
# Design of Homogenous Territorial Units. A Methodological Proposal and Applications

Juan Carlos Duque Cardona

Departamento de Econometría Estadística y Economía Española
UNIVERSIDAD DE BARCELONA

# DESIGN OF HOMOGENOUS TERRITORIAL UNITS.

## *A Methodological Proposal and Applications.*



***Juan Carlos Duque Cardona***

MAYO 2004

Tesis Doctoral para optar al título de Doctor en Estudios Empresariales
Director: Dr. Manuel Artís Ortuño
Doctorado en Estudios Empresariales
Programa: Técnicas y Análisis en la Empresa
Bienio 2000-2002

*A mi madre y a mi abuela que en paz descansen*

*Agradecimientos*

Gracias a la Generalitat de Catalunya por su apoyo a través de la beca de formación de personal investigador 2001FI 00296.

Al doctor Miguel Ángel Sierra, director del Departamento de Econometría, Estadística y Economía Española de la Universidad de Barcelona por poner a mi disposición los recursos humanos y técnicos, sin los cuales no hubiera sido posible escribir esta tesis.

Gracias a los doctores Jordi Suriñach y Manuel Artís, directores del grupo de investigación de Anàlisi Quantitativa Regional (AQR) por haber creído en mí sin apenas conocerme, esta ha sido una oportunidad definitiva en mi vida, a nivel profesional y personal. También quiero agradecer el apoyo que me ofrecieron cuando decidí el tema sobre cual desarrollaría mi trabajo de tesis.

A mi Director de Tesis, el Doctor Manuel Artís, quién siempre estuvo dispuesto a colaborarme en lo que pudiéra necesitar para hacer realidad esta tesis.

A mi tutor, doctor Raúl Ramos Lobo. Raúl, tu mereces párrafo aparte, gracias y mil gracias por todo el tiempo que me has dedicado. Gracias por ser uno de los que más se ha involucrado en mi proceso de formación como investigador. Y, sobretodo, gracias por la inmensa confianza que has depositado en mí desde mi primer día de trabajo.

Gracias a todos y cada uno de los integrantes de AQR, por todo lo que me han enseñado y por el afecto que me han transmitido. Gracias por sus críticas constructivas y por todos esos momentos en los que he podido contar con ustedes. Gracias a Ernest, Esther, Javi, Quique, Rosina y Vicente por todos sus comentarios, asesorías, consejos y apoyo que me han ofrecido a lo largo de la elaboración de la tesis.

A mis anteriores compañeros de despacho, Mónica, Juan Luis, Isabel y Jorge Iván, por haberme soportado tanto tiempo, por ser mis amigos, por darme esos empujones morales que se necesitan para poder alcanzar cada objetivo y por respetar esos momentos de concentración en los cuales me sumergía, de vez en cuando. También, gracias a mi actual compañero de despacho, Jordi, por darle esa chispa de alegría a cada día de trabajo.

Gracias a mi familia por apoyarme en la decisión de dejar mi trabajo en Colombia para ir detrás de un sueño al otro lado del océano. Toda esta experiencia nos ha unido más que nunca y nos ha dado la oportunidad de decirnos constantemente lo mucho que nos queremos.

Finalmente, para cerrar con broche de oro, quiero expresar mis más profundos agradecimientos a la persona que me ha apoyado durante estos años de arduo trabajo, mi novia Lolita. Preciosa, gracias por ser mi amiga y cómplice en todos mis proyectos, por creer en mi e interesarte por mi trabajo. Y sobretodo, gracias por enseñarme con infinita paciencia que fuera del trabajo hay otra vida que debemos cuidar y cultivar con mucho esmero. Le doy gracias a Dios por regalarme tu compañía y también le pido que me dé lo que necesite para saberla conservar.

*CONTENTS*

## *TABLES*

## *FIGURES*

# *CHAPTER 1*

## *Introduction and objectives*

The interest in geographical information technologies has grown considerably over the last three decades. Today, geographical information is no longer the exclusive domain of government and public administrations (in the areas of planning, demography and topography), thanks to the development of computer tools which have enabled firms and academic institutions to use this information.

Statistical information of this kind is usually published at a variety of territorial levels in order to provide information of interest to all potential users. When using this information, researchers have two alternatives for defining the basic territorial units for use in the study: first, they may use geographical units designed following normative criteria (that is, officially established territorial units such as towns or provinces) or, second, they can apply analytical criteria in order to design geographical units directly related to the phenomena under examination.

*"Normative regions are the expression of a political will; their limits are fixed according to the tasks allocated to the territorial communities, to the sizes of population necessary to carry out these tasks efficiently and economically, or according to historical, cultural and other factors. Whereas analytical (or functional) regions are defined according to analytical requirements: functional regions are formed by zones grouped together using geographical criteria (e.g., altitude or type of soil) or/and using socio-economic criteria (e.g., homogeneity, complementarity or polarity of regional economies)"* (Eurostat, 2004).

Most empirical studies use geographical units based on normative criteria, for several reasons: these units are officially established, they have traditionally been used in other studies, their use makes comparison of results easier and there is less scope for criticism. However, studies using units of this type may have an "Achilles' heel": they may be very restrictive, or unsuited the problem considered. For example, if we are analysing phenomena such as the regional effects of monetary and fiscal policy, how

will the results be affected if the aggregated areas[1] in each region are heterogeneous? Can these results change if the areas are redefined in such a way that each region contains similar areas?

This situation could be improved by the use of regionalisation processes to design geographical units based on analytical criteria by aggregating small geographical units[2], but without reaching the upper level, or alternatively by combining information obtained from different levels[3]. In this context, the design of analytical geographical units should consider three fundamental aspects:

a. *Geographical contiguity*: The aggregation of areas into regions such that the areas assigned to a region are internally connected or contiguous.

b. *Equality*: In some cases, it is important that the regions designed are "equal" in terms of a particular variable (for example population, size, presence of infrastructures, etc).

---

[1] In this thesis dissertation, the term "area" will be used 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.

[2] Apart from aspects such as statistical secrets or other legislation on 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 inaccuracies in the data, and to simplify information requirements for calculations and to facilitate its visualisation and interpretations in maps.

[3] See, for example, Albert *et al.* (2003), who analyse the spatial distribution of economic activity using information with different levels of regional aggregation, NUTS III for Spain and France and NUTS II for other countries, with the objective of "using similar territorial units". López-Bazo *et al.* (1999) analyse inequalities and regional convergence 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 data for the United Kingdom, Ireland and Luxembourg so as to ensure the comparability of geographical units.

c. *Interaction between areas*: Some variables do not exactly define geographical characteristics that can be used to aggregate the different areas, but may describe some kind of interaction between them (for example, distance, time, number or trips between areas, etc). These variables can also be used as interaction variables using a dissimilarity measure between areas in terms of socio-economic characteristics. The objective in this kind of regionalisation process is to make the areas belonging to the same region as homogeneous as possible with regard to the attribute(s) specified.

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

The main objective in this thesis dissertation is to implement a new automated regionalisation tool to design homogeneous geographical units directly related to the phenomena analysed which overcomes some of the disadvantages of the methodologies currently available.

Thus, the specific objectives are:

a. To formulate the regionalisation problem as a linear optimisation model able to take into account not only the areal characteristics but also their non-metric and contiguity relationships.

b. To propose a heuristic model able to solve bigger regionalisation problems, incorporating in its search procedure the characteristics of a regionalisation process.

c. To compare the homogeneity of the analytical regions designed by applying the regionalisation model proposed in this thesis with those obtained using another regionalisation method based on normative criteria. For this comparison, provincial time series of unemployment rates in Spain will be used.

This dissertation is organised as follows. Chapter 2 briefly summarises the literature on different regionalisation methods. Special emphasis will be placed on those methodologies which have made the greatest impact on the specialist literature and on those that have been tested satisfactorily in real problems.

In chapter 3 the regionalisation problem is formulated as a linear optimisation model in which the problem of obtaining $r$ homogeneous regions is based on the minimisation of the total heterogeneity, measured as the sum of the dissimilarity relationships between areas belonging to the same region.The proposed model has the following characteristics:

a. It is an automated regionalisation model that is able to design a given number of homogeneous geographical units from aggregated small areas subject to contiguity requirements.

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

c. By formulating the regionalisation problem as a linear optimisation problem, we have the chance of finding the global optimum from among all feasible solutions.

d. More consistent solutions can be easily obtained by introducing additional constraints taking into account other specific requirements that are relevant for the regionalisation process.

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

f. With this model a region consists of two or more contiguous areas; this implies that no region can be formed by a single area[4].

In order to apply this model in larger-scale regionalisation processes, chapter 4 presents an algorithm called the *RASS* (Regionalisation Algorithm with Selective Search). The key advantage of this new algorithm is that the way it operates is based on the features of regionalisation processes themselves, where available information about the relationships between areas can play a crucial role in directing the search process more selective, more efficient and less random fashion. In fact, the *RASS* incorporates inside the linear optimisation model presented in chapter 3 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 adapted iterative methods.

[4] As Crone (2003) highlights, this is one of the conditions followed by the Bureau of Economic Analysis (BEA) for the regionalisation of the United States of America.

In chapter 5 provincial time series of unemployment rates in Spain are used to compare the results obtained by applying two analytical regionalisation models, each one representing a different regionalisation strategy: a two-stage procedure based on cluster analysis and the *RASS* algorithm. The results will also be compared with normative regions available at two different scales: NUTS II and NUTS I.

Lastly, in chapter 6 we present the most important conclusions and make proposals for further research lines.

# *CHAPTER 2*

## *Regionalisation procedures: literature review*

## 2.1. Introduction.

In this chapter the most frequently used methodologies for territorial aggregation will be briefly summarised. The summary will focus on methodologies with a higher impact in the specialised literature and on those that have been tested satisfactorily in real problems.

Most of these methodologies use techniques based on cluster analysis[5]. In this context, the problem of aggregation of spatial data is considered as a particular case of clustering in which the geographical contiguity between the elements to be grouped should be considered. This particular case of clustering methods is usually known as contiguity-constrained clustering or simply the 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).

In this context, regionalisation algorithms can be categorized under three methodological strategies: two-stage 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.

## 2.2. Two stages aggregation.

In this strategy, the first stage consists of applying a conventional clustering model without taking into account the contiguity constraint. In the second stage, the clusters are revised in terms of geographical contiguity. With this methodology, if the areas

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

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.

### 2.2.1. Hierarchical algorithms.

These algorithms are usually applied when the researcher is interested in obtaining a hierarchical and nested classification (for every scale level), that is usually summarised using dendograms[6]. The main disadvantage of using hierarchical clustering algorithms, apart from the high computational requirements (Wise *et al.*, 1997), is the high probability of obtaining a local optimum due to the fact that once two elements have been grouped in a particular aggregation level, they will not be re-evaluated independently at higher levels (Semple and Green, 1984). On the other hand, their main advantage is that there is no need to specify initial partitions to apply the algorithm (Macmillan and Pierce, 1994).

### 2.2.2. Partitioning algorithms.

The K-means procedure, which belongs to the partitioning clustering category, is more frequently used in regionalisation processes. This iterative technique consists of selecting from the elements to be grouped a predetermined number of $k$ elements that will act as centroids (as many as there are groups to be formed). Each of the other elements is then assigned to the closest centroid.

---

[6] Graphic representations of the solutions of hierarchical cluster (Gordon, 1996).

The aggregation process is based on minimising a 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[7]: see equation 2.1.

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

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

The K-means algorithm is based on an iterative process in which initial centroids are explicitly or randomly assigned and the other elements are assigned to the nearest centroid. After this initial assignation, the initial centroids are reassigned in order to minimise the squared euclidean distance. The iterative process is terminated if there is no 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 (that is, the designation of the initial centroids). This makes it quite difficult to obtain a global optimum solution.

Finally, when a K-means algorithm is applied in a two-stage regionalisation process, it may be the case that the number of regions to be designed will not necessarily be equal to the value given to parameter $k$, since areas belonging to the same cluster have to be counted as different regions if they are not contiguous. So, different proofs have to be made with different values of $k$ (lower than the number of regions desired), until

[7] 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).

contiguous regions are obtained. In some cases it may in fact be impossible to obtain the desired number of contiguous regions.

Among the advantages of the two-stage aggregation methodology, Openshaw and Wymer (1995) highlight the fact that the homogeneity of the regions defined is guaranteed by the first stage. Moreover, this methodology may also help to obtain evidence of spatial dependence between 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[8] and not on the researcher may present a substantial problem.

## 2.3. Inclusion of geographical information as classification variables.

The second strategy includes 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 achieve geographical contiguity, the geographical coordinates are included in the calculation of dissimilarities between areas and conventional classification algorithms are then applied.

An approach of this kind was 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 intra-group variance taking into account the non-spatial attributes; the second, the geographical component, includes the sum of the distances from areal centroids to the cluster centroids in order to achieve geographical contiguity, and the third component is a measure of the deviation between the regional value of an attribute and its average value. A different weight is assigned to each of these components in the objective

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

function in order to obtain a unique value to minimise. The regionalisation procedure is based on K-means a partitioning algorithm (Andemberg, 1973).

Calciu (1996) uses the same territorial aggregation strategy, defining it as "*contrainte spatiale implicite*" (implicit spatial constraint), which incorporates as geographical variables the Cartesian coordinates, appropriately transformed, of the points representing each area. This author favours applying a hierarchical classification algorithm, where the inclusion of the coordinates makes it possible to obtain an improved geographical continuity, although it also involves a certain degree of loss in terms of intragroup homogeneity compared with the case where the hierarchical algorithm is applied without considering these geographical variables.

The main drawback associated with this methodology is the difficulty of simultaneously treating variables expressed in different measurement units and the definition of objective weights for each of the variables, especially the geographical variables 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 may change depending on the method used to localise the centroid representing each of the areas to be grouped, especially in cases in which the areas are particularly large (Horn, 1995, Martin *et al.*, 2001).

## 2.4. Additional instruments to control for the continuity restriction.

The last and perhaps the commonest strategy for solving territorial aggregation problems is to control for the geographical contiguity constraint using additional instruments such as the contact matrix or its corresponding contiguity graph. The 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[9].

These elements are used to adapt conventional clustering algorithms, either hierarchical or partitioning, in order to respect the continuity constraint.

The main problem with adapted hierarchical algorithms in the context of regionalisation processes is that there may be breaks in monotonicity between elements. This problem is known as *reversal*, that is, the distance between two objects may be higher than the distance between the union of this object with a third one (Calciu, 1996, Gordon 1996, Ferligoj and Batagelj, 1982). *Reversals* complicate the task of interpreting classification.

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

As far as mathematical programming is concerned, Macmillan and Pierce (1994) define the regionalisation problem as an optimisation problem in which, given a predetermined number of groups to form, the solution will define the optimum territorial aggregation. The solution proposed by these authors to ensure that geographical continuity consists of exponentiating the contact matrix, bearing in mind that for the formation of a region with $n$ continuous areas the $(n-1)^{th}$ power of the contact matrix may 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 become trapped in a local optimal solution.

---

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

Cutting algorithms for graph partitioning is another way to see the regionalisation problem from the perspective of mathematical programming. In these models, the contiguity graph stores in each arc $(i,j)$ a value of dissimilarity between areas $i$ and $j$, i.e. $G=(V,E)$, with a weight function $w : E \rightarrow N$.

Cutting algorithm looks for a partition of the node set $V$ into $k$ disjoint sets $F=\{C_1, C_2,..., C_k\}$ where $k$ is an integer and $k \in [2..|V]$. Thus, in a regionalisation process, the idea would be to maximise the isolation between groups, so the objective in a "maximum k-cut" is to maximise 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[10].

Another method, cited by Neves *et al.* (2001), consists of the reduction of the contiguity graph ($G=(V,E)$) with $w : E \rightarrow N$). The reduction progressively eliminates the 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 that model the dissimilarity relationships by using the arcs of the contiguity graph is that they do not consider a large number of dissimilarity relationships between areas that are not contiguous.

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

Since the resolution of problems of this kind using conventional optimisation methods is extremely complex[11], other methodologies have been developed in the field of regionalisation which have proved highly effective in those cases where a large number of elements are to be grouped. Among these solutions, the algorithms known as *Iterative Relocation Algorithms* have been widely analysed. These methods try to find the best regional configuration using a non-optimal configuration as a starting point[12] and then performing different movements of areas between regions in order to improve the objective function. Ferligoj and Batagelj (1982) provide different iterative reallocation algorithms that allow movement of an area to a different region only if contiguity constraints 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 in the context of the particular case of splitting a country into administrative areas or electoral districts so that the final regionalisation minimises the effects of the Modifiable Areal Unit Problem (MAUP)[13].

*Iterative Relocation Algorithms* have been improved by using heuristics that permit an improved search among the different feasible solutions and avoid the risk of being trapped in a local optimum. The heuristics most used in this context are the *Simulated*

---

[11] 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).

[12] Different alternatives for determining the initial solution can be found in Wise et al. (1997).

[13] Openshaw defined the Modifiable Areal Unit Problem (MAUP) as a potential source of error that can affect the results of studies based on geographical aggregated information, as these results could vary depending on the configuration of this aggregation. The MAUP is related to two different problems regarding the analysis of spatial data: the problem of scale, related to the desired number of regions, and the problem of aggregation, related to 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).

*Annealing* (AZP-SA) and the *Tabu Search Algorithm*[14,15] (AZP-TABU), proposed by Openshaw and Rao (1995), and the *Anneal Redistricting Algorithm* proposed by Macmillan and Pierce (1994).

As algorithms of this kind are of special interest in this thesis dissertation, in the next section we present a brief description of the heuristics with the greatest impact in the field of regionalisation, and which fulfil the following conditions: their objective is to divide a territory into 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 or those that try to find core areas or centroids in order to assign the rest of the areas.

### 2.4.1. Automatic Zoning Procedure (AZP).

This heuristic proposed by Openshaw (1977) is based on an iterative procedure. It consists of 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 has 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.

---

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

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

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 is the possibility of using any objective function that is sensitive to the aggregation of zones. This characteristic is extremely useful for approximating the limit of the aggregation effects. It also served to demonstrate that MAUP exists.

Its main disadvantages are related to the local search procedure (restricted to the selected region) and to the strong dependence of the results on the starting point selected (step 1). Equally, the strategy of ignoring the possibility of moving a zone that implies a decrease in the objective function may cause the heuristic to be trapped in a local optimum. Openshaw tried to solve this problem in later proposals.

### 2.4.2. Simulated Annealing Variant of AZP (AZP-SA).

This proposal of Openshaw and Rao (1995) incorporates a modification to the AZP: step 5 now consists of "*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*":

$$R(0,1) < \exp\left(\frac{\nabla f}{T(k)}\right) \qquad (2\cdot3)$$

| | |
|---|---|
| $\nabla f$ | 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 lies of the possibility of moving towards a 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: $T(k)=fT(K\text{-}1)$ 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 $k$: $T(k)=0.85 \cdot T(k\text{-}1)$ 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 the *ANNEAL redistricting problem*, aims to group $C$ counties into $D$ 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).

Given that much of the complexity of the proposed heuristics for regionalisation problems is due to the control of contiguity, Macmillan (2001) proposes a regionalisation algorithm called SARA, which incorporates a more efficient methodology to control contiguity constraints based on the concept of *switching points*. This new proposal significantly improves the execution times obtained by Openshaw and Rao (1995).

### 2.4.3. 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.

The main advantage of this algorithm is that it permits us to escape from local optimums or cyclical behaviour. Its disadvantage lies in 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.

### 2.4.4. 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) in order to build 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 on the simplification of $G$, such that $G$ is replaced by one of its spanning tree $T$ in which each pair of vertices are connected by one and only one group of arcs. The group of arcs belonging to $T$ is 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 that are intraconnected but not interconnected. This result is consistent with the regionalisation requirements.

The MIDAS heuristic can be summarised in the following steps[16]:

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 $G$, $\Pi_p(T)$;

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

$$\min\{f(\pi): \pi \in \Pi_p(T)\} \qquad (2\cdot4)$$

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

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

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

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

$$\pi \in \Pi_p(\overline{T}) \qquad \text{(iii)}$$

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

The main drawback with this methodology is the loss of control on the number of elements included in each partition, and, more importantly, the use of arcs of the contact matrix $G$ as a way to represent the relationships between vertices, since this implies ignoring other relationships between non-adjacent vertices.

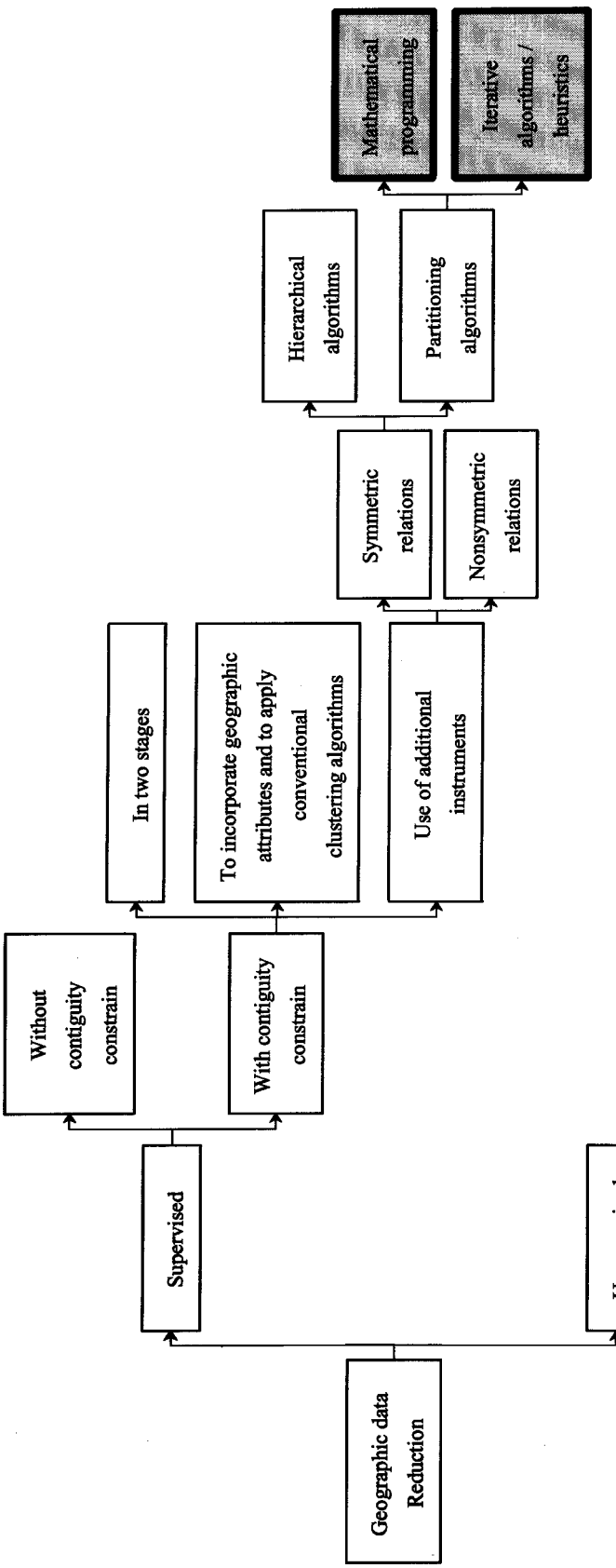
The methodologies of constrained clustering in which additional instruments are included have as a common characteristic that the relationships between the areas to be grouped are symmetrical. Ferligoj and Batagelj (1983) have developed agglomerative algorithms where asymmetric relationships can be considered.

All the methods presented above are "supervised" models, which means that the researcher knows *a priori* the data structure of the phenomenon analysed. But there are other unsupervised models that can be useful when the researcher wants to analyse a large amount of data and there is not enough information on the factors that can affect the system. In these cases, one possibility is to apply a non-parametric analysis of data that will identify patterns and relationships among the elements under consideration. Among the best known applications of these methods in the field of regionalisation are *Self Organization Maps* (SOM) proposed by Kohonen (1984). There is no consensus

among researchers on the validity of this methodology, originally developed in the field of artificial intelligence, due to the lack of a theoretical basis that complicates the interpretation of the results (Openshaw, 1992).

A summary of the methodologies described in this chapter can be found in table 2.1.

**Table 2.1. Summary of the methodologies available for the reduction of geographical data.**




Source: Own elaboration.

# *CHAPTER 3*

## *A linear optimisation model for the design of homogeneous territorial units*

## 3.1. Introduction.

In this chapter, the regionalisation problem is formulated as a linear optimisation model that allows the design of regions taking into account not only the characteristics of the areas but also their relationships. The possibility of treating the regionalisation problem as a linear model implies that, by its mathematical properties, the feasible region is convex and, as a result, it is possible to find the optimal solution. Other advantages of this formulation are that it is easy and relatively cheap to implement in a great variety of commercial software, and is flexible when some changes or additional constraints are needed.

Before introducing the mathematical formalisation of the model, we should mention its main characteristics and assumptions.

## 3.2. Model description.

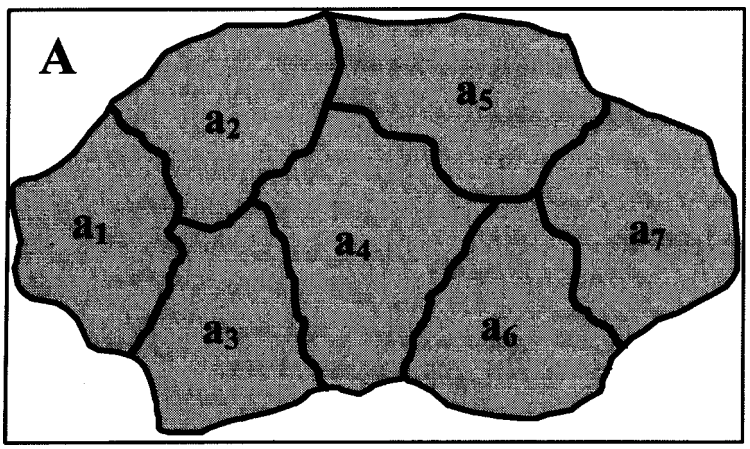
### 3.2.1. Representation of the geographical set.

The starting point of any regionalisation process is the identification of the territory to regionalise. Figure 3.1 shows an example of a territory comprising a finite number ($n$) of geographical areas of smaller size forming a geographical contiguous region $\mathbf{A} = \{a_1, a_2, a_3, \ldots, a_n\}$.

Once the territory of interest has been defined, the next step is to simplify the previously defined geographical set so that each of the elements considered ($n$ areas) and their neighbourhood relationships can be easily represented. This simplification can be done using a graph formed by $n$ nodes, each of them representing one of the areas considered, and arcs that represent the geographical contiguity between them.

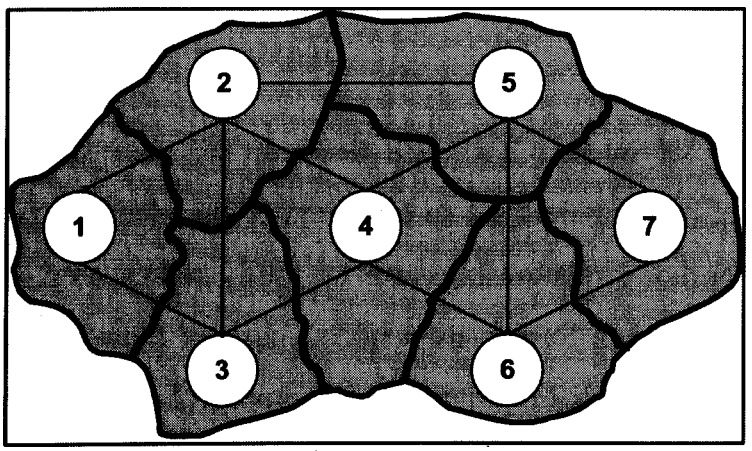
Several methods are available for simplification of this kind. We selected the most general one, *Delaunay Triangulation* (DT) (Aurenhammer, 1991). With this method, each arc relates areas with a common border. One of the main advantages of this method is that the location of the point representing each 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 may differ depending on the location of the areal centroids. Figure 3.2 shows the DT graph of the territory considered in the example.

**Figure 3.1. Group of areas that form the territory to regionalise.**




Source: Own elaboration.

**Figure 3.2. Delaunay Triangulation (DT).**




Source: Own elaboration.

### 3.2.2. Relationships between the elements to be grouped.

The next step consists of the consideration of the relationships between areas (or nodes of the graph). The consideration of these relationships is one of the most important stages in the regionalisation process proposed in this chapter, since taking into account interactions between areas could be very useful in a wide range of applications. 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 also to consider 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 pair of areas $i, j$.

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

$$d_{ij} = d_{ji} \qquad \forall i, \forall j = 1,...,n \tag{3·1}$$

$$d_{ij} \geq 0, \ (d_{ij} = 0 \text{ if } i = j) \qquad \forall i, \forall j = 1,...,n \tag{3·2}$$

These properties imply that the function need not be metric (it does not have to satisfy the triangular inequality[17]):

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

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

The possibility of using distance functions that need not be metric can be understood as a relaxation of the hypothesis used in the regionalisation models based on 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 constraint.

### 3.2.3. Strategy for the configuration of regions.

Once we have information about the territorial configuration and the relationships between the different areas, the next step is to group the $n$ areas $\{a_1, a_2, \ldots, a_n\}$ into $m$ non-empty sets or regions $\{1, 2, \ldots, m\}$ in a way that the areas belonging to each region are geographically contiguous.

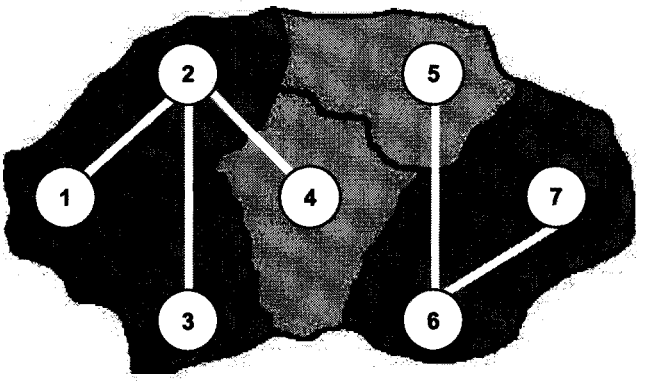
To define these regions we need to select $n$-$m$ arcs from the global set of arcs that define the contiguity graph. These $n$-$m$ arcs can be understood as a necessary but not sufficient condition for the formation of $m$ regions in a way that areas belonging to each region are totally interconnected 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 pair of areas should be connected by one and only one combination of arcs[18]. This kind of regional configurations implies that the minimum number of areas in each region will be two (one arc connecting two areas), this is $m = [n/2]$. This condition is less restrictive as the number of areas forming the territory increases[19]. Figure 3.3 shows a possible solution for designing 2 regions from 7 areas.

---

[18] For more information about the properties of this (and other) configurations, see Ahuja *et al.* (1993).

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

**Figure 3.3. Feasible result for the design of two regions.**




Source: Own elaboration.

The location of arcs in each region has no 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 configured with arcs 1-3, 2-4 and 3-4. This equivalence is due with the fact that the arcs function is only to ensure geographical contiguity, because they have no value assigned to them. This strategy can be very useful to identify regional configurations with a high variety of shapes (elongated or compact regions), as it does not rely on centroids, which tend to produce compact areas.

### 3.2.4. Criteria considered 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, geographically contiguous unit. So a partition criterion establishing which of the possible configurations of $n$ areas in $m$ regions is the best suited should be defined.

With this aim, it is necessary to define a measure of adequacy of a regional configuration. One possibility is to calculate the degree of heterogeneity of the areas assigned to a region; another is to calculate the degree of isolation of the areas of one region related to the rest. The heterogeneity measure selected in this dissertation

consists of 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:

$$H(C_r) \equiv \sum_{\{i,j \in C_r | i<j\}} d_{ij} \qquad (3\cdot4)$$

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

$$P(H,\Sigma) \equiv \sum_{r=1}^{c} H(C_r) \qquad (3\cdot5)$$

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

$$P(H,Max) \equiv \max_{\{r=1,\ldots,c\}} H(C_r) \qquad (3\cdot6)$$

One disadvantage of 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, precluding the chance to make changes that could improve their heterogeneity. For this reason, the selected strategy has been 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:

$$P(H,\Sigma) \equiv P(I,\Sigma) \equiv \sum_{r=1}^{c} I(C_r) \text{ with } I(C_r) \equiv \sum_{i \in C_r} \sum_{j \notin C_r} d_{ij} \quad (3\cdot7)$$

### 3.3. 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 :

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

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

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

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

Subject to:

$$T_{ij} \geq Y_{ik} + Y_{jk} - 1, \qquad \forall i, \forall j = 1,...,n \; ; \; \forall k = 1,...,m \qquad (3\cdot8)$$

$$\sum_{i=1}^{n} Y_{ik} \geq 2, \qquad \forall k = 1,...,m \qquad (3\cdot9)$$

$$\sum_{k=1}^{m} Y_{ik} = 1, \qquad \forall i = 1,...,n \qquad (3\cdot10)$$

$$\sum_{j \in N_i} X_{ijk} \leq Y_{ik} \cdot M, \qquad \forall i = 1,...,n \; ; \; \forall k = 1,...,m \qquad (3\cdot11)$$

$$\sum_{j \in N_i} X_{jik} \leq Y_{ik} \cdot M, \qquad \forall i = 1,...,n \; ; \; \forall k = 1,...,m \qquad (3\cdot12)$$

$$\sum_{i=1}^{n} \sum_{j \in N_i} X_{ijk} = \sum_{i=1}^{n} Y_{ik} - 1, \qquad \forall k = 1,...,m \qquad (3\cdot13)$$

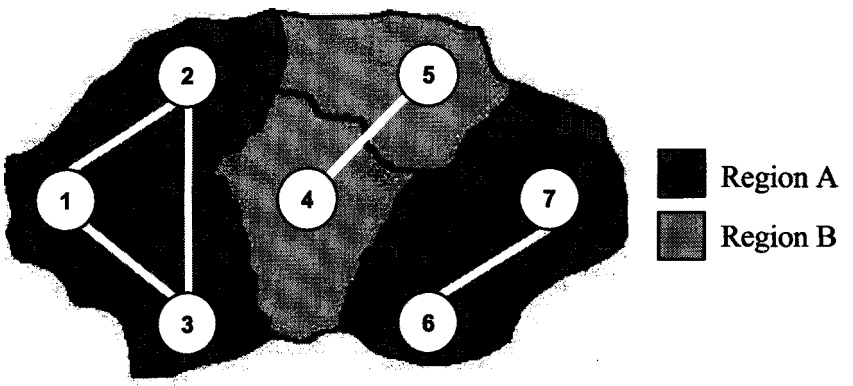
$$\sum_{i,j \in C} X_{ijk} \leq |C| - 1, \quad \forall \text{ non - empty subset of } C \subseteq \{3,...,(n\text{-}2m+1)\}; \quad \forall k = 1,...,m \qquad (3\cdot14)$$

$$X_{ijk} \in \{1,0\} \; ; \; Y_{ik} \in \{1,0\}; \; T_{ij} \geq 0, \quad \forall i, \forall j = 1,...,n \; ; \; \forall k = 1,...,m \qquad (3\cdot15)$$

As previously mentioned, the objective function seeks the minimisation of 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 (3·8) 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 (3·9) imposes the requirement that the minimum number of areas defining a region is two. As previously mentioned, the restriction is less strong as the number of areas increases. Restriction (3·10) imposes the condition that each area must be

assigned to one and only one region. Restrictions (3·11) and (3·12) require 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 greater than one. Restriction (3·13) imposes the condition 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 3.4, where region *A*, formed by areas 1, 2, 3, 6 and 7, satisfies restriction (3·13) – 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 some of the arcs, if there are cycles and this is the origin of restriction (3·14).

**Figure 3.4. Non-feasible regional configuration.**




Source: Own elaboration.

The problem of cycles has been treated in the literature as the analysis of *subtours* in transport models such as the *Vehicle Routing Problem* (VRP)[20]. The VRP consists of defining vehicle routes with a given origin and end in the same node (called the *depot*) and trying to minimize costs. The design of a tour for a certain vehicle cannot contain

[20] 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).

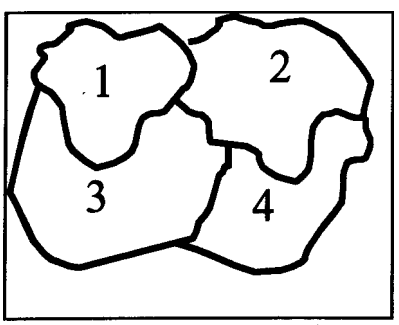
subtours and to control for this condition, the VRP incorporates the following constraint:

$$\sum_{j,i\in S} X_{ijk} \leq |S| - 1, \ \forall \text{ non-empty subset of } S \subseteq \{2,...,n\};\ k=1,...,m. \qquad (3\cdot16)$$

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 size), but not for the regionalisation problem. For example, a *depot* node must be established *a priori* as 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 3.5 we can clearly identify the different combination of arcs $c_{ij}$ that can generate cycles. The combination of arcs 1-2, 1-3, 2-3 (or 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 between the four areas.

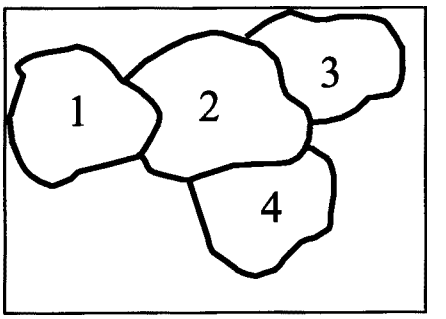
**Figure 3.5. Configuration of areas with potential cycles.**




Source: Own elaboration.

Moreover, in a territorial configuration such as the one shown in Figure 3.6, there is no combination of arcs $c_{ij}$ 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 $c_{ij}$ is limited to those combinations $i,j$ where the value of the contact matrix $wij$ =1. This is the set of potential arcs $c_{ij}$ that are included in $N_i$.

**Figure 3.6. 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? The answer is yes: we only have to identify the 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 3.5, 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 which 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, what is the maximum number of areas that can be involved in a cycle? As the model, in restriction (3·9), 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 have only one possible arc (restriction 3·13). 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:

$$n\text{-}2m+1 \qquad (3\cdot17)$$

So, the minimum number of areas where the possibility of finding a cycle should be evaluated is three, as this problem is impossible for a lower number of areas. As a result, restriction (3·14) is related to the modification of the set $S$ as proposed in the VRP, which in the context of our problem was stated as $C$. Using this modification, we achieve a substantial reduction in the number of restrictions to satisfy, avoiding the exponential increase in the number of restrictions with $n$ and $m$. This allows us to use commercial software in the context of regionalisation problems with a high number of areas and regions.

Last, restriction (3·15) 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 (3·8) and the objective of minimisation of the model[21].

### 3.4. Application of the model.

In this subsection, several examples are shown with the aim of illustrating the model's capacity to design regional configurations with different characteristics. A first set of four examples is used, each one with 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

[21] 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 sub-problems is drastically reduced. For more information about this algorithm, see Hiriart *et al.* (1983).

optimum regional configurations. The procedure for obtaining the dissimilarity matrix in each example is as follows:

1. $n$ areas have been grouped in $m$ contiguous regions, assigning each area $i = \{1,...,n\}$ to a region $k = \{1,...,m\}$. This aggregation makes it possible to build the set $R_k$ $\{i/i \in k\}$.

2. A value is assigned to each of the areas $i = \{1,...,n\}$ depending on their region. This value is given by the sum of a constant with a random term, generated from a uniform distribution between 0 and 1. The value of the constant is different for each region, as there should be a large enough difference ($D$) to obtain significant different average values for each region. The expression applied is:

$$A_{i \in R_k} = C + (D * k) + \varepsilon \quad \forall i = 1,...,n; \forall k = 1,...,m; \varepsilon \sim U[0,1] \qquad (3\cdot18)$$

3. Next, the relationships between areas are calculated using a distance function. The weighted euclidean distance is applied in order to calculate distances between the elements of the $A_i$ vector after centering it.

$$d_{ij} = \sqrt{\left(\frac{A_i^c}{S} - \frac{A_j^c}{S}\right)^2}, \qquad \forall i, j = 1,...,n \mid i < j \qquad (3\cdot19)$$

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

$$A_i^c = A_i - \left(\sum_{i=1}^{n} A_i \Big/ n\right), \qquad \forall i = 1,...,n \qquad (3\cdot20)$$

The matrixes obtained with this procedure are shown in Table 3.1.

**Table 3.1. 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 |

*continue...*

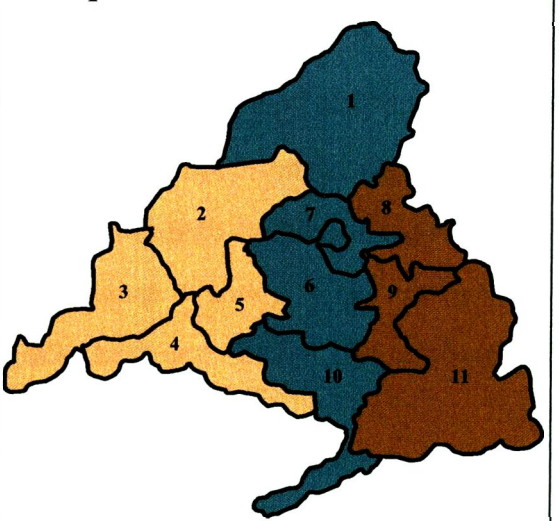
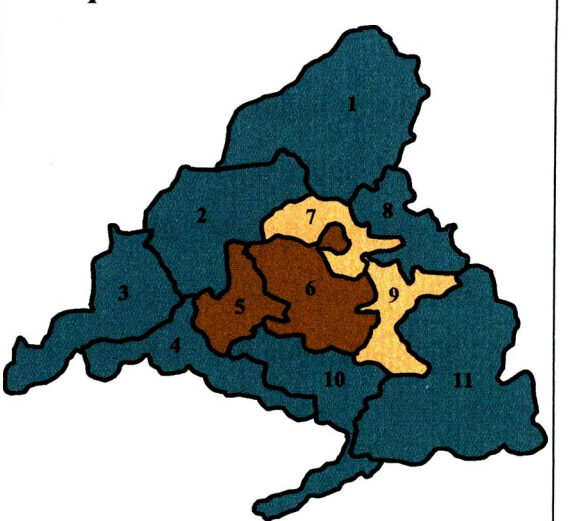
***Example 4***

| 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 |

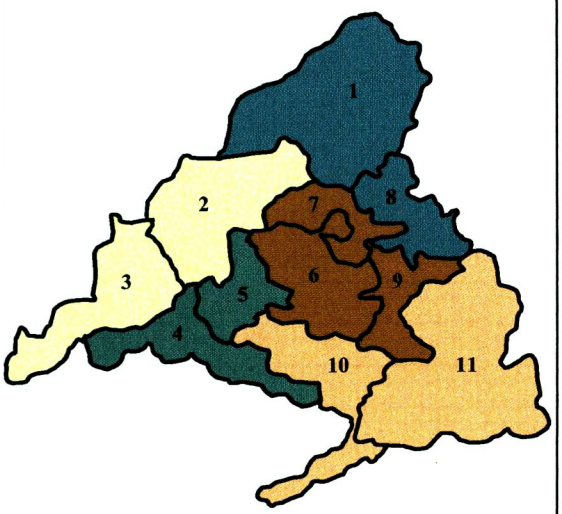
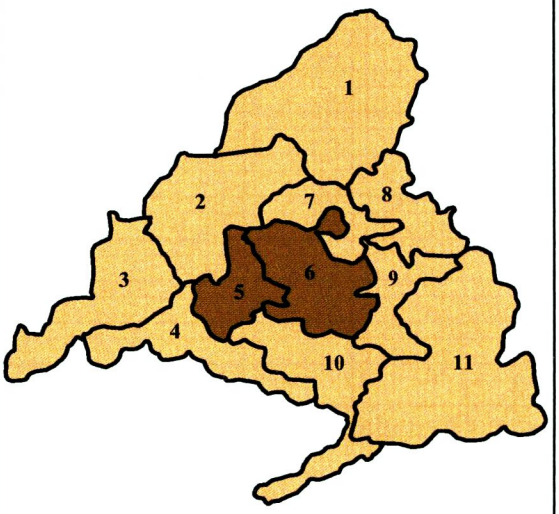
Source: Own elaboration.

The regional configurations obtained after applying the optimisation model with the different relationship matrices are shown in the maps in Table 3.2. The solutions coincide with the optimal regional configurations predefined above and so the model seems able to design regions with a high variety of shapes.

**Table 3.2. Solutions for the relationships matrixes from Table 3.1.**

| ***Example 1*** | ***Example 2*** |
|---|---|
|  |  |
| ***n=11 and m=3*** | ***n=11 and m=3*** |

*continue...*

| *Example 3* | *Example 4* |
|---|---|
|  |  |
| ***n=11 and m=5*** | ***n=11 and m=2*** |

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

### 3.5. 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 section, a relationship matrix ($D_{i,j}$) has been calculated from demographic data[22] for the 11 statistical areas in which the Comunidad de Madrid is divided at the NUTS IV level (see Table 3.3).

[22] 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 3.3. 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: Padrón continuo 1999. Instituto de Estadística de la Comunidad de Madrid.[23]

To combine the information of the three variables (replacing ratio, dependence ratio and progressivity ratio) ($v$ = 1, 2, 3) in the relationship matrix, the following distance function is used:

$$d_{ij} = \sqrt{\sum_{v=1}^{p}\left(\frac{A_{iv}^{c}}{S_v} - \frac{A_{jv}^{c}}{S_v}\right)^2}, \quad \forall i, \forall j = 1,...,n \,|\, i < j \; ; \; \forall v = 1,...,p \qquad (3\cdot21)$$

Expression (3·21) is a multivariate version of (3·19) which 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, from 1 to $p$, of the squared distances between $i$ and $j$. The relationship matrix obtained is shown in Table 3.4.

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

**Table 3.4. Relationships matrix from demographic variables in Table 3.3.**

| 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.

### 3.5.1. Requirement of a population minimum.

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

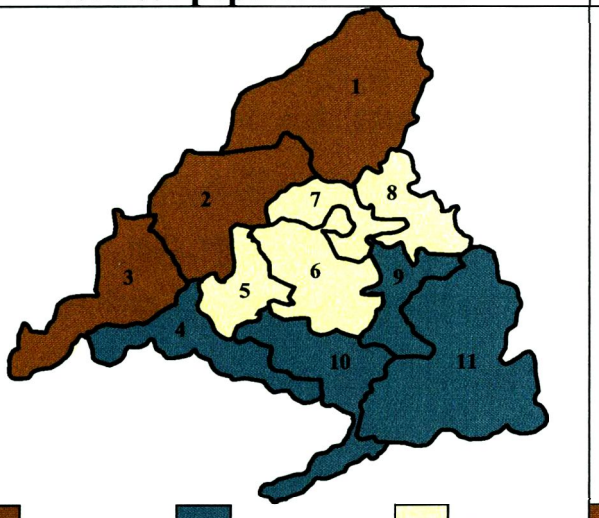
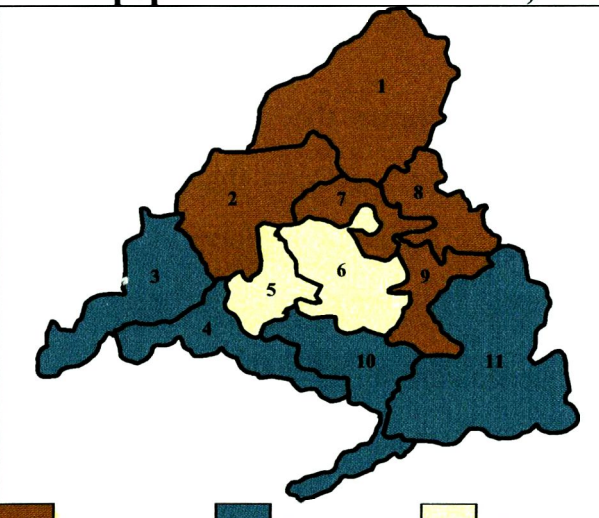
$$\sum_{i=1}^{n} Y_{ik} \cdot P_i \geq L, \quad \forall k = 1,...,m \qquad (3\cdot22)$$

where $P_i$ is a vector containing information on the population of each of the areas considered and $L$ is a constant that specifies the population minimum required. For this example, this minimum has been set at 800,000 inhabitants.

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

The results obtained after solving a model with and without a minimum population requirement are shown in Table 3.5. In the solution for the unrestricted model (left map), region 1 has the lowest value for population, 131,080 inhabitants, while in the solution for the restricted model (right map), the same region still has the lowest population value, but it rises to 820,186 inhabitants (>800,000 inhabitants).

**Table 3.5. 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.

### 3.5.2. Configuration of regions with mandatory isolation.

A different kind of restriction that is possibly of interest is to require 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:

$$\sum_{i=1}^{n} Y_{ik} \cdot O_i = 1, \qquad \forall k = 1,\ldots,m \tag{3·23}$$

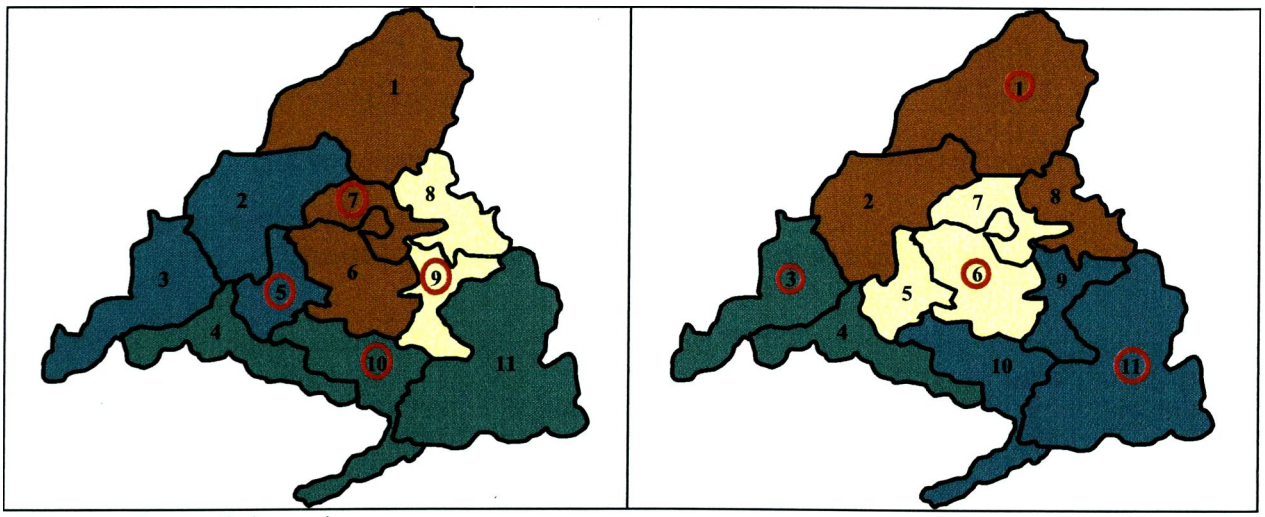
where $O_i$ is a binary vector that takes 1 for selected areas and 0 for the rest. Note that, when defining $O_i$, the number of selected areas must be equal to the desired number of regions:

$$\sum_{i=1}^{n} O_i = m \qquad (3\cdot24)$$

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

Table 3.6 shows the results of applying the model with data from Table 3.4, but imposing the condition 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 3.6. Solutions with mandatory isolation.**




Source: Own elaboration.

## 3.6. Computational results.

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

With the aim of testing the computational capacity of the model, we applied it to different random territorial configurations. The procedure used 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 binary matrix obtained 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 matrices have an average density of 28.3% and a median of 3 neighbourhoods per area, ranging from 1 to 8.

c. Every randomly generated matrix was evaluated in terms of geographical contiguity, and the feasible ones selected[24].

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

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

Table 3.7 shows the average running times[25] for different combinations of areas and regions (5 examples for each combination).

**Table 3.7. 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.

Although the number of restrictions was clearly reduced with the modification of restriction (3·14) controlling the elimination of cycles, the running time is still very high. Indeed, for cases with more than 17 areas the running time increases substantially. For this reason, another alternative that would increase the computational capacity of the model will be considered in the next chapter.

[25] The calculations in this paper are 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.

# *CHAPTER 4*

## *A solution for the "computational problem": The RASS algorithm*

## 4.1. Introduction.

In this chapter, a new algorithm called the *RASS* (*Regionalisation Algorithm with Selective Search)*, is proposed. The most important feature of this algorithm is that the way it operates is based on the characteristics of regionalisation processes, where the information available about the relationships between areas can play a crucial role in directing the search process in a more selective, more efficient and less random way.

The *RASS* incorporates inside its algorithm the optimisation model presented in chapter 3 in order to achieve local improvements in the objective function. These improvements can generate significant changes in regional configurations; these changes would be very difficult to obtain using other iterative methods.

## 4.2. 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 most heterogeneous geographical contiguity formed by $r$ regions with $2 \leq r \leq (m-1)$.

$$H(C_m) \equiv \sum_{\{i,j \in C_m | i<j\}} d_{ij} \rightarrow Max\left( \sum_{m \in M_l} H(C_m) \right) \qquad (4\cdot1)$$

where $M_l$ 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 the highest similarities to any of the regions in $r^*$.

$$I(C_{d,f}) \equiv prom\left(\sum_{i \in C_f} \sum_{j \in C_d | j>i} d_{ij}\right) \rightarrow Min\left(I(C_{d,f})\right) \qquad (4\cdot2)$$

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 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 the greatest differences compared 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*.

$$I(C_{d,e}) \equiv prom\left(\sum_{i \in C_e} \sum_{j \in C_d | j>i} d_{ij}\right) \rightarrow Max\left(I(C_{d,e})\right) \qquad (4\cdot3)$$

**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 were not selected in step 2 have been included at any time in *d*, or until there are no more candidates for selection 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 will be repeated. If the value of the objective function does not improve, step 2 will be repeated, but the next most heterogeneous group will be selected. Steps 2 to 8 will 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 of the most interesting characteristics of the *RASS* algorithm are the following:

a. The application of direct optimisation to a group of regions, in steps 3 to 6, allows improvements in the objective function that can be accompanied by major changes in regional configurations because of the re-assignation of a large 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 aim to keep the regions in the optimisation model (step 3) that have the highest potential to improve the objective function after reconfiguration.

   The objective is to ensure that the region included is the one that presents the highest probability of containing areas belonging to other regions. This potential re-assignation is identified assuming that in two regions with exchanged areas the dissimilarities decrease.

   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 appropriate number of areas for the optimisation model) is the one that differs most from the region to be included. This region has the fewest possibilities of exchanging areas with the region to be included ($e$).

c. The conditions in steps 7 and 9 aim 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* to escape from the local optimum.

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

## 4.3. Computational results and comparison with the direct optimisation.

This section seeks to evaluate the performance of the *RASS* algorithm with respect to the direct optimisation model proposed in chapter 3. The examples solved are the ones that were randomly generated in section 3.6[26]. 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 there are areas) using a uniform distribution between 0 and 1.

b. The interval [0,1] is divided into equal sized intervals, as many as the number of regions to be designed. 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,

[26] 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 lead directly to the optimal solution.

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 assigns areas to regions (potential initial partition).

c. If the initial partition is feasible in terms of geographical contiguity, this partition is used as a starting point for *RASS*.

Some descriptions of the results for the 30 problems considered (5 for each combination of regions and areas) are shown in Table 4.1. *RASS* achieved the optimal solution in 100% of the examples considered in a considerably shorter time than the direct solution method.

**Table 4.1. Comparison of *RASS* with the direct solution method.**

| Regions | Areas | Optimum/ 5 | Seconds (*RASS*) | Seconds (Direct) | (IOF - OF1c) (IOF - OS*) |
|---|---|---|---|---|---|
| 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% |

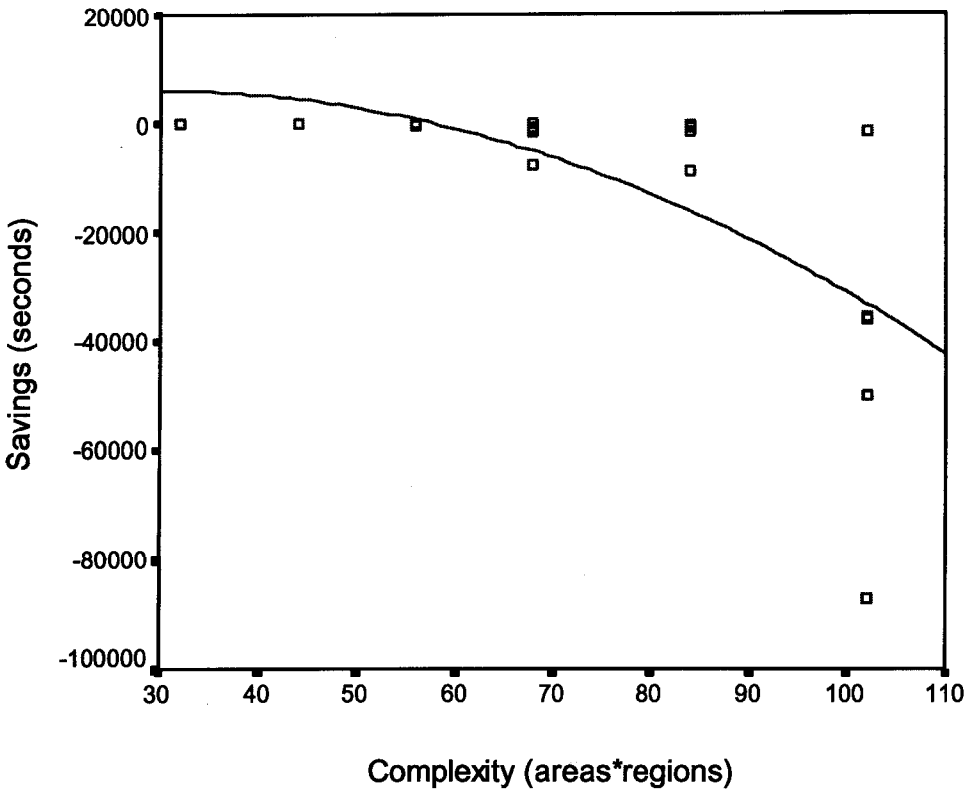
*IOF= Initial objective function, OF1c= Objective function after the first cycle, OF*= Optimal solution.*
Source: Own elaboration.

In the last column, it can be seen that after the first cycle of the *RASS*, the value of the objective function is reduced by 80% of the total reduction required to achieve the global optimum.

Using the information available about running times of both regionalisation methods, the direct method and the *RASS*, it is possible to calculate the time savings by applying the algorithm. Figure 4.1 shows the relationship between the savings and an indicator of complexity that is defined as the product between the number of areas and the

number of regions considered. The results in this figure show that in less complex models the direct method is the best option, while in complex models the *RASS* provides better results. The results suggest that this change happens for models with a complexity above 57.83 (58 if we retain the discrete nature of the variable[27]).

**Figure 4.1. Relationship between the complexity of the problem and the time savings obtained after applying *RASS*.**




Source: Own elaboration.

In order to obtain a better measure of the time savings achieved with *RASS*, we estimated a quadratic model between time savings and the measure of complexity[28,29].

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

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

The results of estimating this model are shown in Table 4.2. There is a significant relationship between the two variables at 1% significance level. With a marginal increase in the complexity of the problem, the use of *RASS* offers a time saving of 426.08-14.73 (*areas*regions*), a result that confirms our intuition mentioned above.

**Table 4.2. Quadratic regression among the time savings obtained with *RASS* and the complexity indicator.**

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

* *Significant at 1%*
Source: Own elaboration.

## 4.4. 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 running time for the latter would increases considerably. In this section we present the solution obtained for a regionalisation process where 38 areas are grouped in 10 regions (complexity of 38*10 = 380). For this comparison, the same procedure was applied as in the examples in section 3.4: 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*.

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

### 4.4.1. Data

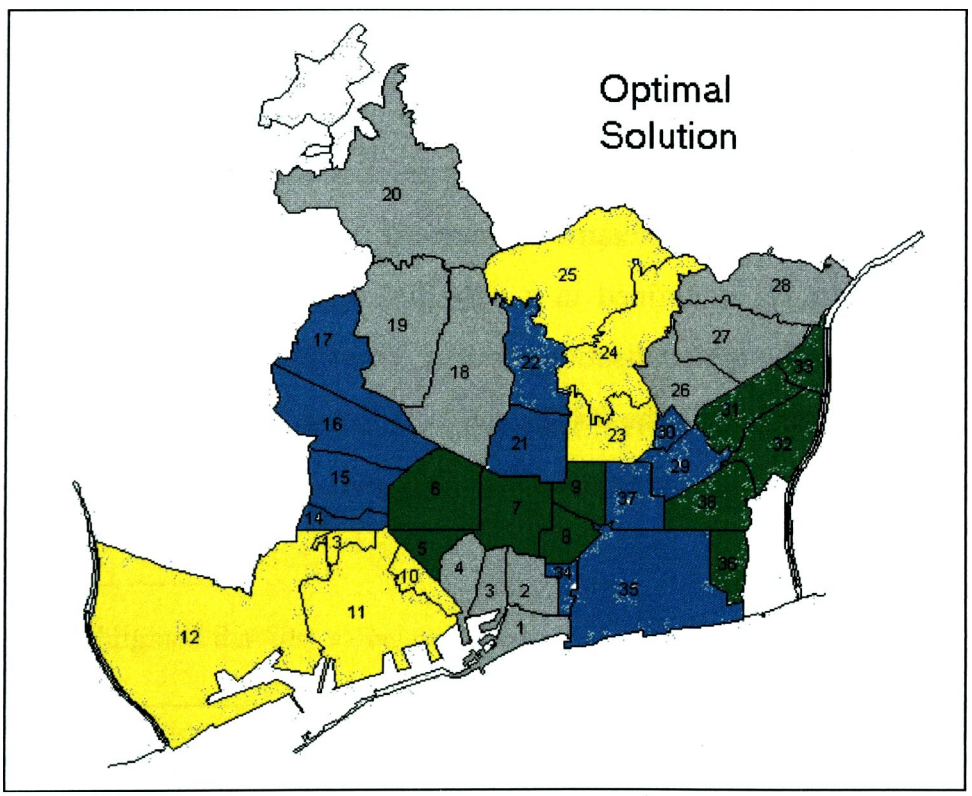
#### *4.4.1.1. Characteristics of the territory to regionalise.*

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

#### *4.4.1.2. Relationships between areas.*

The relationships between areas (see Table 4.3) were created in such a way that the optimal solution grouped the 38 areas in 10 regions, each of them with different shapes and sizes (between 2 and 6 areas by region). This optimal solution is shown in Figure 4.2, and this is the solution that the *RASS* algorithm should be able to identify.

**Figure 4.2. Pre-established optimal regional configuration.**




Source: Own elaboration.

**Table 4.3. Relationships matrix between the 38 areas.**

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

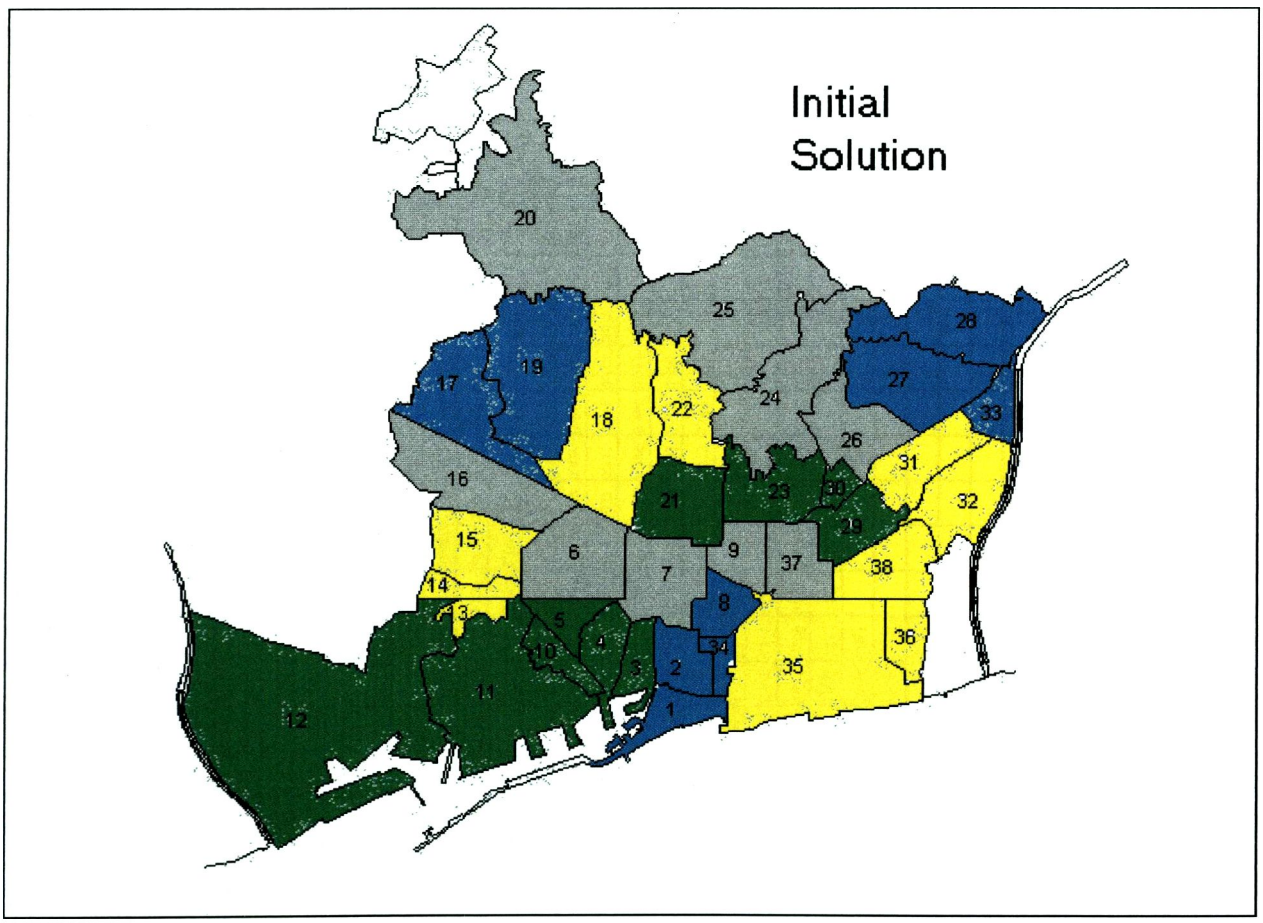
Source: Own elaboration.

### 4.4.2. Evaluation of results.

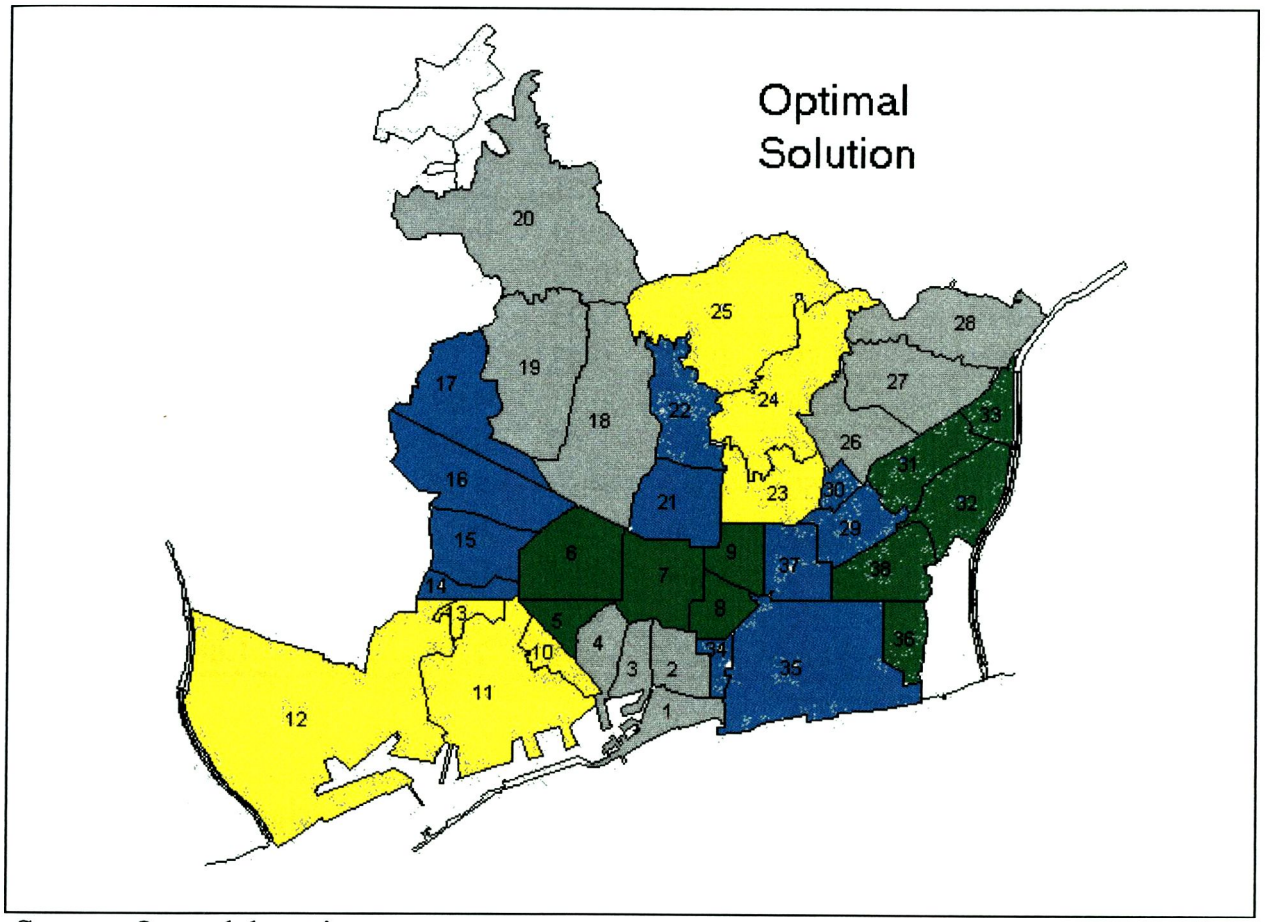
The initial partition is shown in Table 4.4. This is the partition that is considered by the *RASS* in step 1. Note that this configuration differs substantially from the optimal one. After 5 cycles, the *RASS* algorithm reaches the optimal solution.

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

**Table 4.4. Initial partition and solution obtained by the *RASS*.**




*continue...*




Source: Own elaboration.

In order to evaluate the evolution of the results from the initial partition up to the final results, Table 4.5 presents the value of the objective function 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 its minimum value is reached (1.08).

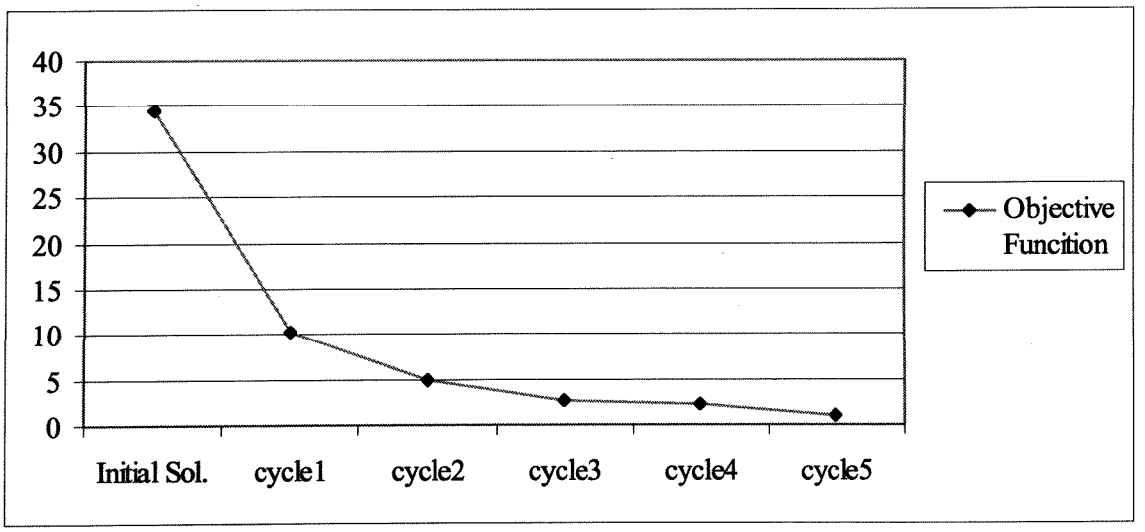
As Figure 4.3 shows, the behaviour of the objective function is as expected: the highest improvements are achieved in the first cycles.It is also confirmed that in each cycle the value of the objective function is better than, or at least equal to, the value in the previous cycle.

**Table 4.5. 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.

**Figure 4.3. Evolution of the objective function during the application of *RASS*.**

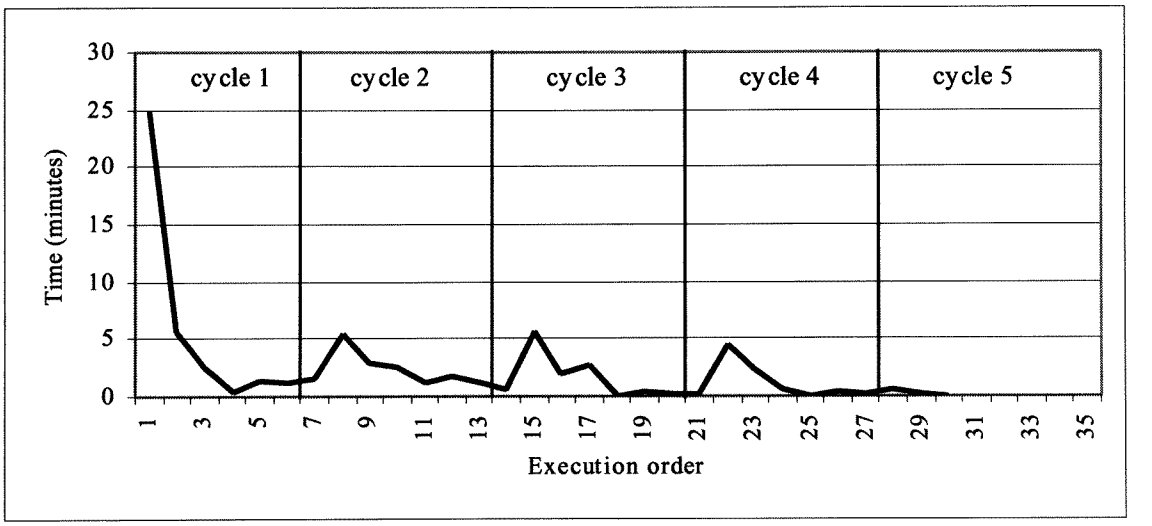



Source: Own elaboration.

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 were appropriate: the average

running time was 2.43 minutes by model. These running times are shown in Figure 4.4.

**Figure 4.4. Running times of optimisation models.**




Source: Own elaboration.

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

### 4.4.3. Sensitivity of the results to the initial partition.

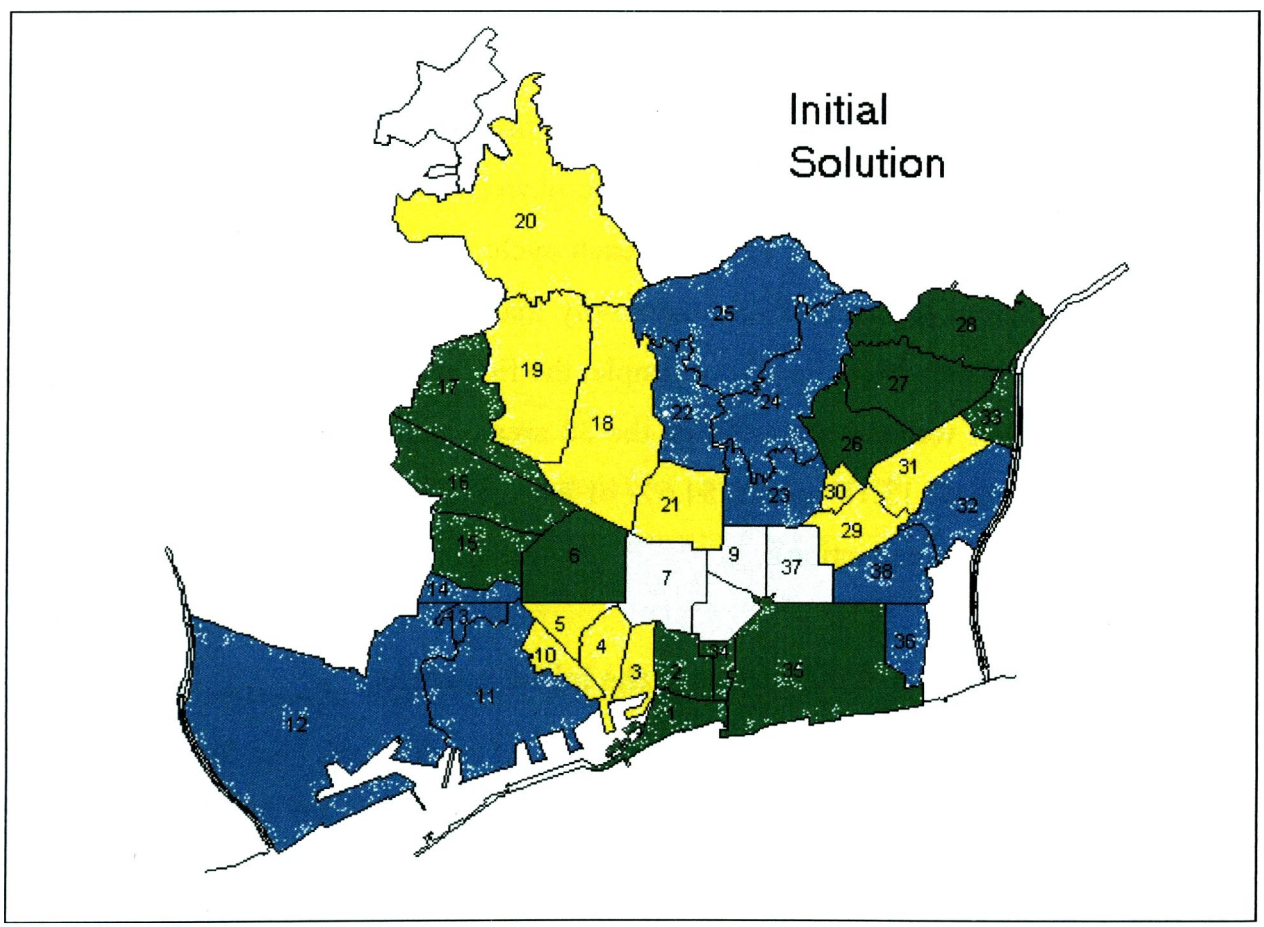
How can the initial partition affect the final result? In this sub-section, a different initial partition is used to solve the same problem as above. Thus, the initial partition in step 1 of *RASS* will be closer to the optimum regional configuration. With this

partition, a lower number of cycles are expected, and results similar to those in the previous sub-section .

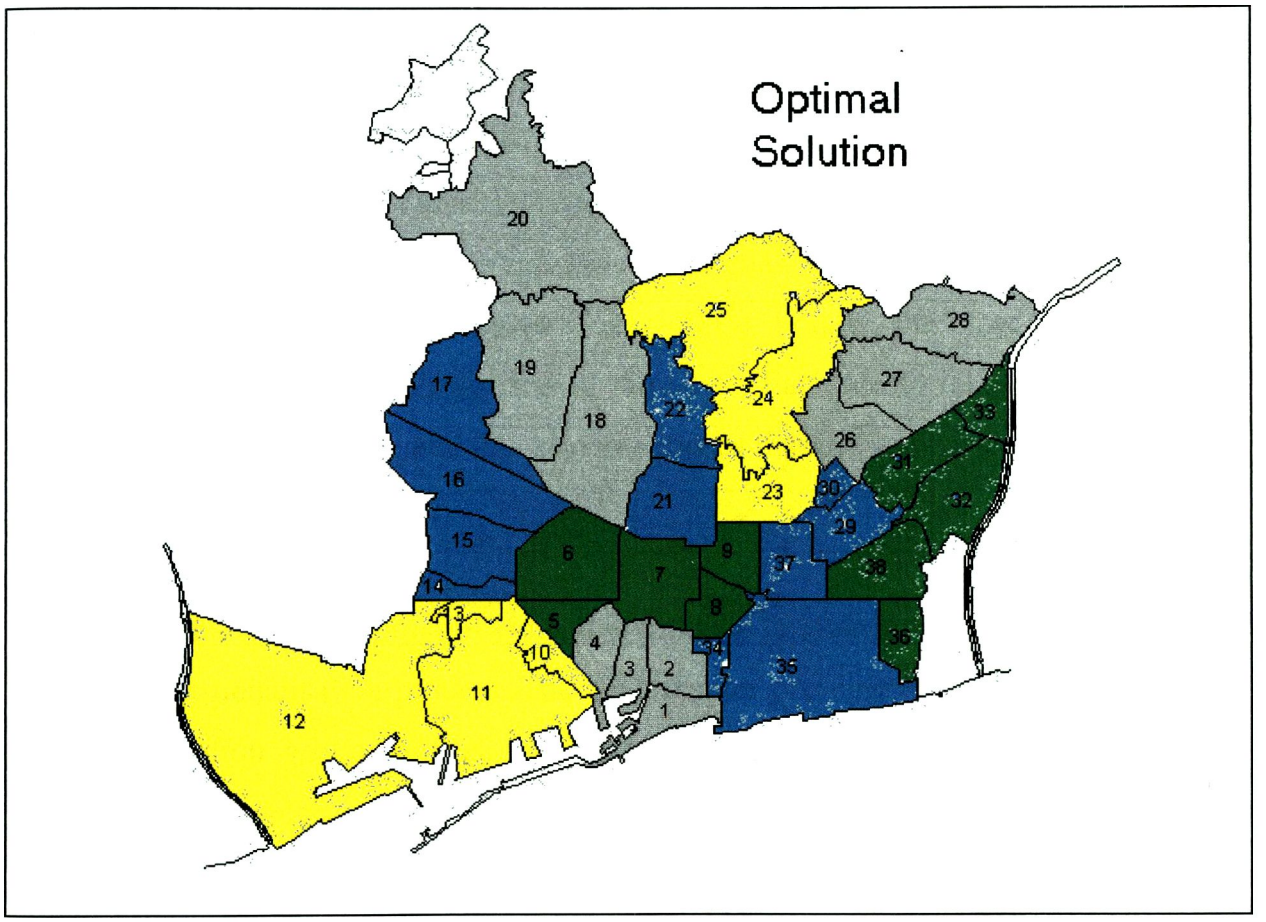
In this case, the optimal configuration was found after 2 cycles (see Table 4.6), 3 cycles less than in the previous example. The results shown in Table 4.7 and Figure 4.5 indicate that, as before, the greatest reductions in the objective function are achieved in the initial cycles of the *RASS*.

Regarding the impact of the first optimisation model on the objective function, there is now a reduction of 19.33 points (from 26.94 to 7.61), 79.25% of the total reduction obtained in the first cycle. Fifty per cent of the areas in the 4 (*r*) regions considered are now re-assigned (21.1% if the 38 areas are considered).

**Table 4.6. Initial partition (close to optimum) and solution obtained.**




*continue...*



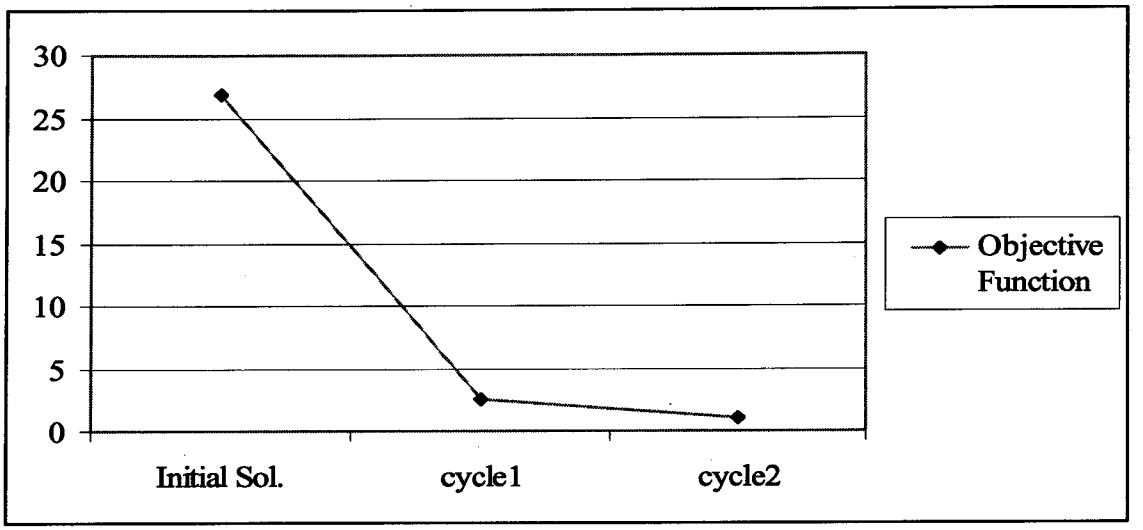

Source: Own elaboration.

**Table 4.7. Values of the objective function in the initial partition (closest 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 4.5. Evolution of the objective function during the application of *RASS* with the initial partition closest to the optimal solution.**




Source: Own elaboration.

## 4.5. Final remarks.

The results obtained allow us to conclude that the *RASS*, due to the incorporation of a direct optimisation routine as part of the algorithm, is well suited to achieving global optima 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 in such a way that the number of regions considered by the direct optimisation model ($r$) is 2 or higher and these regions should contain a number of areas in line with the computational capacity of the model. It has been calculated that the most appropriate $m/n$ relationship is above 14%. For example, if a territory formed by 8,000 areas is considered, the number of regions that can be obtained will be higher than or equal to 1,120 regions (an average size of 7 areas per region). This relationship ensures that $r$ can take values higher than or equal to 2 without substantially increasing the running time.

If the relationship between regions and the number of areas is very low, one possible strategy is to design nested regionalisation problems, which would involve sequential application of the *RASS*. For example, the city of Barcelona is divided into 1,919 statistical sections (*Seccions Estadístiques*, SE), which are grouped in 248 small research areas (*Zones de Recerca Petites*, ZRP). These areas are also grouped in 110 basic statistical units (*Unitats Estadístiques Bàsiques*, UEB) which form the 38 large statistical areas (*Zones Estadístiques Grans*, ZEG). Finally, the large statistical areas are grouped to obtain the 10 districts of the city[30]. Each territorial level is formed by grouping the previous one, and this also guarantees that the various grouping levels are self-contained.

Thus, from a theoretical point of view, it is possible to introduce useful elements for an analytical regionalisation process. The next chapter presents an empirical application. This application will allow us to assess the algorithm's performance in realistic regionalisation processes, and the results can be compared with other normative and analytical regionalisation methods.

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