selection should take into account the following conditions: each region must have a number of arcs equal to the number of areas belonging to the region less one, each region should be formed by a minimum of two areas and, last, in each region, every couple of areas should be connected by a one and only one combination of arcs[16]. This system of regional configurations implies that the minimum number of areas in each region will be two (one arc connecting two areas), or in mathematical terms $m = [n/2]$. This condition is less restrictive as the

[16] For more information about the properties of this (and other) configurations, see Ahuja, Magnanti and Orlin (1993).

number of areas forming the territory increases[17]. Figure 3 shows a possible solution to design 2 regions from 7 areas.

Figure 3. Feasible result for the design of two regions



Source: Own elaboration.

The localisation of arcs in each region does not have influence on the final result. For example, the region formed by the areas connected by arcs 1-2, 2-3 and 2-4 can be also configurated with arcs 1-3, 2-4 and 3-4. This result is related with the fact that the arcs function is only to ensure geographical contiguity, because of they do not have any value assigned. This strategy can be very useful to identify regional configurations with a high variety of shapes (longed or compact regions), as it does not rely on centroids, which tend to produce compact areas.

[17] If we have one area that is considered as an outlier it should be treated as a region, the solution will be to exclude from the analysis and forming m-1 groups with the other n-1 areas.

3.1.4. Considered criteria for the configuration of regions: the objective function

The objective of grouping n areas in m regions is that the areas belonging to each region form a homogeneous geographical contiguity. So, a partition criterion considering which one of the possible configurations of n areas in m regions is the most adequate should be defined.

With this aim, it is necessary to define a measure of adequacy of a regional configuration. One possibility consists in calculating the degree of heterogeneity of the areas assigned to a region or, other alternative could be to calculate the degree of isolation of the areas of one region related to the rest. The heterogeneity measure selected in this paper consists in the sum of the elements of the upper triangular matrix of dissimilarity relationships between the areas in the considered region. Following Gordon (1999), the heterogeneity measure for region r, C_r can be calculated as follows:

$$H(C_r) \equiv \sum_{\{i,j \in C_r | i<j\}} d_{ij} \quad (4)$$

Taking this into account, the problem of obtaining r homogeneous classes (regions) can be understood as the minimisation of the sum of the heterogeneity measures of each class (region) r:

$$P(H,\Sigma) \equiv \sum_{r=1}^{c} H(C_r) \quad (5)$$

or, following the MIN-MAX strategy, we can also try to minimise the value of the most heterogeneous region as this imply that the rest of the regions would be equal or less heterogeneous:

$$P(H, Max) \equiv \max_{\{r=1,\ldots,c\}} H(C_r) \qquad (6)$$

One disadvantage associated to the second strategy is that once the value of the most heterogeneous region is minimised, the configuration of the rest of the regions will not be revised, avoiding the possibility of making changes that could improve their heterogeneity. For this reason, the strategy followed in this paper consists in the minimisation of the sum of the heterogeneity measures of each region $(P(H,\Sigma))$.

It is worth mentioning that both objectives, minimising internal heterogeneity $H(C_r)$ and maximising the isolation among regions $I(C_r)$, are not independent. In fact, we can formulate an equivalent objective in terms of isolation criteria:

$$P(H,\Sigma) \equiv P(I,\Sigma) \equiv \sum_{r=1}^{c} I(C_r) \text{ with } I(C_r) \equiv \sum_{i \in C_r} \sum_{j \notin C_r} d_{ij} \qquad (7)$$

3.2. Mathematical model

Parameters:

i, I index and set of areas, $i = \{1,...,n\}$;

k,K index and set of regions, $k = \{1,...,m\}$;

c_{ij} $\begin{cases} 1, \text{if } i \text{ and } j \text{ are continuous (share a border), with } i < j, \\ 0, \text{otherwise;} \end{cases}$

M $Max\left(\sum_{j=1}^{n} c_{1j},...,\sum_{j=1}^{n} c_{nj}\right)$;

N_i $\{j | c_{i,j} = 1\}$;

$D_{i,j}$ Dissimilarity relationships between areas i and j, with $i<j$;

Decision Variables:

X_{ijk} $\begin{cases} 1, \text{if areas } i \text{ and } j | j \in N_i \text{ belong to the same region } k, \text{with } i < j, \\ 0, \text{otherwise;} \end{cases}$

Y_{ik} $\begin{cases} 1, \text{if area } i \text{ belongs to region } k, \\ 0, \text{otherwise;} \end{cases}$

T_{ij} $\begin{cases} 1, \text{the disimilarity relationship between } i \text{ and } j \text{ is considered if the two areas} \\ \quad \text{belong to the same region } k, \; i < j, \\ 0, \text{otherwise;} \end{cases}$

Objective function: $Min \sum_{i=1}^{n} \sum_{j=1}^{n} D_{ij} \cdot T_{ij}$

Subject to:

$$T_{ij} \geq Y_{ik} + Y_{jk} - 1, \qquad \forall i, \forall j = 1,...,n \; ; \; \forall k = 1,...,m \qquad (8)$$

$$\sum_{i=1}^{n} Y_{ik} \geq 2, \qquad \forall k = 1,...,m \qquad (9)$$

$$\sum_{k=1}^{m} Y_{ik} = 1, \qquad \forall i = 1,...,n \tag{10}$$

$$\sum_{j \in N_i} X_{ijk} \leq Y_{ik} \cdot M, \qquad \forall i = 1,...,n \; ; \; \forall k = 1,...,m \tag{11}$$

$$\sum_{j \in N_i} X_{jik} \leq Y_{ik} \cdot M, \qquad \forall i = 1,...,n \; ; \; \forall k = 1,...,m \tag{12}$$

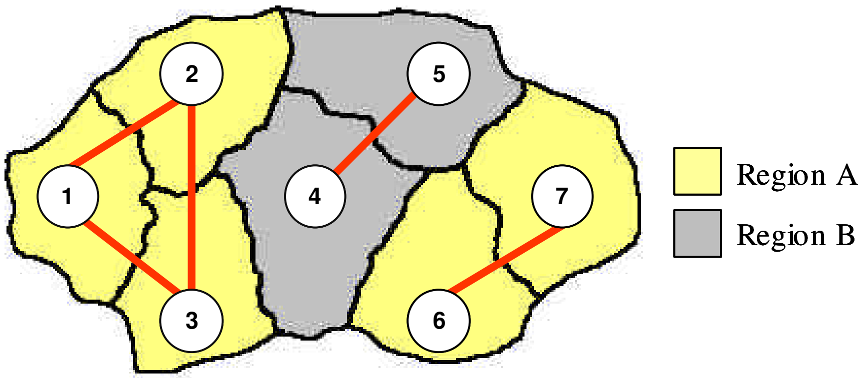
$$\sum_{i=1}^{n} \sum_{j \in N_i} X_{ijk} = \sum_{i=1}^{n} Y_{ik} - 1, \qquad \forall k = 1,...,m \tag{13}$$

$$\begin{aligned} &\sum_{i,j \in C} X_{ijk} \leq |C| - 1, \; \forall \text{ non - empty subset of } C \subseteq \{3,...,(n\text{-}2m+1)\}; \\ &\qquad\qquad \forall k = 1,...,m \end{aligned} \tag{14}$$

$$X_{ijk} \in \{1,0\} \, ; \, Y_{ik} \in \{1,0\}; \, T_{ij} \geq 0, \quad \forall i, \forall j = 1,...,n \; ; \; \forall k = 1,...,m \tag{15}$$

As it was previously mentioned, the objective function of the model is related with the minimisation of the total heterogeneity, measured as the sum of the elements of the upper triangular matrix (D_{ij}) of dissimilarity relationships between areas belonging to the same region (the elements defined by the binary matrix T_{ij}). Restriction (8) controls the assignation of the values of matrix T_{ij} where, by the nature of the objective function, the relationship between areas i and j will only be taken into account if they belong to the same region. Restriction (9) imposes that the minimum number of areas defining a region is two. As it was previously mentioned, the restriction is less strong as the number of areas increases. Restriction (10) imposes that each area must be assigned to one and only one region. Restrictions (11) and (12) imposes that only when the area i is assigned to region k, it will be possible to establish arcs to the neighbourhoods of the area ($j \in N_i$). To avoid an excessive reduction of feasible regional configurations, the number of arcs from an area can be higher than one. Restriction (13) imposes that the number of arcs to ensure geographical contiguity of the areas assigned to one region must be equal to the number of areas in the region less one. However, this restriction does not totally ensure that the final solution will be formed by contiguous regions. There are cases such as the one shown in figure 4, where region *A*, formed by areas 1, 2, 3, 6 and 7, satisfies restriction (13) –there are four connecting arcs for five areas- but the combination of arcs 1-2, 1-3, 2-3 generates a cycle that breaks the geographical contiguity of the region. For this reason, it will be necessary to control, a part of the number of arcs, if there are cycles and this is the origin of restriction (14).

Figure 4. Non-feasible regional configuration



Source: Own elaboration.

The problem of cycles has been treated in the literature as the analysis of *subtour* in transport models such as the *Vehicle Routing Problem* (VRP)[18]. The VRP consists in defining vehicles routes with a given origin and end in the same node (called *depot*) and trying to minimize costs. The design of a

[18] This problem was first proposed by Dantzing and Ramser (1959). A survey about the models derived from this approach can be found in Laport and Osman (1995).

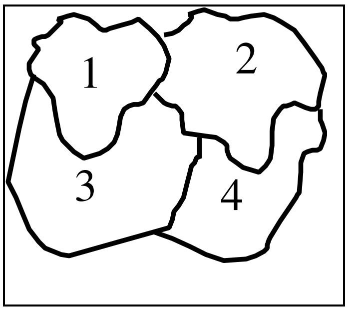
tour for a certain vehicle cannot contain *subtours* and to control this condition, the VRP incorporates the following constraint:

$$\sum_{j,i \in S} X_{ijk} \leq |S| - 1, \; \forall \text{ non-empty subset of } S \subseteq \{2,...,n\};\; k=1,...,m. \qquad (16)$$

The main disadvantage of this approach is that the number of restrictions increases exponentially with *n and m*. For this reason, and although the proposal is theoretically adequate, at the practical level it has been necessary to implement other restrictions to solve this problem in a more efficient way. These alternatives can be appropriated for the specific problem of the VRP (although they do not ensure the elimination of *subtours* in problems of a certain dimension), but not for the regionalisation problem. For example, it is required to establish *a priori* a *depot* node that will be the origin and end of all the *tours*, and it is also necessary to establish a sequential order among nodes.

However, the theoretical restriction of the VRP can be adapted in an efficient way in this geographical context as we know the number of elements of the set *S*. For example, in the territorial configuration of figure 5 we can clearly identify the different combination of arcs $c_{i,j,k}$ that can generate cycles. The combination of arcs 1-2, 1-3, 2-3 (o 2-3, 2-4, 3-4) will produce a cycle where 3 areas would be involved, 1, 2 and 3 (or 2, 3, 4), while the combination of arcs 1-2, 1-3, 3-4, 2-4 will generate a cycle among the four areas.

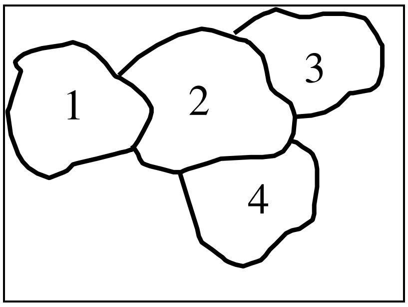
Figure 5. Configuration of areas with potential cycles



Source: Own elaboration.

Moreover, in a territorial configuration as the one shown in figure 6, there is no combination of arcs $c_{i,j,k}$ that could generate a cycle. For this reason, at the territorial level, not every subset *S* can have cycles as the number of potential arcs $c_{i,j,k}$ is limited to those combinations *i*,*j* where the value of the contact matrix *wij* =1. This is the set of potential arcs $c_{i,j,k}$ that are included in $\boldsymbol{N_i}$.

Figure 6. Configuration of areas without potential cycles



Source: Own elaboration.

But, is there any special pattern that could help to detect potential cycles in a specific territorial configuration? Yes, we only have to identify those combinations of arcs where the number or arcs is equal to the number of areas connected through them. For example, in the case shown in figure 5, the three arcs 1-2, 1-3, 2-3 (or 2-3, 2-4, 3-4) connect three areas, 1,2,3 (or 2,3,4), and as a result, 3 arcs and 3 areas imply the existence of a cycle. The same happens with the combination of arcs 1-2, 1-3, 3-4, 2-4 that connect four areas (1,2,3,4). Again, 4 arcs and 4 areas imply the existence of a cycle of 4 elements.

But, for a territorial configuration of n areas that will be grouped in m regions, which is the maximum number of areas that can be involved in a cycle? As the model, in restriction (9), requires that the minimum number of areas in a region is 2, in the case where (m-1) regions are formed by two areas, there will be no possibility of cycles, as each region will only have one possible arc (restriction 13). For this reason, when creating m-1 regions with 2 areas, we will have a region formed by n-2(m-1) areas with (n-2(m-1))-1 arc, which is the maximum number of arcs that can create a cycle. Simplifying this expression, we have that:

$$n-2m+1 \tag{17}$$

So, the minimum number of areas where the possibility of finding a cycle should be evaluated is three, as it is impossible that for a lower number of areas we find this problem.

As a result, restriction (14) is related with the modification of the set S as proposed in the VRP. Using this modification, we achieve an important reduction in the number of restrictions to satisfy, avoiding that the number of restrictions increases exponentially with n and m. This fact permits to use commercial software in the context of regionalisation problems with a high number of areas and regions.

Last, restriction (15) only implies that X_{ijk} and Y_{ik} should be binary variables. Although the variable T_{ij} has been defined as positive, and not as binary, it will always take values 0 or 1 because of the combination of restriction (8) with the objective of minimisation of the model[19].

3.3. Application of the model

In this subsection, different examples are shown with the aim of illustrating the model capacity to design regional configurations with different characteristics. With this aim, we have implemented a first set of four examples each one with a different dissimilarity matrixes ($D_{i,j}$), where values $d_{i,j}$ have been established in such a way that it is possible to know *a priori* the optimum regional configurations. The procedure to obtain the dissimilarity matrix in each example has been the following:

1. We have grouped the n areas in m contiguous regions, assigning each area $i = \{1,...,n\}$ to a Region $k = \{1,...,m\}$. This aggregation permits to built the set $R_k \{i | i \in k\}$.

2. We have assigned a value to each of the areas $i = \{1,...,n\}$ depending on the region they have been assigned. This value is given by the sum of a constant with a random term, generated from a uniform distribution among 0 and 1. The value of the constant is different for each region, as there should be a big enough difference (D) in order to obtain significant

[19] The possibility of defining a variable taking values 0 or 1 as positive and not as a binary variable has an advantage when using the branch and bound algorithm, as the number of

different average values for each region. The applied expression has been the following:

$$A_{i \in R_k} = C + (D * k) + \varepsilon \quad \forall i = 1,...,n; \forall k = 1,...,m; \varepsilon \sim U[0,1] \quad (18)$$

3. Next, we have calculated the relationships between areas using a distance function. In particular, the function that we have used is the weighted Euclidean distance calculated among the elements of the A_i vector after centering it.

$$d_{ij} = \sqrt{\left(\frac{A_i^c}{S} - \frac{A_j^c}{S} \right)^2}, \qquad \forall i, j = 1,...,n \mid i < j \qquad (19)$$

where S is the standard deviation of the A_i, vector and A_i^c is a centered vector calculated as follows from A_i:

$$A_i^c = A_i - \left(\sum_{i=1}^{n} A_i \Big/ n \right), \qquad \forall i = 1,...,n \qquad (20)$$

The matrixes obtained with this procedure are shown in Table 2.

---

sub-problems is drastically reduced. For more information about this algorithm, see Hiriart, Oettli and Store (1983).

Table 2. Relationships matrixes for examples 1 to 4

Example 1

area	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	1.04	1.21	1.18	1.11	0.17	0.14	2.26	2.31	0.09	2.31
2		0.17	0.14	0.07	1.22	1.18	1.22	1.27	1.14	1.27
3			0.03	0.10	1.38	1.35	1.05	1.10	1.31	1.10
4				0.07	1.35	1.32	1.08	1.13	1.27	1.13
5					1.29	1.25	1.15	1.20	1.21	1.20
6						0.03	2.43	2.48	0.08	2.49
7							2.40	2.45	0.05	2.45
8								0.05	2.36	0.05
9									2.41	0.00
10										2.41

Example 2

area	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	0.06	0.02	0.03	2.42	2.49	1.23	0.03	1.19	0.04	0.02
2		0.07	0.03	2.37	2.44	1.18	0.09	1.13	0.01	0.04
3			0.04	2.44	2.51	1.25	0.01	1.20	0.06	0.03
4				2.40	2.47	1.21	0.06	1.16	0.02	0.01
5					0.07	1.19	2.45	1.23	2.38	2.40
6						1.26	2.52	1.31	2.45	2.48
7							1.27	0.05	1.19	1.22
8								1.22	0.07	0.05
9									1.14	1.17
10										0.02

Example 3

area	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	0.64	0.80	1.36	1.27	2.03	1.98	0.08	1.98	2.78	2.79
2		0.15	0.72	0.62	1.39	1.34	0.73	1.34	2.13	2.14
3			0.57	0.47	1.23	1.19	0.88	1.18	1.98	1.99
4				0.10	0.67	0.62	1.45	0.62	1.41	1.42
5					0.76	0.72	1.35	0.71	1.51	1.52
6						0.05	2.11	0.05	0.75	0.76
7							2.07	0.00	0.79	0.80
8								2.06	2.86	2.87
9									0.79	0.80
10										0.01

Example 4

area	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**
1	0.23	0.27	0.16	2.45	2.56	0.22	0.04	0.06	0.17	0.04
2		0.05	0.06	2.23	2.34	0.00	0.27	0.28	0.40	0.27
3			0.11	2.18	2.29	0.05	0.31	0.33	0.45	0.31
4				2.29	2.40	0.06	0.21	0.22	0.34	0.21
5					0.11	2.23	2.49	2.51	2.63	2.50
6						2.34	2.61	2.62	2.74	2.61
7							0.26	0.28	0.40	0.26
8								0.02	0.13	0.00
9									0.11	0.02
10										0.13

Source: Own elaboration.

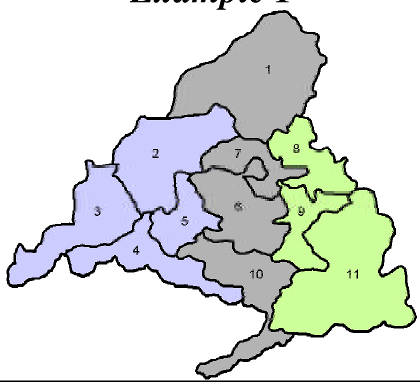
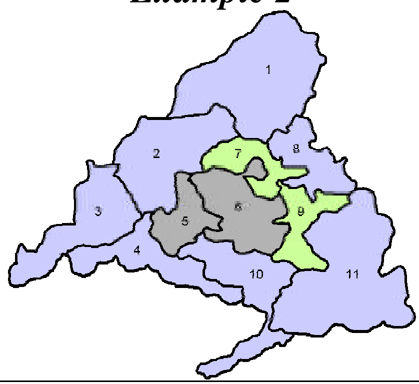
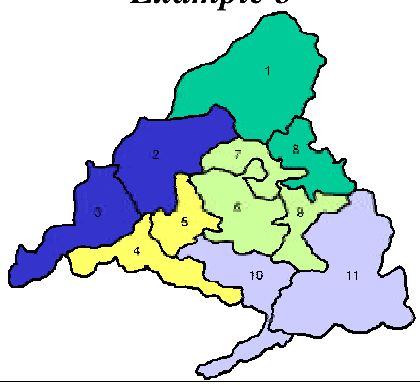
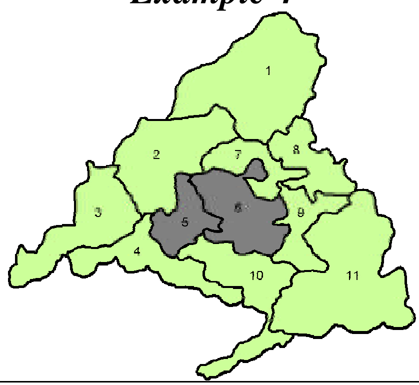
The obtained regional configurations after applying the optimisation model with the different relationships matrix are shown in the maps in table 3. The solutions coincide with the optimal regional configurations predefined above and, so, it seems that the model can design regions with a high variety of shapes

3.4. Additional restrictions that can be incorporated to the model

In this sub-section a second block of examples are shown in order to introduce some restrictions in the model that are usually considered in regionalisation processes.

Using a similar procedure to the one explained in the previous sub-section, we have calculated a relationships matrix ($D_{i,j}$) using demographic data[20] for the 11 statistical areas in which the Comunidad de Madrid is divided at the NUTS IV level (see table 4).

Table 3. Solutions for the relationships matrixes from Table 2

<table>
<tr><td>Example 1

</td><td>Example 2

</td></tr>
<tr><td>n=11 and m=3</td><td>n=11 and m=3</td></tr>
<tr><td>Example 3

</td><td>Example 4

</td></tr>
<tr><td>n=11 and m=5</td><td>n=11 and m=2</td></tr>
</table>

n: number of areas, *m*: number of regions.
Source: Own elaboration.

[20] Replacing ratio: (Population between 15 and 39 years old)/(Population between 40 and 64 years old). Dependence ratio: (Younger than 15 years old and older than 64 years old)/(Population between 15 and 64 years old). Progressivity ratio: (Population between 0 and 4 years old)/(Population between 5 and 9 years old)x100.

Table 4. Demographic variables at the NUTS IV level zones of the Madrid Autonomous Community

AREA	REPLACING RATIO	DEPENDENCE RATIO	PROGRESIVITY RATIO	POPULATION
1	1.333494	0.544033	95.220244	22,407
2	1.491906	0.430047	95.915703	86,954
3	1.343378	0.577842	89.237288	21,719
4	1.564950	0.440989	90.867430	48,655
5	1.440734	0.369530	97.272824	292,155
6	1.263530	0.464020	100.935145	2,879,052
7	1.502627	0.355461	95.658407	233,035
8	1.706222	0.435573	96.254891	25,602
9	1.511078	0.342928	87.525416	452,188
10	1.445924	0.316330	88.654766	1,024,513
11	1.463349	0.529148	86.576424	59,045
	1.35	**0.41**	**95.68**	**5,145,325**

Source: Padron continuo 1999. Instituto de Estadística de la Comunidad de Madrid.[21]

To combine the information of the tree variables (replacing ratio, dependence ratio and progressivity ratio) (v = 1, 2, 3) in the relationships matrix, we used the following distance function:

$$d_{ij} = \sqrt{\sum_{v=1}^{p} \left(\frac{A_{iv}^{c}}{S_v} - \frac{A_{jv}^{c}}{S_v} \right)^2}, \quad \forall i, \forall j = 1,\ldots,n \,\middle|\, i < j \; ; \forall v = 1,\ldots, p \qquad (21)$$

Expression (21) is a multivariate version of (19) that permits the incorporation of p variables, thanks to the inclusion of subindex v. Using this expression, the distance between areas i and j is the square root of the sum of the squared distances between i and j calculated for each of p considered variables. The obtained relationship matrix is shown in table 5.

[21] http://www.madrid.org/iestadis/pc99_d99.htm.

Table 5. Relationships matrix from demographic variables in table 4

area	2	3	4	5	6	7	8	9	10	11
1	1.86	1.33	2.44	2.24	1.63	2.59	3.34	3.20	3.12	2.14
2		2.54	1.24	0.87	2.21	0.87	1.78	2.05	2.06	2.30
3			2.44	3.06	2.89	3.19	3.73	3.07	3.14	1.27
4				1.90	3.30	1.51	1.64	1.41	1.81	1.60
5					1.99	0.64	2.34	2.18	1.93	2.93
6						2.60	3.81	3.78	3.46	3.55
7							1.93	1.74	1.63	2.80
8								2.68	3.02	3.07
9									0.67	2.19
10										2.50

Source: Own elaboration.

3.4.1. Requirement of a population minimum

In order to guarantee that each of the designed regions has a population minimum, it is necessary to introduce the following restriction in the mathematical model:

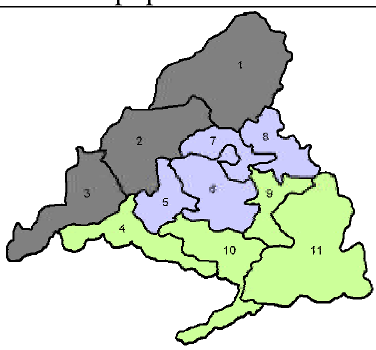
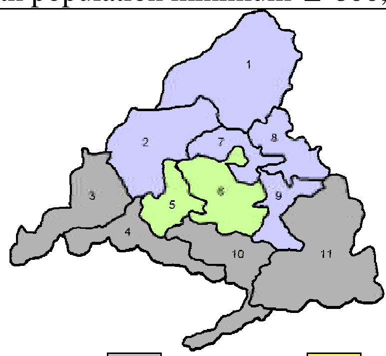
$$\sum_{i=1}^{n} Y_{ik} \cdot P_i \geq L, \quad \forall k = 1,...,m \tag{22}$$

where P_i is a vector containing information of the population of each of the considered areas and L is a constant that specifies the population minimum required. For this example, we have fixed this minimum in 800,000 inhabitants.

Following the suggestions by Openshaw *et al.* (1998), the objective of reducing the population differences among regions has been formulated as an inequality restriction. Using this formulation, it is clear than if the value of L is very high, the problem can be not feasible. However, this kind of situations can be avoided using a multi-objective function with the objective of minimising the regional heterogeneity but also the differences in terms of population. The problem with this approach is to assign weights for these objectives.

The obtained results after solving a model with and without a minimum population requirement are shown in table 6. In the solution for the unrestricted model (left map), the region 1 has the lower value for population, 131,080 inhabitants, while in the solution for the restricted model (right map), the same region is still the one with the minimum population but its value is 820,186 inhabitants (>800,000 inhabitants).

Table 6. Solutions with and without requirements of a population minimum

Without population minimum			With population minimum ≥ 800,000		
					
Region 1	***Region 2***	***Region 3***	***Region 1***	***Region 2***	***Region 3***
Population $_1$	***Population*** $_2$	***Population*** $_3$	***Population*** $_1$	***Population*** $_2$	***Population*** $_3$
131,080	1,584,401	3,429,844	820,186	1,153,932	3,171,207

Source: Own elaboration.

3.4.2. Configuration of regions with mandatory isolation

A different kind of restriction that could be of interest consists in imposing that certain areas belong to different regions in the final solution (mandatory isolation). In this case, the following restriction should be added to the model:

$$\sum_{i=1}^{n} Y_{ik} \cdot O_i = 1, \quad \forall k = 1,...,m \qquad (23)$$

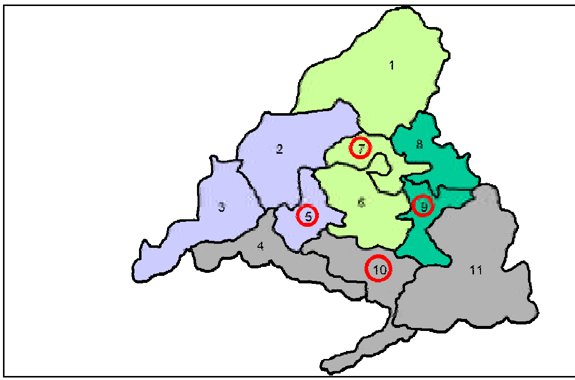
where O_i is a binary vector that takes 1 for selected areas and 0 for the rest. It is important to take into account that, when defining O_i, the number of selected areas must be equal to the desired number of regions:

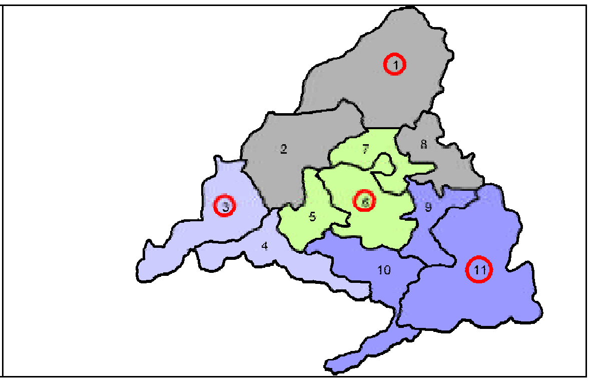
$$\sum_{i=1}^{n} O_i = m \qquad (24)$$

In marketing research, this restriction would be useful when it is necessary to divide a territory in zones in such a way each zone must be assigned to pre-located warehouses.

Table 7 shows the results of applying the model with data from table 5, but imposing that some preselected areas, marked with a red circle, must belong to different regions in the final solution. The results show that the model achieves both objectives: the areas are assigned to different regions and each region is homogeneous in terms of demographic variables.

Table 7. Solutions with mandatory isolation





Source: Own elaboration.

3.5. Computational results

One of the most interesting features of optimisation models when applied in real problems is the required computational time to achieve the optimal solution.

With the aim of testing the computational capacity of the model, it was applied to different random territorial configurations. The procedure to obtain these random configurations was the following:

a. For a given number *n* of areas, a triangular matrix was randomly generated following a [0,1] uniform distribution.

b. A threshold point, between 0 and 1, was fixed in a way that random numbers above this point were replaced by 1, and 0 otherwise. The obtained binary matrix can be interpreted as a contact matrix, which should be evaluated in terms of contiguity.

   The threshold value was assigned taking into account that the resulting territorial configuration (or connecting arcs) was realistic in term of the neighbourhoods of each area. The selected matrixes have an average density of 28.3% and a median of neighbourhoods of 3 per area, ranging from 1 to 8.

c. Every randomly generated matrix was evaluated in terms of geographical contiguity and we only selected feasible ones[22].

d. Last, the relationships between the *n* considered areas were randomly generated from a [0,1] uniform distribution. Using this method, we are assuming a scenario where relationships between areas are not geographically dependent.

Table 8 shows the average running times[23] for different combinations of areas and regions (5 examples for each combination).

Table 8. Average running time, in seconds, for different combinations (areas-regions)

		Regions		
		2	**4**	**6**
Areas	**5**	<1*	-	-
	8	<1*	3.00	-
	11	<1*	19.00	-
	14	5.80	117.40	2,571.00
	17	2.20	2,458.20	42,283.80

Note: Five examples for each combination of areas and regions.
* Execution times lower than a second.
Source: Own elaboration.

[22] Although the decision of evaluating *a posteriori* the contiguity of the matrix would imply a higher computation time for the generation of the different examples, this methodology assures that the territorial configurations in each example are totally random.

[23] The calculations in this paper have been performed using Extended LINGO/PC 6.0 in a PC computer with a Pentium 4 processor at 2.40C GHz and 256 Mb of RAM memory.

Although the number of restrictions was clearly reduced with the modification of constraint (14), that controls the elimination of cycles, the running time stills very high. In fact, for those cases with more than 17 areas the time increases substantially. For this reason, other alternatives that would permit to increase the computational capacity of the model would be considered in the next section

4. *RASS* (*Regionalisation Algorithm with Selective Search*)

The characteristics of a regionalisation process can generate certain inefficiencies in the heuristics that have been adapted for this context (a summary of the different heuristic is presented in Annex 1). In fact, one of the aspects that have been less considered in the literature is the capacity of heuristics such as the *Tabu Search* or the *Simulated Annealing* when used in regionalisation process.

In this section, we propose a new algorithm, called *RASS* (*Regionalisation Algorithm with Selective Search)*, as a regionalisation tool that solves some of the inconvenients associated to the previously mentioned methodologies. The most relevant characteristic of this algorithm is related with the fact that the way it operates is inspired in the own characteristics of regionalisation processes, where available information about the relationships between areas can play a crucial role in directing the searching process in a more selective and efficient way (less random).

The *RASS* incorporates inside its algorithm the optimisation model presented in sub-section 3.2. in order to achieve local improvements in the objective function. These improvements can generate significant changes in regional configurations, changes that would be very difficult to obtain using other iterative methods.

4.1. Steps for the application of *RASS*

Step 1: Take as a starting point, a feasible solution of m regions that group n areas.

Step 2: Select from these m regions the more heterogeneous geographical contiguity formed by r regions with $(m-1) \geq r \geq 2$.

$$H(C_m) \equiv \sum_{\{i,j \in C_m | i<j\}} d_{ij} \rightarrow Max\left(\sum_{m \in M_i} H(C_m) \right) \qquad (25)$$

where M_i is the set formed by the different alternatives of selection of r contiguous regions of the available m regions.

Step 3: Application of the direct optimisation model to the areas of the r selected regions to create r^* regions.

Step 4: Select a region to include (e): From the (m-r) regions that were not considered, identify those areas bordering on territory formed by the r^* regions and select the one with higher similarities with any of the regions in r.

$$I(C_{d,f}) \equiv prom\left(\sum_{i \in C_f} \sum_{j \in C_d | j>i} d_{ij}\right) \rightarrow Min\left(I(C_{d,f})\right) \qquad (26)$$

where d is the set of the r^* regions which are inside, and f is a subset of regions bordering on d. Each of the (m-r) regions that were not selected in the step 2 will only be selected once in every cycle (steps 2 to 8).

Step 5: Select the region that will be removed (s): The region with higher differences with the region to be included (e) in step 4 will be removed from d. The region to be removed cannot destroy the internal contiguity of d.

$$I(C_{d,e}) \equiv prom\left(\sum_{i \in C_e} \sum_{j \in C_d | j>i} d_{ij}\right) \rightarrow Max\left(I(C_{d,e})\right) \qquad (27)$$

Step 6: Include in the set of r regions the region (e) and remove (s): d=(d+e-s). The direct optimisation model will be applied to the new configuration of r regions to create r^* regions.

Step 7: Repeat steps 4 to 6 until the (m-r) regions that where not selected in step 2 have been included at any time in d, or until there are no more candidates to be selected in the bordering on d.

Step 8: Calculate the value of the objective function.

Step 9: If the value of the objective function improves, step 2 would be repeated. If the value of the objective function does not improve, step 2 would be repeated but selecting the next more heterogeneous group. Steps 2 to 8 would be repeated until no significant improvement in the objective function is found in a given number of cycles (*C*) or until the list of alternative *r* contiguous regions is exhausted.

Some characteristics to highlight from the *RASS* algorithm are the following:

a) The application of direct optimisation to a group of regions, in steps 3 to 5, permits to achieve improvements in the objective function that can be accompanied by important changes in regional configurations because of the reassignation of an important number of areas.

b) The criteria used in step 2 for the selection of *r* regions and the criteria for including/removing regions in steps 4 and 5 try to keep in the optimisation model, step 3, those regions with a higher potential to improve the objective function after reconfiguration.

   The objective is to ensure that the included region is the one that presents the higher probability of containing areas belonging to other regions. This potential reassignation is identified assuming that two regions with exchanged areas, decreases the dissimilarities among these regions.

   Last, when the region to be included (*e*) is selected, the next step establishes that the region to be removed (*s*) (in order to keep an appropriated number of areas for the optimisation model) is the more different one from the region to include. This region has lower possibilities of exchanging areas with the region to be included (*e*).

c) The conditions in steps 7 and 9 try to avoid repetitive searching patterns. Moreover, the criteria for including/removing regions and the use of the optimisation model clearly improve the capacity of RASS of escaping from local optimum.

d) The fact of applying the optimisation model only to a part of the considered territory does not imply that each local improvement could worsen the global solution. In fact, after each cycle, the value of the objective function will be always lower or equal to the value of the objective function at the beginning of the cycle.

4.2. Computational results and comparison with the direct optimisation

This sub-section tries to evaluate the performance of the *RASS* algorithm respect the direct optimisation model. The solved examples are the ones that were randomly generated in sub-section 3.5[24]. In order to apply the algorithm to these examples, it was necessary to define an initial feasible partition that could be used as a starting point for *RASS*. The initial partition was randomly generated following these steps:

a) Generate a vector with n values (as many as areas) using a uniform distribution between 0 and 1.

[24] In this analysis we have excluded the examples where 2 regions should be formed, as in this case the application of the *RASS* would be equivalent to the direct application of the optimisation model: there is no difference between the values of parameters m and r of *RASS* and, as a result, the application of step 3 will take directly to the optimal solution.

b) The interval [0,1] is divided in equal sized intervals, as many as the number of regions to design. For example: for 2 regions we used the intervals [0, 0.5) and [0.5, 1] and for 4 regions, the intervals were [0, 0.25), [0.25, 0.5), [0.5, 0.75) and [0.75, 1). Each of these intervals represents a region, in such a way that the elements of the random vectors can be transformed in a vector that assignates areas to regions (potential initial partition).

c) If the initial partition was feasible in terms of geographical contiguity, this partition was used as starting point for *RASS*. If this were not the case, we went back to step a).

Some descriptives of the results for the 30 considered problems (5 for each combination of regions and areas) are shown in table 9. *RASS* achieved the optimal solution in the 100% of the considered examples in a considerably lower time than the direct solution method.

Table 9. Comparison of *RASS* with the direct solution method

Regions	Areas	Optimum/5	Seconds (*RASS*)	Seconds (Direct)	$\frac{(FOI - FO1c)}{(FOI - SO^*)}$
4	8	5/5	3.40	3.00	76.45%
	11	5/5	5.80	19.00	86.70%
	14	5/5	29.00	117.40	74.31%
	17	5/5	247.20	2,458.20	69.46%
6	14	5/5	25.20	25,710.00	85.93%
	17	5/5	250.00	42,283.80	66.71%

FOI= Initial objective function, FO1c= Objective function after the first cycle, SO= Optimal solution*

Source: Own elaboration.

In the last column, we can also see that after the first cycle of the *RASS*, the value of the objective function is reduced in an 80% of the total reduction required to achieve the global optimum.

Using the available information about running times of both methods, the direct method and the *RASS*, it is possible to calculate the time savings by applying the algorithm. Figure 7 shows the relationship between the savings and an indicator of complexity that has been defined as the product between the number of considered areas and the number of considered regions. The results in this figure show that in less complex models the direct method is a better option, while in complex models the RASS provides better results. According to these results, this change happens for models with a complexity over 57.83 (58 if we keep the discrete nature of the variable[25]).

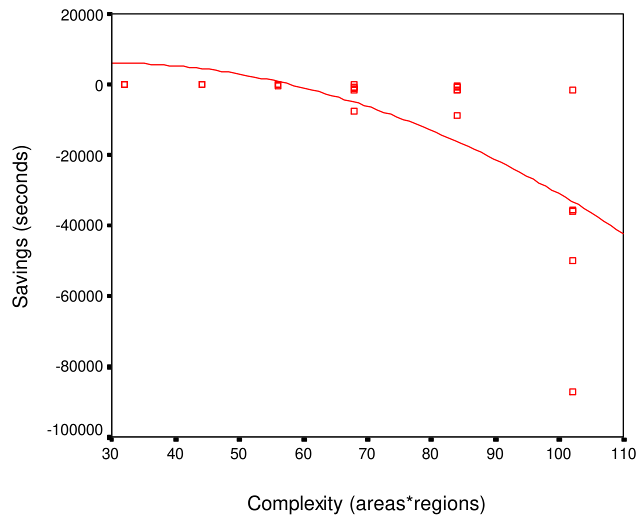
In order to obtain a better measure of the time savings achieved with *RASS*, we have estimated a quadratic model between time savings and the measure of complexity[26,27]. The results of estimating this model are shown in table 10. There is a significant relationship between the two variables at 1% significance level. In front of a marginal increase in the complexity of the problem, the use of RASS implies a time saving of 426.08-14.73 (*areas*regions*), a result that confirms the previously mentioned intuition.

[25] It should be highlighted that this value can be obtained with different combinations of areas and regions.

[26] We have considered together the effects of the number of areas and regions because when introduced separately in the regression, there is a problem of collinearity due to the high correlation among them.

[27] We have excluded the intercept from this regression in order to impose that the execution time is equal to zero when the complexity is equal to zero.

Figure 7. Relationship between the complexity of the problem and the time savings obtained after applying *RASS*



Source: Own elaboration.

Table 10. Quadratic regression among the time savings obtained with *RASS* and the complexity indicator

n=30	Coefficient
(areas×regions)	426.078*
$(\text{areas}\times\text{regions})^2$	-7.367*
R^2	*0.566*
F	*18.269**

** Significant at 1%*

4.3. Capacity of the *RASS* to achieve global optimums in more complex problems

As in more complex problems, it is impossible to compare the results obtained by the RASS and direct optimisation because the execution method for the second would be very high, in this section we present the obtained solution for a regionalisation process where we want to group 38 areas in 10 regions (complexity of 38*10 = 380). For this comparison, we have followed the same procedure than in the examples of sub-section 3.3: A relationship matrix D_{ij} is defined in a way that it is possible to know *a priori* the optimal solution of the regionalisation process. This optimal solution can be compared with the solution obtained by the *RASS* .

4.3.1. Data

a) Characteristics of the territory to regionalise

The selected areas for this example are the 38 areas (*Zones Estadístiques Grans*) that form the city of Barcelona. The first step consists in considering the contiguity relationships among these 38 areas or, in other words, in obtaining the contact matrix.

b) Relationships among areas

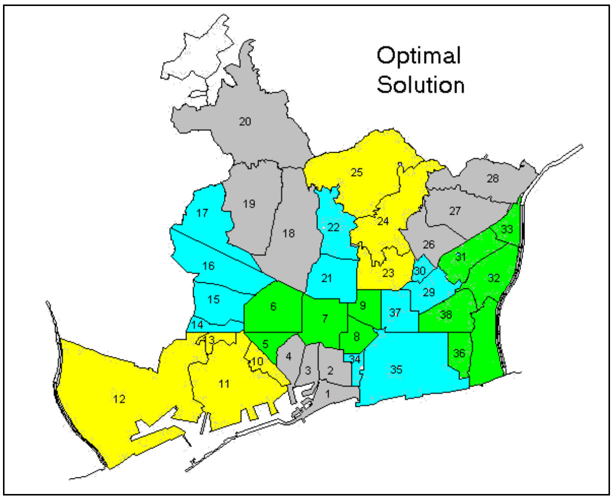
The relationships among areas (see Table 11) were created in a way that the optimal solution grouped the 38 areas in 10 regions, each of them with different shapes and sizes (among 2 and 6 areas by region). This optimal solution is shown in figure 8, and this is the solution that the *RASS* algorithm should be able to identify.

Table 11. Relationships matrix between the 38 areas

	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38
1	0.006	0.000	0.038	0.328	0.330	0.319	0.359	0.340	0.676	0.661	0.671	0.682	0.951	0.977	0.972	0.964	1.270	1.316	1.287	1.603	1.625	1.928	1.898	1.928	2.243	2.235	2.257	2.565	2.558	2.882	2.866	2.873	2.574	2.581	2.901	2.586	2.853
2		0.006	0.032	0.322	0.324	0.313	0.353	0.335	0.670	0.655	0.666	0.677	0.945	0.971	0.966	0.958	1.264	1.310	1.282	1.597	1.620	1.923	1.893	1.923	2.237	2.229	2.252	2.560	2.552	2.876	2.861	2.868	2.568	2.575	2.895	2.580	2.847
3			0.038	0.328	0.330	0.319	0.359	0.340	0.676	0.661	0.671	0.682	0.951	0.977	0.972	0.964	1.270	1.316	1.287	1.603	1.625	1.928	1.898	1.928	2.243	2.235	2.257	2.565	2.558	2.882	2.866	2.873	2.574	2.581	2.901	2.586	2.853
4				0.290	0.291	0.281	0.321	0.302	0.638	0.622	0.633	0.644	0.913	0.939	0.934	0.925	1.231	1.278	1.249	1.565	1.587	1.890	1.860	1.890	2.205	2.197	2.219	2.527	2.520	2.844	2.828	2.835	2.535	2.542	2.863	2.548	2.815
5					0.002	0.009	0.031	0.013	0.348	0.333	0.344	0.354	0.623	0.649	0.644	0.636	0.942	0.988	0.960	1.275	1.298	1.601	1.571	1.601	1.915	1.907	1.930	2.238	2.230	2.554	2.539	2.546	2.246	2.253	2.573	2.258	2.525
6						0.011	0.029	0.011	0.346	0.331	0.342	0.353	0.621	0.647	0.642	0.634	0.940	0.986	0.958	1.273	1.296	1.599	1.569	1.599	1.913	1.905	1.928	2.236	2.228	2.552	2.537	2.544	2.244	2.251	2.571	2.256	2.523
7							0.040	0.022	0.357	0.342	0.353	0.363	0.632	0.658	0.653	0.645	0.951	0.997	0.969	1.284	1.307	1.609	1.580	1.609	1.924	1.916	1.939	2.247	2.239	2.563	2.548	2.555	2.255	2.262	2.582	2.267	2.534
8								0.018	0.317	0.302	0.312	0.323	0.592	0.618	0.613	0.605	0.911	0.957	0.929	1.244	1.266	1.569	1.539	1.569	1.884	1.876	1.898	2.206	2.199	2.523	2.507	2.514	2.215	2.222	2.542	2.227	2.494
9									0.335	0.320	0.331	0.342	0.610	0.636	0.631	0.623	0.929	0.976	0.947	1.263	1.285	1.588	1.558	1.588	1.902	1.895	1.917	2.225	2.217	2.541	2.526	2.533	2.233	2.240	2.560	2.245	2.513
10										0.015	0.004	0.006	0.275	0.301	0.296	0.288	0.594	0.640	0.612	0.927	0.950	1.252	1.223	1.252	1.567	1.559	1.581	1.890	1.882	2.206	2.191	2.198	1.898	1.905	2.225	1.910	2.177
11											0.011	0.022	0.290	0.316	0.311	0.303	0.609	0.656	0.627	0.943	0.965	1.268	1.238	1.268	1.582	1.574	1.597	1.905	1.897	2.221	2.206	2.213	1.913	1.920	2.240	1.925	2.192
12												0.011	0.279	0.305	0.300	0.292	0.598	0.645	0.616	0.932	0.954	1.257	1.227	1.257	1.571	1.564	1.586	1.894	1.886	2.210	2.195	2.202	1.902	1.909	2.229	1.914	2.182
13													0.269	0.295	0.290	0.281	0.587	0.634	0.605	0.921	0.943	1.246	1.216	1.246	1.560	1.553	1.575	1.883	1.876	2.199	2.184	2.191	1.891	1.898	2.219	1.904	2.171
14														0.026	0.021	0.013	0.319	0.365	0.337	0.652	0.675	0.978	0.948	0.978	1.292	1.284	1.307	1.615	1.607	1.931	1.916	1.923	1.623	1.630	1.950	1.635	1.902
15															0.005	0.013	0.293	0.339	0.311	0.626	0.649	0.951	0.922	0.951	1.266	1.258	1.280	1.588	1.581	1.905	1.890	1.897	1.597	1.604	1.924	1.609	1.876
16																0.008	0.298	0.344	0.316	0.631	0.654	0.957	0.927	0.957	1.271	1.263	1.286	1.594	1.586	1.910	1.895	1.902	1.602	1.609	1.929	1.614	1.881
17																	0.306	0.353	0.324	0.640	0.662	0.965	0.935	0.965	1.279	1.272	1.294	1.602	1.594	1.918	1.903	1.910	1.610	1.617	1.937	1.622	1.890
18																		0.046	0.018	0.333	0.356	0.659	0.629	0.659	0.973	0.965	0.988	1.296	1.288	1.612	1.597	1.604	1.304	1.311	1.631	1.316	1.583
19																			0.029	0.287	0.309	0.612	0.582	0.612	0.927	0.919	0.941	1.249	1.242	1.566	1.550	1.557	1.258	1.265	1.585	1.270	1.537
20																				0.316	0.338	0.641	0.611	0.641	0.955	0.948	0.970	1.278	1.270	1.594	1.579	1.586	1.286	1.293	1.613	1.298	1.566
21																					0.022	0.325	0.295	0.325	0.640	0.632	0.654	0.962	0.955	1.279	1.263	1.270	0.971	0.978	1.298	0.983	1.250
22																						0.303	0.273	0.303	0.617	0.610	0.632	0.940	0.932	1.256	1.241	1.248	0.948	0.955	1.275	0.960	1.228
23																							0.030	0.000	0.314	0.307	0.329	0.637	0.629	0.953	0.938	0.945	0.645	0.652	0.972	0.657	0.925
24																								0.030	0.344	0.337	0.359	0.667	0.659	0.983	0.968	0.975	0.675	0.682	1.002	0.687	0.955
25																									0.314	0.307	0.329	0.637	0.629	0.953	0.938	0.945	0.645	0.652	0.972	0.657	0.925
26																										0.008	0.015	0.323	0.315	0.639	0.624	0.631	0.331	0.338	0.658	0.343	0.610
27																											0.022	0.330	0.323	0.647	0.631	0.638	0.339	0.346	0.666	0.351	0.618
28																												0.308	0.300	0.624	0.609	0.616	0.316	0.323	0.643	0.328	0.596
29																													0.008	0.316	0.301	0.308	0.008	0.015	0.335	0.020	0.288
30																														0.324	0.309	0.316	0.016	0.023	0.343	0.028	0.295
31																															0.015	0.008	0.308	0.301	0.019	0.296	0.029
32																																0.007	0.293	0.286	0.034	0.281	0.013
33																																	0.300	0.293	0.027	0.288	0.020
34																																		0.007	0.327	0.012	0.279
35																																			0.320	0.005	0.272
36																																				0.315	0.048
37																																					0.267

Source: Own elaboration.

Figure 8. Preestablished optimal regional configuration



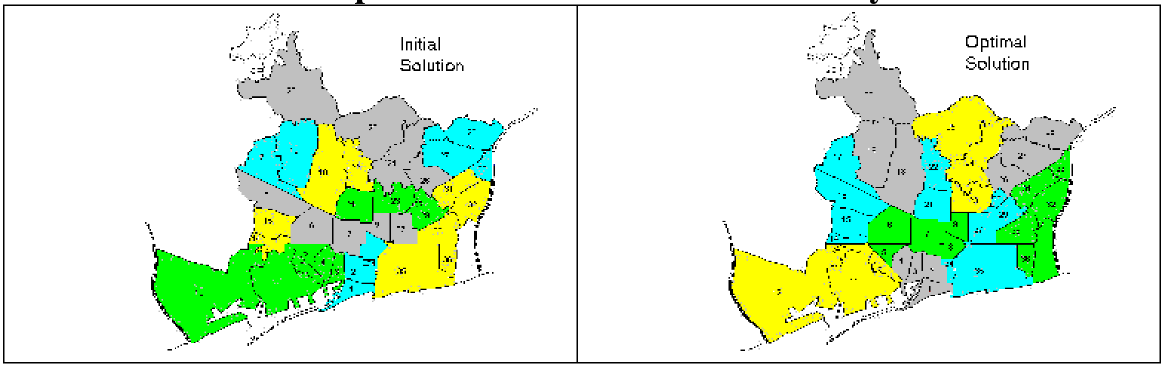
Source: Own elaboration.

4.3.2. Evaluation of results

The initial considered partition is shown in Table 12. This is the partition that is considered by the *RASS* in the step 1. It is worth mentioning that this configuration is very different to the optimal one. After 5 cycles, the *RASS* algorithm properly reaches the optimal solution.

The different regional configurations considered by the *RASS* in the different steps and iterations are shown in Annex 2.

Table 12. Initial partition and solution obtained by the *RASS*



Source: Own elaboration.

In order to evaluate the evolution of the results from the initial partition up to the final results, table 13 presents the value of the objective function and the associated regional configuration at the end of each cycle in the application of the algorithm. The value of the objective function for the initial partition is 34.36 and in the first cycle a reduction of 24.15 is achieved. This value is reduced in the following cycles until achieving its minimum value in 1.08.

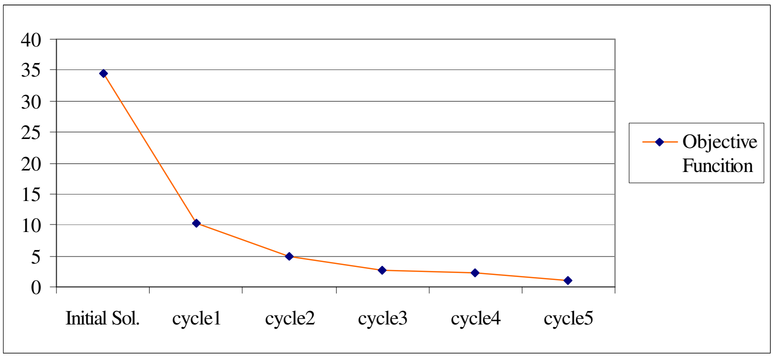
Table 13. Values of the objective function in the initial partition and at the end of each cycle

Regions	Initial	cycle 1	cycle 2	cycle 3	cycle 4	cycle 5
1	10.35	5.21	2.21	1.04	1.04	0.23
2	8.07	2.21	1.04	0.93	0.30	0.18
3	5.61	1.70	0.93	0.23	0.23	0.16
4	3.52	0.60	0.23	0.16	0.18	0.13
5	2.89	0.13	0.13	0.13	0.16	0.10
6	1.34	0.10	0.11	0.09	0.13	0.09
7	1.28	0.09	0.09	0.07	0.09	0.07
8	0.59	0.07	0.07	0.04	0.07	0.06
9	0.36	0.06	0.06	0.02	0.04	0.04
10	0.35	0.04	0.04	0.02	0.03	0.02
Objective function	**34.36**	**10.21**	**4.91**	**2.73**	**2.27**	**1.08**

Source: Own elaboration.

As it can be appreciated in figure 9, the behaviour of the objective function is similar to the expected one: in the first cycles is where higher improvements are achieved. Also, it is confirmed that in every cycle the value of the objective function is improved, or at worst equal, in relation to the previous cycle.

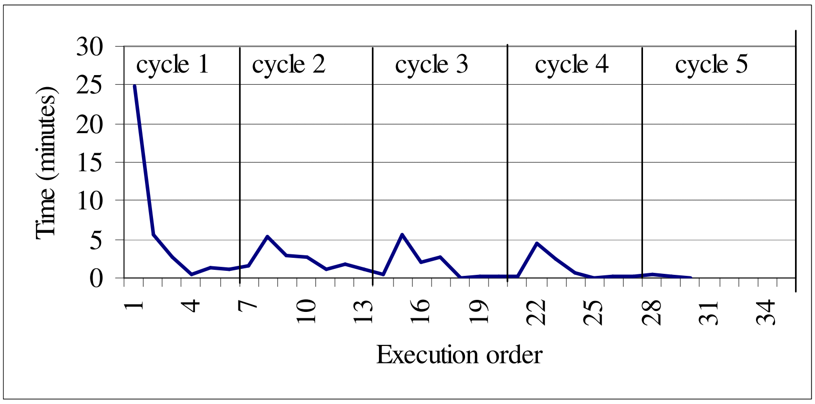
Figure 9. Evolution of the objective function during the application of *RASS*



Source: Own elaboration.

The number of regions in the optimisation model was set to 4 (r = 4). With this value, the average number of areas where each optimisation model was running was 15. This number was enough to permit that the running times where appropriated with an average running time of 2.43 minutes by model. These running times are shown in figure 10.

Figure 10. Running times of optimisation models



Source: Own elaboration.

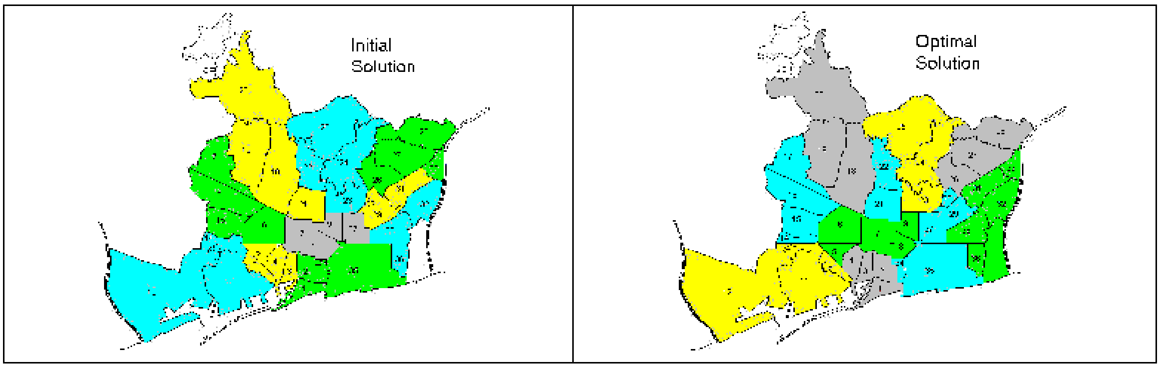
As it can be seen, the running times of the different optimisation models were higher at the beginning of each cycle and, in particular, for the first time it is executed (although it is also when a higher reduction in the objective function is achieved). This is related with the fact that in the first model of each cycle is executed considering the 4 (*r)* most heterogeneous regions, which can imply that the reassignation of the areas in these *r* regions can be very high. For this example, the first model has reassignated the 37% of these areas (or a 18.4% if we take into account the 38 areas) and has achieved a reduction in the objective function of 13.18 points, a 54.6% of the reduction obtained in the first cycle (or a 39.6% of the total reduction).

4.3.3. Sensitivity of the results to the initial partition

How can the initial partition affect to the final result? In this sub-section, we would use a different initial partition to the same problem as above. In particular, the initial partition in the step 1 of *RASS* will be closer to the optimum regional configuration. With this partition, we should expect a lower number of cycles and similar results as in the previous sub-section.

In this case, the optimal configuration was found after 2 cycles (see table 14), 3 cycles less than in the previous example. The results shown in the table 15 and in the figure 11, permit to conclude that, as before, the higher reductions in the objective function are achieved in the initial cycles of the *RASS*.

Table 14. Initial partition (close to optimum) and obtained solution



Source: Own elaboration.

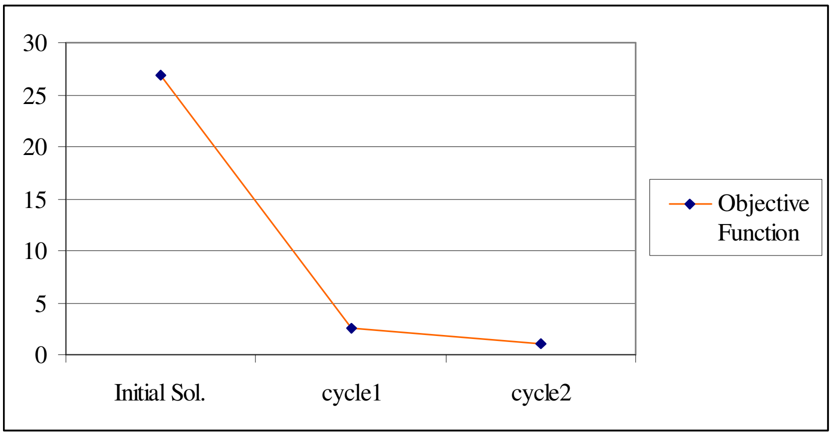
Regarding the impact of the first optimisation model on the objective function, now there is a reduction of 19.33 points (from 26.94 to 7.61), a 79,25% of the total obtained reduction in the first cycle. The 50% of the areas in the 4 (*r*) considered regions are now reassigned (a 21.1% in the 38 areas are considered).

Table 15. Values of the objective function in the initial partition (closes to the optimal solution) and at the end of each cycle

Regions	Initial	cycle 1	cycle 2
1	10.31	1.71	0.23
2	6.83	0.18	0.18
3	2.33	0.15	0.16
4	1.95	0.13	0.13
5	1.93	0.10	0.10
6	1.04	0.09	0.09
7	0.93	0.07	0.07
8	0.88	0.06	0.06
9	0.65	0.04	0.04
10	0.09	0.02	0.02
Objective function	**26.94**	**2.55**	**1.08**

Source: Own elaboration.

Figure 11. Evolution of the objective function during the application of *RASS* with the initial partition closes to the optimal solution



Source: Own elaboration.

The obtained results permit to conclude that the RASS, due to the incorporation of a direct optimisation routine as part of the algorithm, has a big capacity to achieve global optimums in the context of regionalisation problems. However, it is worth mentioning that the relationship between the number of regions (*m*) and the number of areas (*n*) should be defined as a way that the number of regions considered by the optimisation model (*r*) must be 2 or higher and these regions should contain a number of areas in line with the computational capacity of the model. We have calculated that the most appropriate relationship *m/n* must be above the 14%. For example, if we considered a territory formed by 8000 areas, the number of regions that can be obtained will be higher or equal than 1120 regions (an average size of 7 areas per region). This relationship ensures that *r* can take values higher or equal than 2 without increasing substantially the running time.

If the relationship between regions and the number of areas is very low, one possible strategy could consist in designing nested regionalisation problems, which would imply the sequential application of the *RASS*. For example, the city of Barcelona is divided in 1919 statistical sections (*Seccions Estadístiques*, SE), which are grouped in 248 small research areas (*Zones of Recerca Petites*, ZRP). These areas are also grouped in 110 basic statistical units (*Unitats Estadístiques Bàsiques*, UEB) that form the 38 big statistical areas (*Zones Estadístiques Grans*, ZEG). Last, the big statistical areas are grouped to obtained the 10 districts of the city[28]. Each territorial level is formed grouping the previous one, and this also guarantees that the different grouping levels are self-contained.

[28] For more information, see: http://www.bcn.es/estadisitica/catala/terri/index.htm.

5. CONCLUSIONS

The objective of this paper was to propose a new methodology to design regions from lower level territorial units (areas) considering not only their characteristics but also the relationships among them.

This methodology permits to avoid the use of *ad-hoc* regionalisation to obtain territorial units that are representative of the considered phenomenon. This aspect is especially relevant as statistical and econometrical results are sensitive to different levels of aggregation and scale.

We have proposed the use of a lineal optimisation model to find the optimal aggregation of different areas in a given number of regions from the consideration of a geographical contact matrix and a relationships matrix. The minimisation of the "internal" heterogeneity of each region permits to find homogeneous regions according to the considered criteria.

The possibility of treating the regionalisation problem as a linear model permits to ensure that, by its mathematical properties, the feasible region is convex and, as a result, it is possible to find the optimal solution. Another advantage of this kind of formulation is that it is easy to implement in a great variety of commercial software.

The obtained empirical evidence permits to affirm that the proposed methodology has a great capacity to identify different complex territorial configurations. The model takes into account the contiguity constraint but without conditioning the shapes that those regions can adopt.

We would also like to highlight that the model permits to easily introduce additional restrictions in the regionalisation process. As an example, we have shown the possibility of introducing two additional restrictions: the minimum population requirement and the mandatory isolation.

An algorithm called *RASS* (*Regionalisation Algorithm with Selective Search*) has also been introduced as a way of improving the computational capacity of the model. This algorithm tries to take profit of the advantages of applying direct optimisation to a given territorial portion that varies in each iteration, thanks to a selective search strategy. These characteristics permit the *RASS* to escape from local optimum.

The obtained results with the RASS have shown its utility, as in a 100% of the considered simulations the global optimum was found and in a running time considerably lower than the one obtained applying the direct optimisation model.

6. REFERENCES

Aarts, E. and Lenstra, J. K. (1997), *Local search in combinatorial optimization*. Chichester, New York, [etc.] John Wiley and Sons.

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993), *Network flows : theory, algorithms, and applications*. Englewood Cliffs, N.J. Prentice Hall, cop.

Albert, J. M., Mateu, J. and Orts, V. (2003), *Concentración versus dispersion: Un análisis especial of la localización of la actividad económica en la U.E.*, mimeo.

Amrhein, C. G. and Flowerdew, R. (1992), "The effect of data aggregation on a Poisson regression model of Canadian migration", *Environment and Planning A*, 24, 1381-91.

Aurenhammer, F. (1991), "Voronoi diagrams - a survey of a fundamental geometric data structure", *ACM Computing Surveys*, 23, 345-405.

Battiti, R. and Tecchiolli, R. (1994), "The reactive tabu search", *ORSA Journal on Computing*, 6 (2), 126-40.

Benabdallah, S. and Wright, J. R. (1992), "Multiple subregion allocation models", *Journal of Urban Planning and Development*, 118 (1), 24-40.

Browdy, M. (1990), "Simulatea Annealing - an improved computer model for political redistricting", *Yale Law and Policy Review*, 8, 163-79.

Calciu, M. (1996), "Une méthode of classification sous contrainte of contiguïté en géo-marketing." Cahiers of recherche of l'IAE of Lille (96/5). Université des Sciences et Technologies of Lille.

Dantzing, G. B. and Ramser, J. H. (1959), "The truch dispatching problem", *Management Science*, 6, 80-91.

Ferligoj, A. and Batagelj, V. (1982), "Clustering with relational constraint", *Psychometrika*, 47, 413-26.

Ferligoj, A. and Batagelj, V. (1983), "Some types of clustering with relational constraints", *Psychometrika*, 48, 541-52.

Fisher, M. M. (1980), "Regional taxonomy", *Regional Science and Urban Economics*, 10, 503-37.

Fotheringham, A. S. and Wong, D. W. S. (1991), "The modifiable areal unit problem in multivariate statistical analysis", *Environment and Planning A*, 23, 1025-44.

Glover, F. (1977), "Heuristic for integer programming using surrogate constraints”, *Decision Sciences*, 8, 156-66.

Glover, F. (1989), "Tabu search, part I”, *ORSA Journal on Computing*, 1, 190-206.

Glover, F. (1990), "Tabu search, part II”, *ORSA Journal on Computing*, 2, 4-32.

Gordon, A. D. (1996), "A survey of constrained classification”, *Computational Statistics & Data Analysis*, 21, 17-29.

Gordon, A. D. (1999), Classification (segunda edición ed.). Boca Raton [etc.].

Gower, J. C. and Legendre, P. (1986), "Metric and euclidean properties of dissimilarity coefficients”, *Journal of Classification*, 3, 5-48.

Graham, R. R. and Hell, P. (1985), "On the history of the minimum spanning tree problem”, *Annals of the history of computing*, 7, 43-57.

Haining, R. P., Wise, S. M., and Ma, J. (1996), "The design of a software system for interactive spatial statistical analysis linked to a GIS”, *Computational Statistics*, 11, 449-66.

Hiriart, J. B., Oettli, W., and Stoer, J. (1983), Optimization: Theory and Algorithms. J. Baptiste and otros, (eds.) New York [etc.]: Marcel Dekker, cop.

Horn, M. E. T. (1995), "Solution techniques for large regional partitioning problems”, *Geographical Analysis*, 27, 230-48.

Jobson, J. D. (1991), *Applied multivariate data analysis: Categorical and multivariate methods*. New York, Barcelona [etc.] Springer.

Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983), "Optimization by simulated annealing”, *Science*, 220, 671-80.

Kohonen, T. (1984), *Self-Organisation and Associative Memory*. Berlin [etc.] Springer.

Laport, G. and Osman, I. H. (1995), "Routing problems: A bibliography”, *Annals of Operations Research*, 61, 227-62.

López-Bazo, E., Vaya, E., Mora, A. and Suriñach, J. (1999), "Regional Economic Dynamics and Convergence in the European Union", *Annals of Regional Science*, 33, 343-370.

Macmillan, B. and Pierce, T. (1994), "Optimization modelling in GIS framework: the problem of political redistricting”, in *Spatial analysis and GIS*, Stewart Fotheringham and Peter Rogerson, (eds.). London [etc.]: Taylor & Francis, pp 221-46.

Macmillan, W. (2001), "Redistricting in a GIS environment: An optimization algorithm using switching-points", *Journal of Geographical Systems*, 3, 167-80.

Maravalle, M. and Simeone, B. (1995), "A spanning tree heuristic for regional clustering", *Communications in Statistics - Theory and Methods*, 24 (3), 625-39.

Martin, D., Nolan, A. and Tranmer, M. (2001), "The application of zone-design methodology in the 2001 UK Census", *Environment and Planning A*, 33, 1949-62.

Matula, D. W. and Sokal, R. R. (1980), "Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane", *Geographical Analysis*, 12, 205-22.

Murtagh, F. (1985), "A survey of Algorithms for Contiguity-constrained Clusterind and Related Problems", *The Computer Journal*, 28 (1), 82-88.

Neves, M. C., Freitas, C. C. and Câmara, G. (2001), "Mineraçâo of datos em grandes bancos of datos geographical." Brasil: Instituto nacional of pesquisas espaciais. Ministério da ciência e tecnologia.

Ohsumi, N. (1984), "Practical techniques for areal clustering", in *Data analysis and informatics, Vol III*, E. Diday, M. Jambu, L. Lebart, J. Pagès and R. Tomassone, (eds.) Vol. III. North-Holland, Amsterdam, pp 247-58.

Openshaw, S. (1977), "Algorithm3: a procedure to generate pseudo random aggregation of N zones into M zones where M is less than N", *Environment and Planning A*, 9, 1423-28.

Openshaw, S. (1984), "The modifiable areal unit problem", *Concepts and Techniques in Modern Geography*, 38 (GeoAbstracts, Norwich).

Openshaw, S. (1992), "Some suggestions concerning the development of artificial intelligence tools for spatial modelling and analysis in GIS", *The Annals of Regional Science*, 26, 35-51.

Openshaw, S., Alvanides, S. and Whalley, S. (1998), *Some further experiments with designing output areas for the 2001 UK census*, Centre for Computational Geography. School of Geography. University of Leeds, Leeds.
(http://www.geog.leeds.ac.uk/people/s.alvanides/papers/somefur4.html)

Openshaw, S. and Rao, L. (1995), "Algorithms for reengineering 1991 census geography", *Environment and Planning A*, 27, 425-46.

Openshaw, S. and Taylor, P. J. (1981), "The modifiable areal unit problem", in *Quantitative Geography*, N. Wrigley and R. J. Bennett, (eds.). London, pp 60-70.

Openshaw, S. and Wymer, C. (1995), "Classifying and regionalizing census data", in *Census Users Handbook*, S. Openshaw, (eds.). Cambridge, UK: Geo Information International, pp 239-70.

Perruchet, C. (1983), "Constrained agglomerative hierarchical classification", *Pattern Recognition*, 16, 213-17.

Semple, R. K. and Green, M. B. (1984), "Classification in human geography", in *Spatial statistics and models*, G. L. Gaile and C. J. Wilmott, (eds.). Reidel, Dordrecht, pp 55-79.

Toussaint, G. T. (1980), "The relative neighbourhood graph of a finite planar set", *Pattern Recognition*, 12, 261-68.

Webster, R. and Burrough, P. A. (1972), "Computer-based soil mapping of small areas from sample data II. Classification smoothing", *Journal of Soil Science*, 23, 222-34.

Wise, S. M., Haining, R. P. and Ma, J. (1997), "Regionalization Tools for Exploratory Spatial Analysis of Health Data", in *Recent Developments in Spatial Analysis: Spatial statistics, behavioural modelling, and computational intelligence*, Manfred M. Fisher and Arthur Gentis, (eds.). Berlin [etc.]: Springer, pp 83-100.

ANNEXES

Annex 1. Using heuristics for the configuration of regions: some solutions proposed in the literature

In this annex we present a brief description of the heuristics with a higher impact in the field of regionalisation and that fulfil the following conditions: their objective is to divide a territory in a pre-defined number of regions and the areas to be grouped do not have a specific role. So, we are not considering here the heuristics applied in hierarchical partitions and those that try to find core areas or centroids in order to assign the rest of the areas.

a) Automatic Zoning Procedure (AZP)

This heuristic proposed by Openshaw (1977) is based in an iterative procedure. It consists in the optimisation of an objective function $F(Z)$, where Z is the allocation of each of the N zones to one of M regions such that each zone is assigned to only one region and each region should have at least one zone.

The AZP algorithm consists of the following steps:

Step 1 Start by generating a random zoning system of N small zones into M regions, $M<N$.

Step 2 Make a list of the M regions.

Step 3 Select and remove any region K at a random from this list.

Step 4 Identify a set of zones bordering on members of region *K* that could be moved into region *K* without destroying the internal contiguity of the donor region(s).

Step 5 Randomly select zones from this list until there is a local improvement in the current value of the objective function *or* a move that is equivalently as good as the current best. Then make the move, update the list of candidate zones, and return to step 4 or else repeat step 5 until the list is exhausted.

Step 6 When the list for region *K* is exhausted return to steps 3, select another region, and repeat steps 4-6.

Step 7 Repeat steps 2-6 until no further improving moves are made.

Among the main advantages of this heuristic, there is the possibility of using any objective function sensitive to the aggregation of zones. This characteristic is of great utility to approximate limit of aggregation effects. It has also been useful to demonstrate that MAUP exists.

The main disadvantages of this heuristic are related to the local search procedure (restricted to the selected region) and to the strong dependence of the results to selected starting point (step 1). Last, the strategy of not considering the possibility of moving a zone that implies a decrease in the objective function can produce that the heuristic is trapped in a local optimum. Openshaw tried to solve this problem in later proposals.

b) Simulated Annealing Variant of AZP (AZP-SA)

This proposal of Openshaw and Rao (1995) consists in a modification to the AZP. In particular, the step 5 now consists in "*Randomly sample this list* *until there is a local improvement in the objective function or an equivalently good move. Then make the move. Otherwise make the move with a probability given Boltzmann's equation"*:

$$R(0,1) < \exp\left(\frac{\nabla f}{T(k)}\right) \qquad (28)$$

∇f	is the change in the objective function caused by the move.
$T(k)$	is the temperature being applied at annealing time step k.
$R(0,1)$	is a uniformly distributed random number in the range 0.0 to 1.0.

The interest of this modification consists in the possibility of moving towards solution that decrease the objective function, but with a probability that diminishes gradually, through iteration time.

In this heuristic, special attention should be given to the definition of the initial value of $T(0)$ and the cooling schedule, looking for an appropriate "*trade-off*" between the execution time and a good solution. Openshaw adopts an exponential cooling scheme where the temperature in k is equal to a fraction of the temperature in k-1, this is: $T(k)=fT(K-1)$ where f is typically between 0.8 and 0.95.

So, the AZP-SA can be summarised in the following steps:

Step a Set $T(0)$, k=0.

Step b Apply AZP with the modified step 5 until either MAXIT (a user-defined maximum number of) iterations or convergence or at least a minimum of Q simulated annealing moves have been made.

Step c Update T and k: $T(k)=0.85 \cdot T(k-1)$ and $k=k+1$.

Step d Repeat steps b and c until no further moves occur over at least three different *k* values.

Macmillan and Pierce (1994) apply the *Simulated Annealing* in the *Redistricting Problem*. Their heuristic, called *ANNEAL redistricting problem*, is defined to group *C* counties in *D* districts with the restrictions that each district should contain at least one county, and each county can only be assigned to one district. The optimisation criterion is the minimisation of the sum of the squares of the deviations of the district populations from their population target (*P*/*D* of the state's P electors).

Taking into account that a big amount of the complexity of the proposed heuristics for regionalisation problems is related with the control of contiguity, Macmillan (2001) proposes a regionalisation algorithm called SARA, which incorporates a more efficient methodology to control it based in the concept of *switching points*. This new proposal improves significantly the execution times obtained by Openshaw and Rao (1995).

c) Tabu Search Algorithm (AZP-TABU)

This heuristic was adapted by Openshaw for regionalisation problems. Its main advantage is the possibility of achieving similar results to the *Simulated Annealing*, but with a lower computational cost. The AZP-TABU steps are the following:

Step 1 Find the global best move that is not prohibited or tabu.

Step 2 Make this move if it is an improvement or equivalent in value, else:

Step 3 If no improving move can be made, then see if a tabu move can be made which improves on the current local best (termed an aspiration move), else:

Step 4 If there is no improving and no aspirational move, then make the best move even if it is nonimproving (that is, results in a worse value of the objective function).

Step 5 Tabu the reverse move for R iterations.

Step 6 Return to step 1.

This algorithm is a powerful optimisation tool as it allows the possibility to escape from local optimums or cyclical behaviour. However, its main disadvantage is related with the definition of an adequate value of R, as the results depend heavily on this parameter. Battiti and Tecchiolli (1994) propose the *Reactive Tabu Search* where R is dynamically adjusted.

d) Heuristic based on spanning trees for territorial aggregation

The heuristic proposed by Maravalle and Simeone (1995), called MIDAS (*Méthode Itérative D'Agrégation Spatiale*) incorporates the relationships between vertices (areas) with the objective of building homogeneous regions with respect to a certain set of characteristics. The problem is formulated in the following way: *"Given a connected graph G, in which a vector of characteristics is associated with each vertex, find a minimum inertia partition of the vertex-set of G into a prescribed number of connected clusters"*.

The proposed heuristic follows a strategy based in the simplification of G, in a way that G is replaced by one of its spanning tree T in which each couple of vertices are connected by one and only one group of arcs. The group of arcs belonging to *T* are a subgroup of the arcs belonging to *G*. The most relevant characteristic of *T* is that deleting one of its arcs will generate a partition of the vertices in two groups connected inside but disconnected among them. This result is consistent with the regionalisation requirements.

The MIDAS heuristic can be summarised in the following steps[29]:

Step 1 (Initial tree) Find a good initial spanning tree *T* of *G*.

Step 2 (Initial partition) Find a good initial partition $\hat{\pi}$ from the whole group of possible partitions of *G*, $\prod_p(T)$;

Step 3 (Tree-optimisation) Starting from $\hat{\pi}$, perform a local search to find a near-optimal solution π^* to the problem

$$\min\{f(\pi):\pi \in \prod_p(T)\}$$

Step 4 (Tree-modification) Attempt to find, if possible, another p-partition $\bar{\pi}$ and another tree $\bar{T}$ of *G* such that:

$$f(\bar{\pi}) < f(\pi^*) \qquad \text{(i)}$$

$$\bar{\pi} \notin \prod_p(T) \qquad \text{(ii)}$$

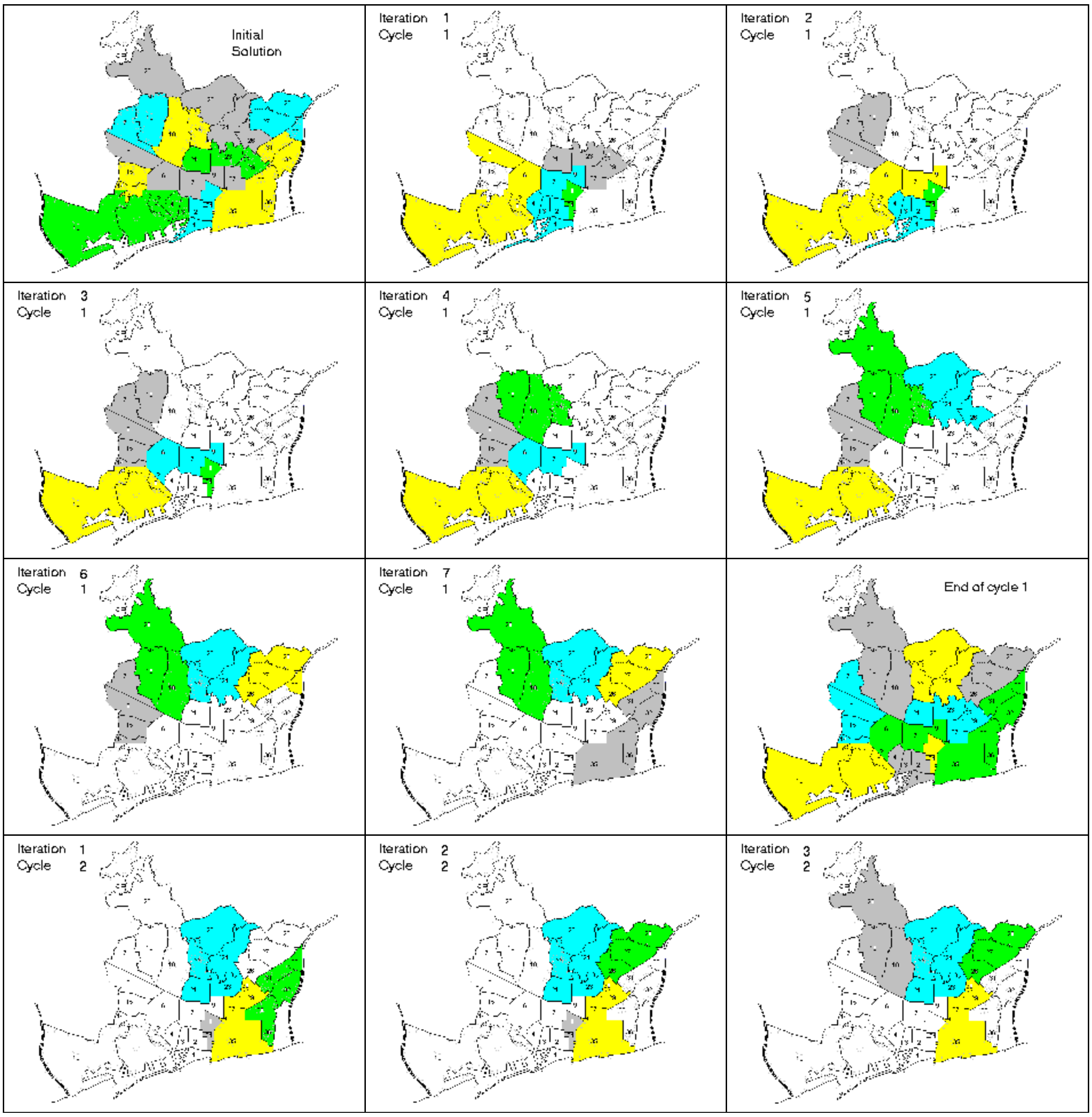
$$\bar{\pi} \in \prod_p(\bar{T}) \qquad \text{(iii)}$$

If no such pair $(\bar{\pi}, \bar{T})$ can be found, then stop: output the current partition π^* (since π^* is feasible in *T*, it is also feasible in *G*); else replace $\hat{\pi}$ by $\bar{\pi}$ and go to step 3.

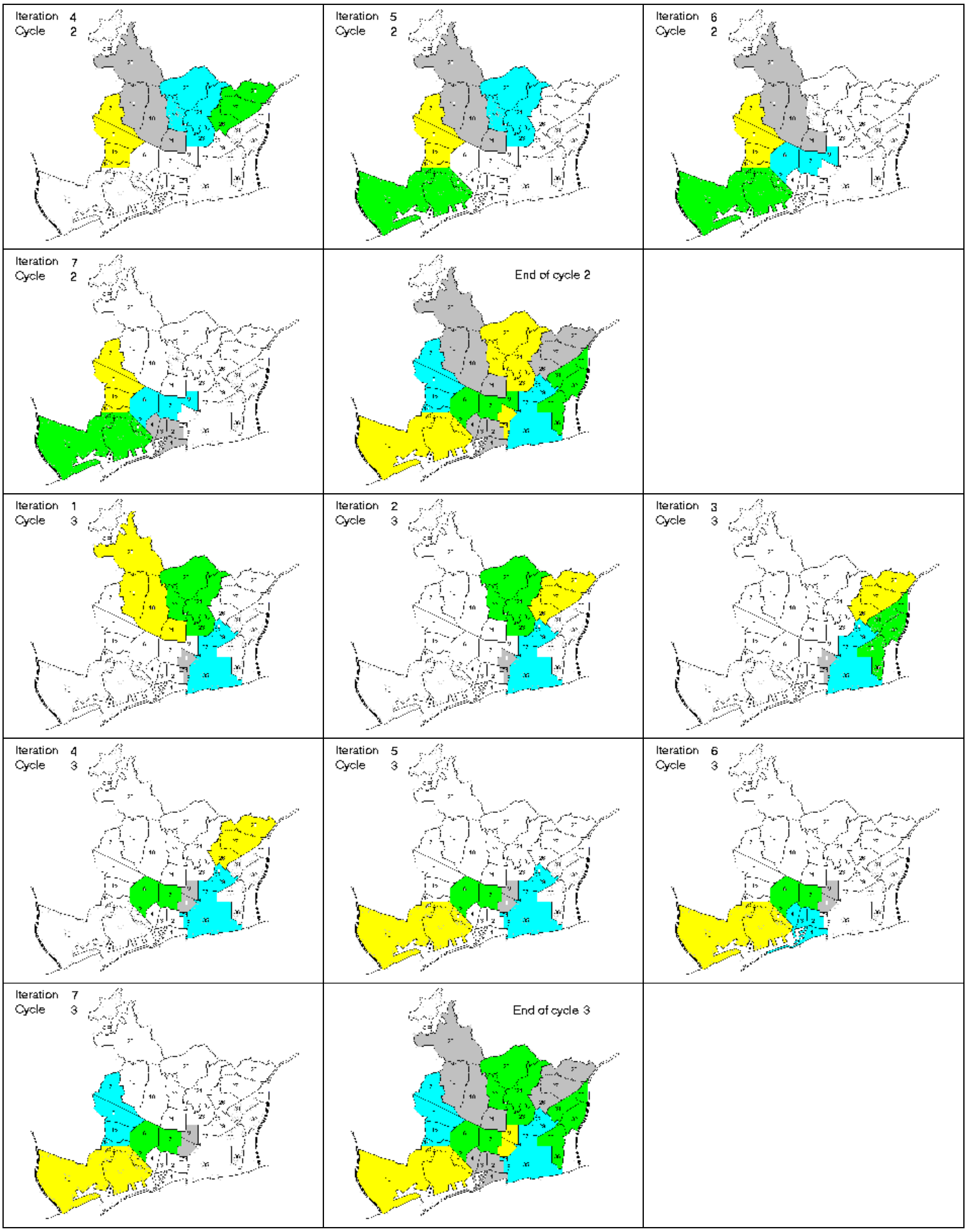
[29] A detailed description of the different steps can be found in Maravalle and Simeone (1995).

The main inconvenients associated to this methodology are: the loss of control on the number of elements included in each partition, and, more relevant, the utilisation of arcs of the contact matrix *G* as a way to represent the relationships between vertices, since this imply not considering other relationships between non-adjacent vertices.

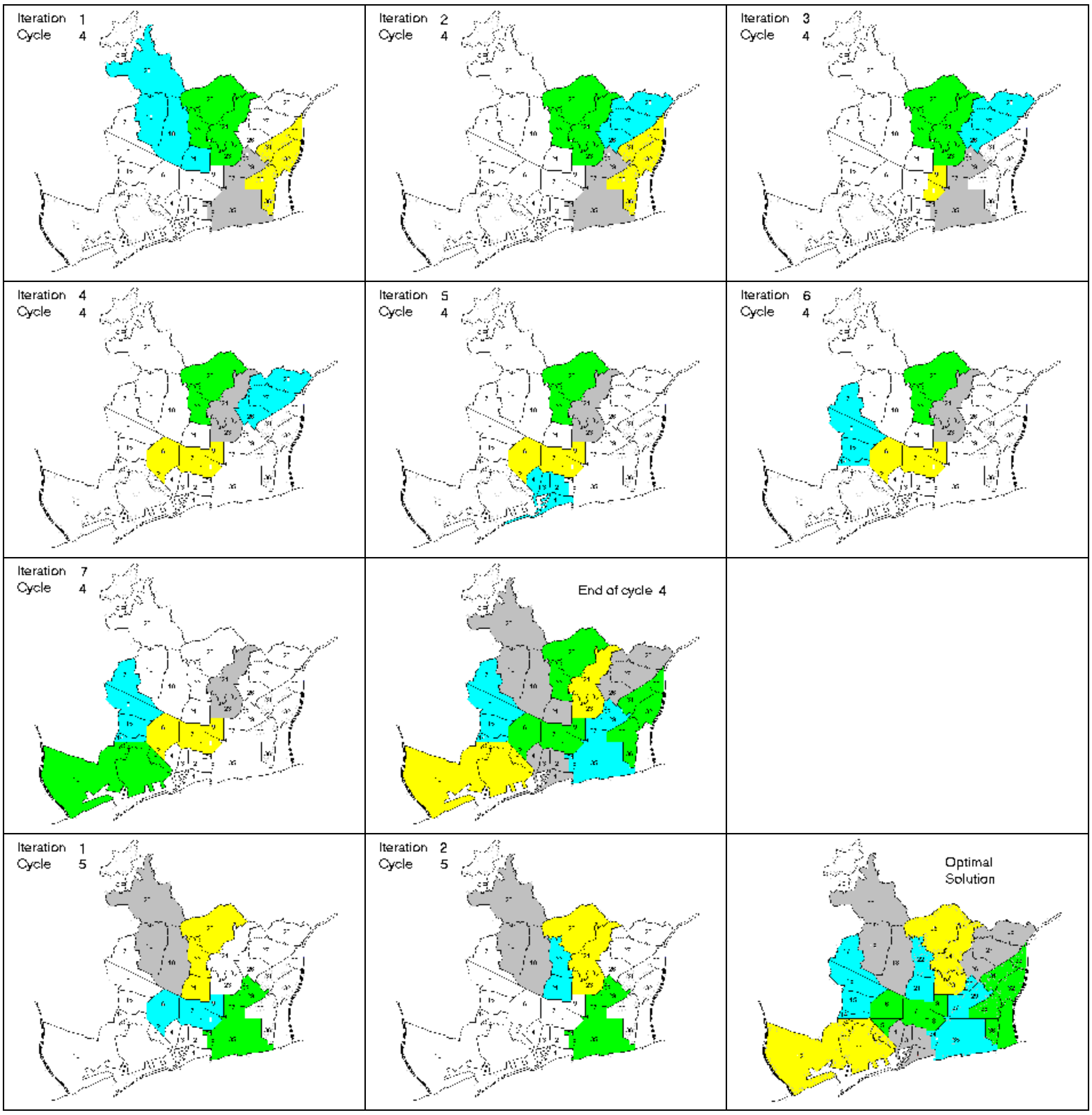
Annex 2. Maps of the different territorial configurations obtained using RASS



Continue...


Iteration 4
Cycle 2
Iteration 5
Cycle 2
Iteration 6
Cycle 2
Iteration 7
Cycle 2
End of cycle 2
Iteration 1
Cycle 3
Iteration 2
Cycle 3
Iteration 3
Cycle 3
Iteration 4
Cycle 3
Iteration 5
Cycle 3
Iteration 6
Cycle 3
Iteration 7
Cycle 3
End of cycle 3

Continue...


Iteration 1
Cycle 4
Iteration 2
Cycle 4
Iteration 3
Cycle 4
Iteration 4
Cycle 4
Iteration 5
Cycle 4
Iteration 6
Cycle 4
Iteration 7
Cycle 4
End of cycle 4
Iteration 1
Cycle 5
Iteration 2
Cycle 5
Optimal Solution