Evaluating Redistricting of Electoral Areas

Tiago Miguel Correia de Almeida

Thesis to obtain the Master of Science Degree in

Computer Science and Engineering

Supervisor: Prof. Vasco Miguel Gomes Nunes Manquinho

Examination Committee

Chairperson: Prof. Francisco António Chaves Saraiva de Melo
Supervisor: Prof. Vasco Miguel Gomes Nunes Manquinho
Member of the Committee: Prof. Luís Manuel Silveira Russo

January 2021

Acknowledgments

I must thank my supervisor, Professor Vasco Manquinho, for his continuous support, patience and availability throughout the last year (and a half, now). His precious feedback and brilliant ideas guided me in this work and will continue to inspire in the future.

I would also like to express my gratitude towards my dear friends, parents and brother without whom all this and much more would have never happened or, at least, would not be such a joy. Because of you, I am truly lucky.

This work was partially supported by national funds through FCT with reference UIDB/50021/2020 and PTDC/CCI-COM/31198/2017.

Resumo

O problema da redistribuição de áreas eleitorais é um problema antigo em ciência computacional e política. Desde os anos 60 vários autores propuseram diferentes abordagens que pretendiam prevenir a redistribuição de áreas eleitorais de forma a que estas fossem benéficas para um determinado partido ou facção política (gerrymandering). Com o intuito de evitar casos de gerrymandering, a maioria das abordagens procura maximizar a compactidade dos distritos eleitorais. Contudo, o problema é computacionalmente complexo e vários requisitos têm de ser satisfeitos aquando do desenho de mapas eleitorais.

Neste trabalho, uma nova e compacta formulação Booleana é proposta para resolver o problema da redistribuição de áreas eleitorais. Esta formulação satisfaz todos as características típicas de mapas eleitorais, particularmente no quadro da contiguidade e igual representação popular. Além disso, é também proposta uma nova medida de compactidade que não depende dos centros geográficos. Adicionalmente, é apresentada uma formulação incompleta para ser utilizada em instâncias do problema nas quais é difícil encontrar valores óptimos.

Os resultados experimentais são obtidos desenhando mapas eleitorais para Portugal continental, considerando as propostas de mudança do actual sistema eleitoral Português. Os resultados obtidos revelam que a formulação proposta é mais eficiente que as anteriores a desenhar os mapas eleitorais Portugueses. Finalmente, utilizando os mais recentes resultados eleitorais, vários cenários de gerrymandering são estudados, mostrando como o resultado eleitoral pode ser deturpado alterando apenas o desenho dos mapas eleitorais.

Palavras-chave: Redistribuição de Áreas Eleitorais; Gerrymandering; Optimização Combinatória Multi-Objectivo; Desenho Territorial; Investigação Operacional

Abstract

The political districting problem is a long lasting problem in computer and political science. Since the 1960s several approaches have been proposed with the goal of preventing the redrawing of electoral districts in such way that they are beneficial to a certain party or political faction (gerrymandering). In order to avoid gerrymandering, most approaches focus solely on redrawing electoral maps that try to maximize the compactness of the electoral districts. However, this problem is computationally hard and several criteria must be satisfied when drawing electoral maps.

In this work, a new compact Boolean formulation is proposed for solving the electoral districting problem. This formulation satisfies all common criteria for building electoral maps, in particular the contiguity and representation criteria. Moreover, a new compactness measure is also proposed that does not depend on geographic centers. Additionally, an incomplete formulation is also devised for problem instances where the optimum values are hard to find.

Experimental results are obtained by drawing electoral maps for continental Portugal assuming a change in the Portuguese electoral system. Results show that the proposed formulations are more effective than previous ones in drawing the electoral districts in Portugal. Moreover, based on results from previous elections, several gerrymandering scenarios are devised, showing that electoral outcomes can be twisted depending on the drawing of the electoral maps.

Keywords: Political Districting; Gerrymandering; Multi-Objective Combinatorial Optimization; Territorial Design; Operations Research

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The power of gerrymandering is undeniable and it is a menace to democracy. Recently, in the 2018 United States House of Representatives elections in North Carolina the redistricting process was controlled by the Republican party. The Republicans received 50.39% of the popular vote (the Democrats received 48.35%) but ended up with 10 out of 13 congressional seats (76.92%) [37]. This power is the reason why the man behind the electoral maps of North Carolina, the Republican political strategist Thomas Hofeller said: ”Redistricting is like an election in reverse (...) usually the voters get to pick the politicians, in redistricting, the politicians get to pick the voters”.

Portugal does not suffer from gerrymandering due to its voting system. However, for most parties, there is still a large difference between the percentage of popular votes received and the percentage of seats won. The principle of representative democracy established in the Portuguese constitution is threatened by the electoral system in place which cannot deliver proportional representation for legislative elections. As an example, in the most recent legislative elections (2019) the Socialist party (the most voted) received 35.08% of the popular vote but elected 46.90% (106 out of 226 seats) of the national assembly.[1]

This is due to two reasons. Firstly, the electoral districts match the administrative subdivisions of the country and the population differences are huge between districts. Secondly, the D'Hondt method [21, 28] is used to distribute seats between parties. Because several districts only elect 4 or less officials and the D'Hondt method cannot guarantee proportionality when distributing a small number of seats (it is beneficial to the parties with the most votes) [28] the seat distribution ends up not being proportional to the popular vote.

Therefore, a discussion sprouted in Portugal around the electoral system to use for the legislative elections and the topic is not new. Since the 1990s, it has been on the political agenda of the two main Portuguese political parties. In 1997, in an attempt to bring populations closer to politics, increase the plummeting turnout rates and guarantee proportional representation, the national assembly agreed that it was necessary to change the electoral system and the constitution was changed to allow the

[1] Available at (in Portuguese): `https://www.eleicoes.mai.gov.pt/legislativas2019/territorio-nacional.html`, accessed 22/12/2020.

use of single-member districts.[2] The following year three different propositions were made using single-member districts. However, a consensus was not reached. The result was that all three propositions were rejected in parliament and, ultimately, nothing changed. Hence, the problem persists.

1.1 Why Complete Methods and Multi-Objective Optimization?

Building electoral maps is a computationally hard problem (NP-Complete), and complete (exact) methods are, traditionally, more time consuming than incomplete (heuristic) methods [2, 53]. Although, we find that the use of complete methods is appropriate for two main reasons. Complete methods are the only option that guarantees global optimal solutions and that is of the utmost importance in a democracy given the relevance of the redistricting process in the electoral results and its influence in turnout rates [32, 35]. Additionally, the electoral maps need only to be redrawn after every census which is, typically, every 10 years and during this period they are used for (at least) 2 elections justifying the computational time necessary to find such solutions.

Several measures have been proposed to evaluate the quality of the new electoral maps. The most important are: equal popular representation, district contiguity and district compactness [52, 53]. Multi-objective combinatorial optimization allows the optimization of several key features concurrently, e.g., maximizing the compactness while, at the same time, minimizing the popular representation differences between electoral districts. Moreover, one of the main focus of this work is to study the impacts of gerrymandering electoral areas. Therefore, it requires at least the maximization of the electoral results of a party along with the maximization of electoral district compactness (two objective functions).

1.2 Goals and Contributions

This document provides an overview of previous approaches to the political districting problem and the territorial districting applications of this problem to different fields. There is a focus on complete methods, although, the advantages and disadvantages of incomplete methods are also presented.

The main goal of this work is to build upon the state of the art and develop a new and efficient multi-objective combinatorial optimization model. It must fulfill the three main characteristics of electoral maps (compactness, contiguity and equal representation) and also possess the ability to optimize the electoral results of a party. We study the adaptation of the optimizations added to our model to previous approaches to the problem and conclude that some can be successfully adapted and help significantly reduce execution times. Two methods to create contiguous districts are proposed and compared, a complete method and an incomplete one, where some solutions are not contemplated.

The developed model is versatile and is used to create unbiased electoral maps for Portugal using only official real-world geographical data. Additionally, using real electoral results from previous elections, biased (gerrymandered) maps can also be generated. The unbiased electoral maps follow the

[2] Article No. 93 of the constitutional revision of 1997. Available at (in Portuguese): `https://dre.pt/pesquisa/-/search/653562/details/maximized`, accessed 22/12/2020.

main propositions of the Portuguese parties with the goal of shedding some light on what the electoral maps for Portugal using single-member districts would look like. To our knowledge, this is the first time intentionally gerrymandered electoral maps are created automatically. Using real-world data, the goal is to study the effects of gerrymandering, not only in the electoral results but also in the design (shape) of the electoral districts, contributing to the detection of such cases.

Extensive experimental tests carried out using data from Portugal compare our proposed model with a classical approach to the problem. Overall, we improve upon the state of the art by being able to build compact and contiguous electoral district maps more effectively than previous approaches.

To summarize, this thesis makes the following contributions:

- We create a new and more compact multi-objective combinatorial optimization model to be used in the political districting problem;
- We propose a new measure of compactness based on the size of the frontiers between electoral districts;
- We propose and compare two new formulations to create contiguous districts, one using complete methods an one using incomplete methods;
- We propose optimizations that help improve previous approaches to the problem, as well as our own models;
- Using official data, we present the first complete electoral maps of Portugal using single-member districts;
- We create gerrymandered electoral maps and study its effects on electoral results and electoral map design.

1.3 Document Structure

This document is organized as follows:

Chapter 2 defines important concepts for this work, including electoral and voting systems, the concept of gerrymandering, the formulation of problems using multi-objective combinatorial optimization (MOCO) and a method of comparing MOCO solutions.

Afterwards, Chapter 3 classifies approaches to the territorial districting problem and presents relevant works in the particular problem of political districting using incomplete and complete methods.

Chapter 4 explains our proposed multi-objective combinatorial optimization model and the techniques used to guarantee equal popular representation (Section 4.1) and compact electoral districts (Section 4.2). Complete and incomplete methods to generate contiguous electoral districts are presented in Sections 4.3.1 and 4.3.2 while the necessary objective functions to create biased electoral maps are presented in Section 4.4. This is followed by Section 4.5 detailing the optimizations that are added to our model and adapting them to a classical approach to the political districting problem. Finally, the chapter ends by exploring the complexity of the created formulations in Section 4.6.

The experimental procedure along with its results are presented and discussed in Chapter 5. Multiple scenarios are tested focusing on the territory of Portugal creating electoral maps that only maximize the compactness of electoral districts (Sections 5.1 and 5.2) and also gerrymandered electoral maps (Sections 5.3, 5.4 and 5.5). A full comparison of approaches, methods and solvers is performed in Section 5.6.

Finally, the thesis concludes in Chapter 6 with the main conclusions drawn from this work and a brief discussion of possible future work.

Chapter 2

Preliminaries

In this section we present further explanation of all the terms necessary to follow this work. In particular, we introduce the notion of electoral district and briefly refer to the different voting systems used in Europe, gerrymandering, single and multi-objective combinatorial optimization and, finally, the hypervolume indicator.

2.1 Electoral Districts and Voting Systems

An electoral district is a territorial subdivision of a country (does not necessarily have to match existing administrative subdivisions) which elects members to the legislative body of that country.[1] The number of elected members in each district can vary depending on the electoral system in place.

The electoral system is the set of rules that determines when the elections take place, who can vote and the voting system (how candidates are elected). There are many voting systems, but the most used are party-list proportional representation (e.g. the Portuguese system for legislative elections), first-past-the-post voting (also known as *winner takes all*, used in United States of America elections), the two-round system (common in head of state elections such as the French or Portuguese presidential elections) and ranked voting where voters rank several candidates by preference (used to elect members to the legislature of Australia or the president of India, for example) [16, 25].

A single-member district is an electoral district where only one candidate is elected to a body with multiple members (e.g. a legislature). The voting system usually used in these districts is the first-past-the-post (FPTP) where the person with the most votes wins the district even if a majority (50% of the votes) was not achieved. Although less common, instant-runoff voting (IRV) (e.g. Australia) and the two-round system (e.g. France) are also a possibility and these systems guarantee a majority to the winner. One argument against single-member districts is that they tend to favor two-party systems (Duverger's law) [16, 54] resulting in fewer minority parties in parliament. In order to avoid such scenario, many countries, aiming at proportional representation, combine single-member districts with party-list

[1] There can also be (possibly different) electoral districts within administrative subdivisions of a country to elect local legislatures (e.g. state elections in the United States of America or local elections in Portugal).

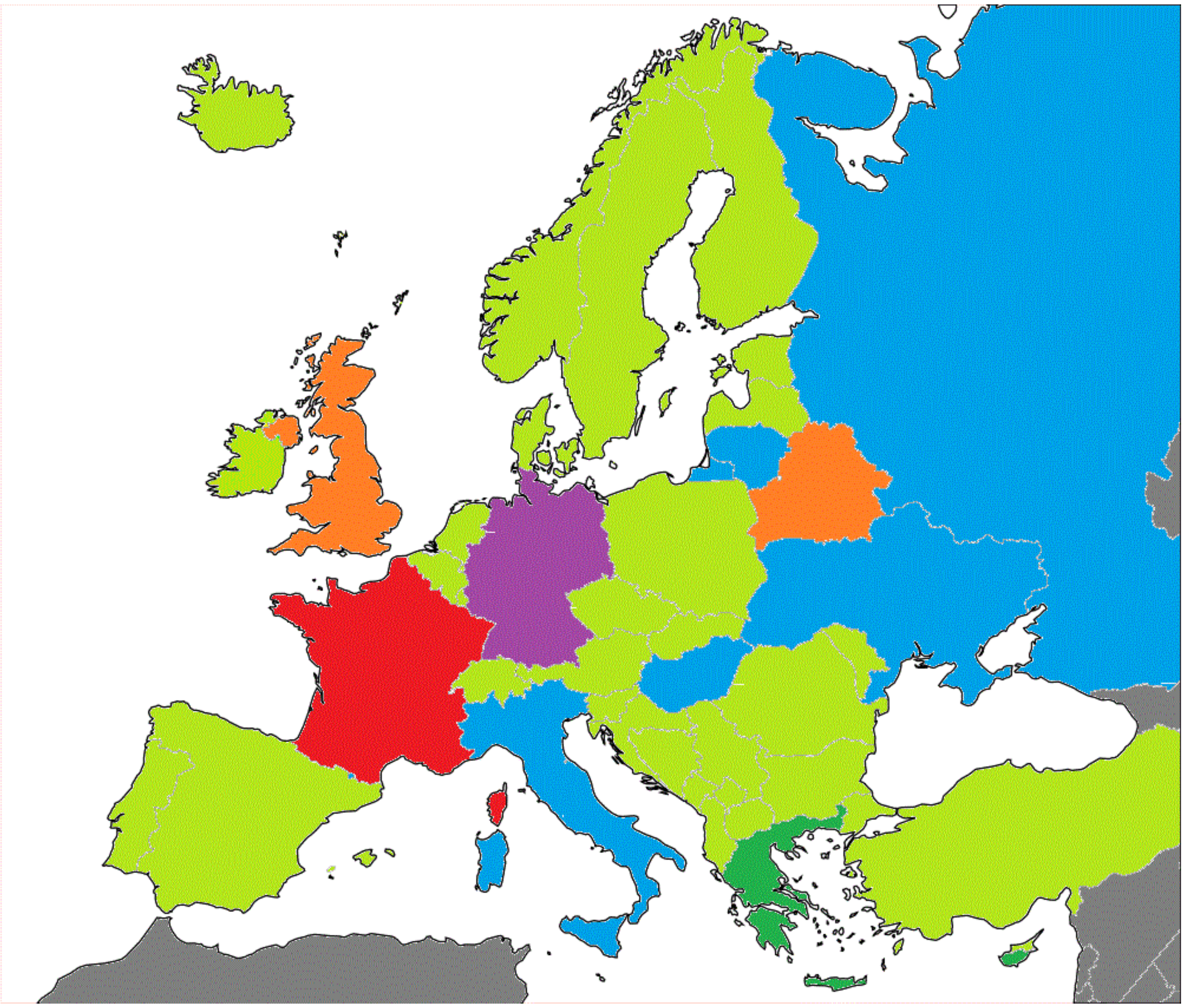


Figure 2.1: Voting systems for legislative elections in Europe.

proportional representation (this is called parallel voting and it is represented by the countries in blue in Figure 2.1).

In order to create new electoral maps, new electoral districts must be created by joining territorial units (indivisible) to form clusters. There are multiple criteria to evaluate the quality of the new map but the core ones are:

1. Each elected member should represent approximately the same number of people. Ideally each elected member would represent the theoretical best value B of people where B equals the total number of people to be considered divided by the number of officials to be elected.

2. All the new electoral districts should be contiguous. An electoral district ED is contiguous if, and only if, to go from any point A_1 inside ED to any other point A_2 inside ED as well, it is not necessary to leave ED.

3. All districts should be compact meaning that odd shapes (like the ones in Figures 2.2 and 2.3) should not exist. To measure compactness multiple ideas have been proposed such as the Schwartzberg score [55], the Reock score [51] or simply summing the Euclidean distances between the geographical centers (centroids) of each territorial unit inside a district [47, 61].

Although less crucial, there are other criteria (sometimes used in political districting) such as the conformity to administrative boundaries, that is, the respect of existing administrative subdivisions (used, as much as possible, in the United Kingdom) and the respect of natural boundaries in cases where mountains or rivers may be a problem for the contiguity of the districts [52].

Figure 2.1 shows the different voting systems for legislatures in Europe where green represents proportional representation obtained either through a party-list proportional representation system[2] or, in Ireland, through single transferable vote (ranked voting).

However, in this work we focus on single member districts used in non-green colored countries. Color orange is used to represent FPTP systems. France (red) uses the two-round system which is essentially the same as FPTP but in order to win a district a candidate must receive over 50% of the votes. Therefore, two rounds might be needed (in the second round only the two most voted candidates of the first round remain). Finally, blue and purple represent parallel voting and mixed-member proportional representation which are essentially a mix between the party-list proportional representation and the FPTP voting systems.

2.2 Gerrymandering

Gerrymandering is the practice of redrawing the boundaries of an electoral district in order to make it more beneficial to a certain party or political faction. Although less usual, gerrymandering can also be used to increase/decrease the voting power of a racial minority (racial gerrymandering) [1, 42]. As a result, the electoral districts often end up with odd shapes (see Figure 2.3 for a real world example). The term gerrymandering first appeared in 1812 in a political cartoon that caricatured a new district in the Boston area. It had been approved by the then governor of Massachusetts, Elbridge Gerry, and looked like a salamander thus being called a gerrymander (Figure 2.2).

There are two main gerrymandering tactics: cracking and packing. Cracking, dilutes the voting power of the opposition in as many electoral districts as possible, weakening them (Figure 2.4 case C). On the other hand, packing takes place when people that are likely to vote for the opposition are packed into a small number of districts, guaranteeing their victory in this small number of districts but rendering their vote useless in the remaining ones (see Figure 2.4 case D).

These techniques are precisely what explains the results in the 2018 United States House of Representatives elections in North Carolina, when the Democrats received 48.35% of the votes but only 3 out of 13 seats (23%) [37]. In reality, the Democrat voters were packed in those 3 districts (district 1, 4 and

[2]in Greece, i.e., dark green, this is applied to elect 250 officials and the remaining 50 seats are given as a bonus to the winning party – majority bonus system.

Figure 2.2: The original cartoon that coined the term gerrymandering as it appeared in the Boston Gazette of March 26, 1812.

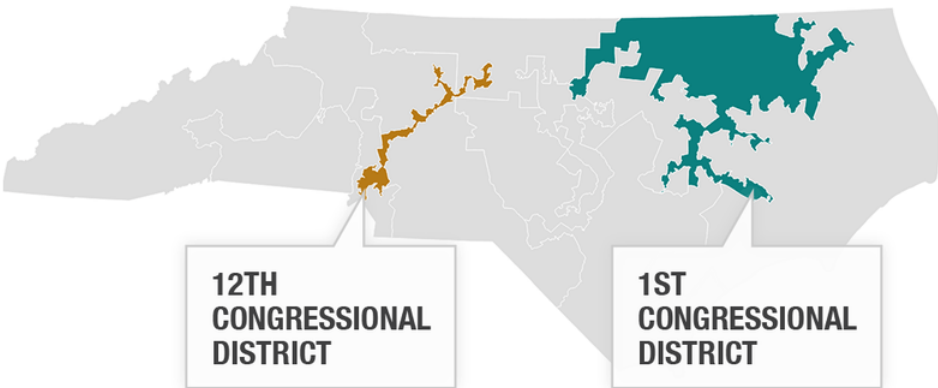


Figure 2.3: Real case of (racial) gerrymandering in North Carolina that created extremely odd shaped districts [38]. These two congressional districts were eventually ruled unconstitutional by the United States Supreme Court in 2017 [1].

12) where they received over 70% of the votes in each of them. They were also cracked between the remaining 10 districts which they ultimately lost by a small margin.

If we take a closer look at Figure 2.4 we can clearly see the potential of gerrymandering to completely alter the outcome of elections. Our objective is to group the 25 voters (red and green define their expected voting preference) into 5 single-member electoral districts, using a *first-past-the-post* system, where each district is won by the party with the most voters inside that circle. In order to guarantee that each elected member represents the same number of people – equal voting power – each district must have five voters.

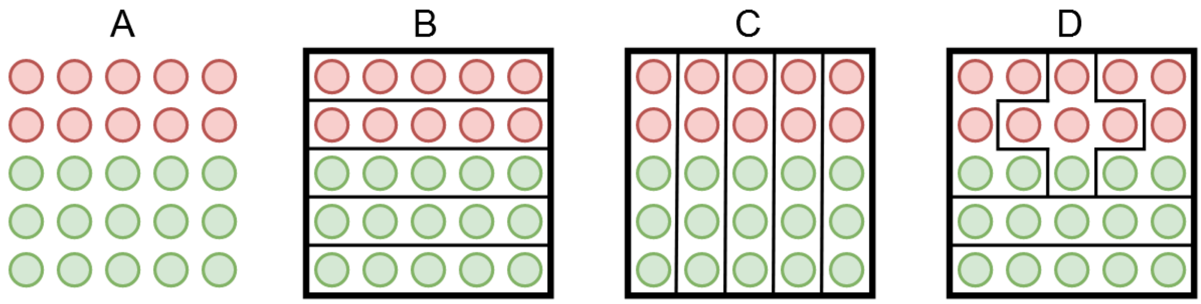


Figure 2.4: A possible example of three redistricting plans given the hypothetical election results in A.

In case B we are upon a fair distribution because the red party, which got 40% of the votes, gets 40% of the districts and the green party with 60% of the votes, 60% of the districts. However, this is not the case in the other 2 examples where gerrymandering techniques have been used. In redistricting example C, by cleverly cracking the red voters into the 5 districts the green party ends up winning all of them. Contrarily, in D most of the green voters have been packed in two districts (their only victories) and the rest cracked in the remaining 3 districts ultimately making the red party win 60% of the districts even though they only received 40% of the overall votes.

2.3 Single and Multi-Objective Combinatorial Optimization

The political districting problem can be formulated as a constraint optimization problem. Hence, in this section we formally define single and multi-objective optimization.

Given a set of Boolean variables $X = \{x_1, x_2, x_3, ..., x_n\}$ and their respective coefficients $W = w_1, w_2, w_3, ..., w_n \in Z$, a Pseudo-Boolean (PB) expression is a weighted sum of Boolean variables, represented as follows:

$$\sum_{i=1}^{n} w_i.x_i \tag{2.1}$$

Given an integer $K \in Z$ a linear PB constraint has the form:

$$\sum_{i=1}^{n} w_i.x_i \geq K \tag{2.2}$$

Given a set of PB constraints $P = \{t_1, t_2, t_3, ..., t_m\}$, a set of Boolean variables X and the costs $C = c_1, c_2, c_3, ..., c_n$ associated with each variable in X, the goal in a single objective Pseudo-Boolean Optimization (PBO) problem is to find an assignment α (solution) that satisfies all constraints in P, and minimizes the objective function (cost function) f where f is a linear PB expression over the Boolean variables. The objective value of α for objective function f is denoted $f(\alpha)$. In general, a PBO problem can be formulated as follows:

$$
\begin{aligned}
\text{minimize} \sum_{i=1}^{n} c_i.x_i & \\
\text{subject to} \sum_{i=1}^{n} w_{it}.x_i \geq K_t & \qquad t \in \{1 \dots m\} \\
x_i \in \{0, 1\} &
\end{aligned}
\tag{2.3}
$$

Example 2.3.1. Consider two Boolean variables $X = \{x_1, x_2\}$, two constraints $P = \{P_1, P_2\}$ and the objective function f where $f(x) = 3x_1 + x_2$. Let $P_1 = \{x_1 + x_2 \geq 1\}$ and $P_2 = \{x_1 + x_2 \leq 1\}$. This instance has only one optimal solution (assignment): $\alpha = \{(x_1, 0), (x_2, 1)\}$ where $f(\alpha) = 1$. All other solutions either do not satisfy P or do not minimize f as much as α does.

However, in many real-world problems, more than one objective function must be considered [3, 7, 62, 4]. Therefore, unlike the formulation in (2.3), a Multi-Objective Combinatorial Optimization (MOCO) problem considers a set $O = \{f_1, f_2, f_3, ..., f_q\}$ of objective functions to minimize instead of a single one.

Although, in most cases we are upon conflicting objective functions meaning that there is not a solution that simultaneously optimizes all of them (non trivial problems) and multiple Pareto optimal solutions exist. Generally the complexity of MOCO problems is exponential on the number of variables.

$$
\begin{aligned}
\text{minimize} \sum_{i=1}^{n} c_i^1.x_i & \\
\text{minimize} \sum_{i=1}^{n} c_i^2.x_i & \\
\vdots & \\
\text{minimize} \sum_{i=1}^{n} c_i^q.x_i & \\
\text{subject to} \sum_{i=1}^{n} w_{it}.x_i \geq K_t & \qquad t \in \{1 \dots m\} \\
x_i \in \{0, 1\} &
\end{aligned}
\tag{2.4}
$$

Given two assignments α_1 and α_2 such that $\alpha_1 \neq \alpha_2$, we say that α_1 dominates α_2 if, and only if, $\forall_{f \in O} f(\alpha_1) \leq f(\alpha_2)$ and $\exists_{f' \in O} f'(\alpha_1) < f'(\alpha_2)$.

A Pareto optimal (or Pareto efficient) solution is an optimal trade-off between objectives, a state from which any further objective optimization is impossible without impairing at least one other objective even more. The set of all Pareto optimal solutions is called the Pareto frontier (or Pareto front) and can be represented graphically (Figure 2.5). All the solutions that are not part of the Pareto frontier are dominated by at least one Pareto optimal solution.

Figure 2.5 presents the set of all solutions for a problem where we want to minimize objective functions f_1 and f_2. The set of Pareto optimal solutions (all non-dominated solutions) is colored in red and makes up the Pareto frontier while all other solutions, colored in gray, are dominated solutions since they

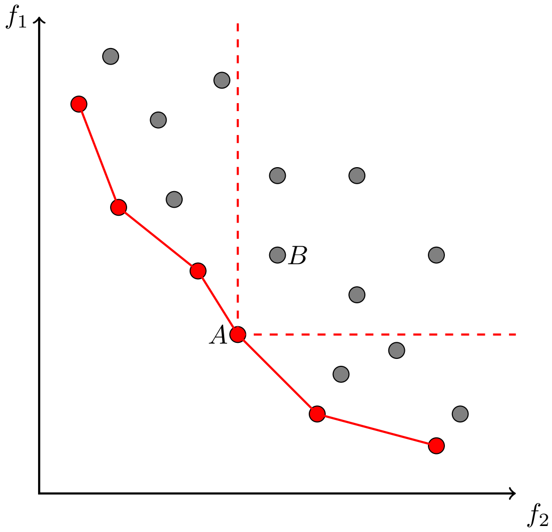


Figure 2.5: Representation of solutions for a multi-objective minimization problem.

are worse than (dominated by) at least one Pareto optimal solution. For example, solution A dominates solution B (and all solutions inside the dashed area) because it minimizes both objective functions more than B does, i.e., $f_1(A) < f_1(B)$ and $f_2(A) < f_2(B)$.

Example 2.3.2. Consider three Boolean variables $X = \{x_1, x_2, x_3\}$, the constraint $P = \{x_1+x_2+x_3 \geq 1\}$ and two objective functions $O = \{f_1, f_2\}$ to minimize. Let $f_1(X) = 3x_1 + x_3$ and $f_2(X) = 2x_1 + x_2$. This instance has 2 Pareto optimal solutions (assignments):

- $\alpha_1 = \{(x_1, 0), (x_2, 1), (x_3, 0)\}$ where $f_1(\alpha_1) = 0$ and $f_2(\alpha_1) = 1$.
- $\alpha_2 = \{(x_1, 0), (x_2, 0), (x_3, 1)\}$ where $f_1(\alpha_2) = 1$ and $f_2(\alpha_2) = 0$.

All other solutions either do not satisfy P or are dominated by (at least) one of the Pareto efficient solutions. For example, consider the assignment $\beta = \{(x_1, 1), (x_2, 1), (x_3, 1)\}$ where $f_1(\beta) = 4$ and $f_2(\beta) = 3$. Because $f_1(\alpha_1) < f_1(\beta)$ and $f_2(\alpha_1) < f_2(\beta)$, α_1 strictly dominates β, i.e., α_1 is strictly better than β.

2.4 Hypervolume Indicator

In order to assess and compare the performance of algorithms for MOCO problems, we need to evaluate the quality of the solutions returned. The hypervolume indicator [63] does precisely that by measuring the size of the dominated space of a given approximation of the Pareto frontier with respect to a certain reference point.

Let $S = \{s_1, s_2, s_3, ..., s_n\}$ be the set of solutions that make up the approximate Pareto frontier for a MOCO problem and Z a strictly dominated solution by all solutions $s_i \in S$. The hypervolume indicator for the set S is given by the volume enclosed by the union of the set of polytopes $H = \{h_1, h_2, h_3, ..., h_n\}$. Each polytope h_i is formed by the intersection of the following hyperplanes arising out of s_i, along with

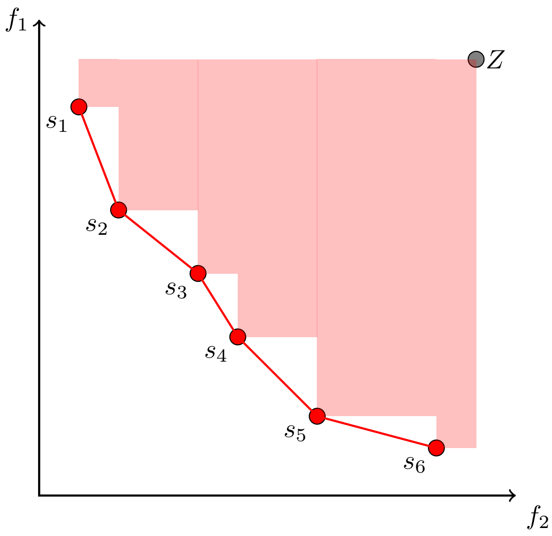


Figure 2.6: Hypervolume indicator of an approximation of the Pareto frontier using reference point Z.

the axes: for each axis in the objective space, there exists a hyperplane perpendicular to the axis and passing through Z. For a two-dimensional case each h_i represents a rectangle (a two-dimensional polytope) with vertices at s_i and Z.

Figure 2.6 presents a set of solutions $S = \{s_1, s_2, s_3, ..., s_6\}$ (an approximation of the Pareto frontier) for a problem where we want to minimize two objective functions (f_1 and f_2) and a reference point Z. The hypervolume indicator is given by the area represented in red which corresponds to the union of all the rectangles with vertices at $s_i \in S$ and Z.

Chapter 3

Related Work

In this section we review relevant previous work on territorial districting. We start with a brief introduction and classification of approaches to the territorial districting problem and then focus on the particular problem of automatic electoral map creation.

3.1 Territorial Districting

The electoral districting problem (also known as the political districting problem) is essentially a particular type of the territorial districting problem (also known as the zone design problem) which has many applications in a broad variety of fields, such as sales districting [64], students distribution between schools [26], harvest scheduling [14, 43, 46], natural reserve creation [9], electrical power distribution [7].

The territorial districting problem consists of grouping the elementary units of the territory to form zones (or districts). Therefore, a territory is composed of zones, each one resulting from a grouping of elementary units. To model this problem, graph theory is usually used along with mathematical optimization techniques. A graph is created composed of nodes that represent the elementary units of the territory while a pair of contiguous elementary units defines an edge of the graph. Depending on the application, different values are associated to the nodes and edges, but population and geographical distance between nodes, respectively, are the most common for political districting.

There have been many approaches to the territorial districting problem (especially to the electoral districting problem) and we can classify them by looking at 3 key aspects:

1. There are two main techniques to solve this problem and create the final map: division and agglomeration. If division is applied, the whole area is considered and the districting algorithm iteratively divides it into smaller areas [5, 27]. Although, the most usual are agglomeration algorithms where the elementary units of the territory are successively pieced together to form contiguous districts [10, 29, 59].

2. Another way of classifying districting problems is to differentiate between single-objective and multi-objective. Single-objective models only consider one criteria for the districting, e.g., minimal

amount of changes to the current map [11], least aggregations to form a district [9] or population equality [29, 59]. However, sometimes it is useful to consider multiple objective functions at the same time and try to find a solution that is the optimal compromise (trade-off) between the objective functions considered [5, 7, 39, 56].

3. The final difference between all these approaches to the territorial districting problem is the use of complete (exact) methods or incomplete (heuristic) ones. Exact techniques are, usually, much more computationally complex since they need to check all possible solutions for a problem. However, they yield global optimal solutions, which, depending on the problem to be solved, may be vital [29, 39, 40, 45]. On the other hand, non-exact approaches make use of heuristics to find a solution faster which makes them interesting for solving large scale problems. Although, there is no guarantee that the solution found is the global optimal one (most of the times it is just a local optimum) [4, 5, 7, 31, 56].

3.2 Automatic Electoral Map Creation

The idea of creating new electoral maps through the use of computers was first proposed in the 1960s. Vickrey (1961) argued that the "Elimination of gerrymandering would seem to require the establishment of an automatic and impersonal procedure for carrying out a redistricting. It appears to be not all difficult to devise rules for doing this which will produce results not markedly inferior" [59]. However, even with greatly increased computational power, 60 years later the problem is still to be solved. The biggest reason lies in the complexity of the problem. Guest et al. [31] argued that this task may be too complex for humans and most cases of gerrymandering may not be intentional at all, particularly in larger districts. Altman proved the redistricting problem to be NP-Complete, i.e., it belongs to the class of computationally intractable problems (no known polynomial time solution) [2]. Nevertheless, it does not mean it is impossible to reach an optimal solution for smaller areas and the author admits that it may be possible to find global optimal solutions given the right conditions. For this reason, most of the proposed solutions we describe next only create maps for smaller districts.

Different approaches to the electoral redistricting problem have been made, using either heuristic methods (most common) or exact methods [53]. However, all of them need to follow at least one restriction: proposed districts must comprise (roughly) the same population.

3.2.1 Incomplete Methods

Heuristic methods usually make iterative improvements. Hill climbing (local search) algorithms are a common option, these work by starting with an existing electoral map of a district and repeatedly trading census blocks (or voting precincts) between electoral districts and calculating if it resulted in an improvement, according to the objective function. A variant of this method is to start a district with a single census block and adding to it the neighbors that most improve it (usually the algorithms try to maximize compactness therefore prioritizing proximity) until the district meets the population requirement. This was the approach proposed by Vickrey [59].

Simulated annealing methods have been proposed since 1990 [11, 18] and they work similarly to the hill climbing methods (also with heuristics) and try to approximate a global optimum for a given objective function. Small changes are made in each iteration, always keeping changes that improve the redistricting plan according to the objective function (e.g. increased compactness) and, occasionally, keeping changes that harm the system (e.g. making districts less equal in population) until an optimal or nearly optimal solution is reached, a solution that minimizes the temperature (energy) of the system. This randomness produces a solution that is independent of the initial conditions and aims at solving the shortcomings of hill climbing. Contrarily to the latest, simulated annealing does not go directly (greedily) to the nearby solution, which most of the times is just a local optimum.

Clustering algorithms have also been widely used and in order to apply them many authors adopt a graph-theoretical model for territorial representation. They usually make use of a *contiguity graph* (or *population graph*) which is an n-node connected weighted graph $G = (N, E)$ where the N nodes correspond to the smallest geographic unit considered (e.g. census blocks in the United States of America) and an edge between nodes exists if and only if those geographic units have a common frontier (are neighbors). In this graph, for each edge $(u, v) \in E$, there is an associated weight that usually represents the (geographical) distance between the center of each geographic unit (either calculated through Euclidean geometry or road distance). The weight associated to the nodes represents the population living inside this geographical area. This model was first introduced by Bodin [10] in 1973 and the author used it to redistrict the State of Arkansas based on the 1960 census data. The proposed algorithm consisted of 2 stages. In the first one, starting from a set of k centers (manually picked) the algorithm constructs k districts (trees), each rooted at one of the centers. In order to do this, it starts from the k centers and successively selects an unassigned node to include in the most advantageous district with respect to population balance. In the second stage, a local search algorithm is applied in order to improve population equality. Given two districts d_1 and d_2 the algorithm tries to make one of the following changes: move a node u from district d_1 to district d_2; or move a node u from district d_1 to district d_2 and a node v in the opposite direction. It is important to notice that the described algorithm takes into account the contiguity and population equilibrium but not the compactness of the new districts since the algorithm does not control the shape of the clusters.

A solution to the aforementioned problem was proposed by Guest et al. [31] who applied the weighted k-means clustering algorithm (taking into account population balance) to generate compact districts (clusters) with roughly the same population. The k starting points are initialised using the procedure from k-means++ (to spread the k clusters apart). Next, in each iteration of the algorithm, census blocks are assigned to the nearest cluster and the centroid (geographical centre of the cluster) is updated to reflect the added census block. This process is repeated until it converges to a local optimum where the mean distance between people within the same district is minimized and the population in each district is roughly the same [31].

Bação et al. applied a genetic algorithm to redistrict Portugal's electoral map [4]. Portugal has a proportional representation system but the authors considered a parallel voting system with 93 single member districts ($k = 93$). They represented each territorial unit by its centroid for which the geographical coordinates are already known, as well as the Euclidean distances between each pair of units. To generate a solution k territorial units are chosen as centers, and then districts are built around those centers by successively assigning other units to the closest center. The algorithm proceeds by generating a new set of solutions through the application of genetic operators (mutation, crossover and selection). For each solution, the contiguity is checked and it is evaluated by the following fitness function: $\sum_{j=1}^{k}(|P_j - \mu| + \sum_{i \in Z_j} d_{ji})$, where μ is the average population in all districts and d_{ji} is the Euclidean distance between the ith territorial unit and the centroid j of district Z. The algorithm stops after a predefined number of generations without improvements is observed.

However, heuristic methods have 2 main problems. Primarily, they cannot guarantee global optimal solutions which is not desirable when creating new electoral maps given their importance in election results. For this reason, the results must be defensible. Otherwise, one can always argue that the computer is no better than humans at this task. Ideally this would be done through optimality but since that is not possible with heuristic methods, all kinds of measures like compactness [47, 61], contiguity, least changes to current maps [11], partisan symmetry [30], and efficiency gap[1] [56] have also been made. The second problem is that they do not guarantee that the drawn map is completely unbiased. The achieved solution may, unintentionally, still be beneficial to a certain political party or diminish the voting power of a minority (racial gerrymandering), something called unintentional gerrymandering. Chen and Rodden defended that sometimes unintentional gerrymandering is a necessity and given the demography of certain areas, may be unavoidable [13].

3.2.2 Complete Methods

Traditionally, complete methods must consider all possibilities of redistricting (making use of *brute force* or *explicit enumeration*) in a given area and choose the best one according to the objective function to be optimized. More sophisticated solutions include the *branch and bound* and *branch and cut* algorithms or the *implicit enumeration* which are able to exclude some classes of solutions that are worse than the best solution found so far without the need to examine them. However, given the complexity of the redistricting problem (NP-Complete [2]), even the more sophisticated methods have only been used to solve small redistricting problems.

Hess was the first to study the political districting problem through a mathematical formulation [34]. In 1965 he proposed an Integer Linear Programming (ILP) model and, even though it was not implemented due to the complexity of the problem and the lack of computation power at the time, we believe it is still worth mentioning in this work given its relevance in the area. In this model n and k are, respectively, the number of TUs and the number of clusters in a territory. The idea is to identify k TUs as the centers of the k clusters and each TU must be assigned to exactly one district center. The model has n^2 binary variables $x_{i,j}$ where $i, j \in \{1, \ldots, n\}$ and $x_{i,j} = 1$ if TU i is assigned to district with center at j, and 0

[1] The Efficiency Gap measures the difference in the wasted votes – the difference of votes between the winner and the second place – between the two major parties.

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{i,j}^2 p_i x_{i,j} \tag{3.1}$$

$$\forall i \in \{1 \ldots n\} \ : \ \sum_{j=1}^{n} x_{i,j} = 1 \tag{3.2}$$

$$\sum_{j=1}^{n} x_{j,j} = k \tag{3.3}$$

$$\forall j \in \{1 \ldots n\} \ : L x_{j,j} \leq \sum_{i=1}^{n} p_i x_{i,j} \leq U x_{j,j} \tag{3.4}$$

$$\forall i, j \in \{1 \ldots n\} \ : x_{i,j} \in \{0, 1\} \tag{3.5}$$

Figure 3.1: Objective function and constraints in Hess et al. (1965) [34].

otherwise. A variable $x_{j,j}$ equals 1 if, and only if, TU j is chosen as the center of a cluster. It also takes into account $d_{i,j}$, the distance between TUs i and j, p_i, the population in i and, the lower and upper bounds on the number of people in each cluster (L and U, respectively). The objective function and the constraints are presented in Figure 3.1:

The objective function of the this model (3.1) takes into account the euclidean distance between the centers of the TUs and the population in each unit and penalizes adding TUs with large populations to a cluster where they are far away from the remaining ones. The constraints are pretty straight forward, a TU must be assigned to only one district center (3.2), there must be exactly k district centers (3.3), the population limits inside each cluster are between the lower and upper bounds (3.4) and, finally, (3.5) states that all variables are Boolean. Although it is not in the original model, some recent implementations of this model [48, 53, 58], add the following constraint (Equation 3.6) for strength, meaning that if a TU i is assigned to a district centered at j, then j must be the center of a cluster.

$$\forall i, j \in \{1 \ldots n\} \ : \ x_{i,j} \leq x_{j,j} \tag{3.6}$$

Garfinkel and Nemhauser were the first (1970) to use an exact approach for the political districting problem [29]. They use implicit enumeration techniques and the algorithm is divided in two stages. In the first one, they construct all sets of feasible districts (districts which are contiguous and meet the population requirements). In the second, they solve a set cover problem to choose the solutions that minimize the population deviation with respect to the ideal value. The authors were considering counties as the smaller geographic subdivision and could not solve the problem for states in the United States of America with more than 55 counties[2], which is the first indication of the complexity of the problem. This scalability problem may be due to the fact that in the worst case scenario the number or feasible districts grows exponentially with the number of counties.

[2]The average number of counties per state is 62.

Mehrotra et al. [45] follow a similar algorithm but suggest a different objective function that takes into account the overall compactness of the districts (and only restrict the population of each district to approximate the ideal value). The authors consider the distance between nodes to be constant and compute the diameter of each cluster[3] (essentially, the number of hops between nodes) and choose the solutions that minimize it. The algorithm is used to redistrict the state of South Carolina (51 territorial units) in 6 districts, a very small problem but the achieved results were considered to be satisfactory [45]. This approach, just like the model proposed by Hess et al. [34] (Figure 3.1) rely on objective functions that favour contiguity but do not add constraints to guarantee it. What this means is, albeit unlikely, it is entirely possible that the most compact solution according to these cost functions turns out with some of the clusters not being contiguous.

Recently, Kotthoff et al. [39] used clustering algorithms to solve this problem. They applied it to redistrict a small city in Ireland. Ireland does not use single member districts therefore the objective is to find k clusters of elected officials (represented by the matrix X), k clusters of constituencies (matrix Y) and the optimal match between them. They define the problem as a pseudo-Boolean optimization problem where the objective is to minimize the difference between the number of people that each elected official represents. A series of constraints are then added to guarantee that all districts are non-empty, that their populations meet the legal requirements (an elected official cannot represent more than 30000 people), and that they are contiguous. The full set of constraints and the objective function used to model the problem are presented in Figure 3.2 where n_1 and n_2 are, respectively, the number of districts and the number of elected officials (EO), Z is a set with the number of individuals each official represents and a_j the number of officials in constituency j. After formalizing the problem a large-scale solver (Gurobi) is used to find all solutions to the problem. Although, the authors face serious scalability problems [39].

In 2020, Validi et al. [58], in the context of the Hess model (Figure 3.1), present and test 4 different approaches to add contiguity to this model and refer to the original paper for the remaining formulations. We will focus on one interesting flow-based formulation using Boolean variables, they call it the MCF model. Remember that the Hess model uses the following n^2 binary variables:

$x_{i,j} = 1$, if vertex i is assigned to (the district centered at) vertex j

$x_{i,j} = 0$, otherwise

The authors start by creating a bi-directional version of the contiguity graph denoted $D = (V, A)$ obtained from $G = (V, E)$ and replacing each undirected edge $\{i, j\} \in E$ by its directed counterparts (i, j) and (j, i). The set of edges pointing away from vertex i is denoted by $\delta^+(i)$ while, inversely, the set of edges pointing towards vertex i is denoted $\delta^-(i)$. They propose adding the following variables $f_{i,j}^{a,b}$ where:

$f_{i,j}^{a,b} = 1$, if edge $(i, j) \in A$ is on the path to vertex a from its district center b

$f_{i,j}^{a,b} = 0$, otherwise

[3] The diameter of a network is the largest shortest distance between all pairs of nodes.

Objective:

$$min(max(Z) - min(Z))$$

$X = [x_{i,j}]_{n_1 \times k}$ and $Y = [y_{i,j}]_{n_2 \times k}$ are matrices with Boolean entries.

Subject To:

1. Bounds on the number of clusters in which an object may appear

$$\forall i \in \{1 \ldots n_1\} \ : \ min_{k_1} \leq \sum_{j=1}^{k} x_{i,j} \leq max_{k_1}$$
$$\forall i \in \{1 \ldots n_2\} \ : \ min_{k_2} \leq \sum_{j=1}^{k} y_{i,j} \leq max_{k_2}$$

2. Ensuring that each cluster is non-empty

$$\forall j \in \{1 \ldots k\} \ : \ \sum_{i=1}^{n_1} x_{i,j} \geq 1$$
$$\forall j \in \{1 \ldots k\} \ : \ \sum_{j=1}^{n_2} y_{i,j} \geq 1$$

3. The number of elected officials per cluster is between EO_{min} and EO_{max}

$$\forall j \in \{1 \ldots k\} \ : \ Let \ a_j = \sum_{i=1}^{n_2} y_{i,j}$$
$$\forall j \in \{1 \ldots k\} \ : \ EO_{min} \leq a_j \leq EO_{max}$$

4. Bound on the population represented by each elected official POP_{max}

$$\forall j \in \{1 \ldots k\} \ : \ z_j \leq POP_{max}$$

5. Ensuring that each cluster is connected

$\forall i \in \{1 \ldots n_1\}, j$ is adjacent to $i, i \neq j$:

$$(adj_i = j) \Longrightarrow ((rank_i > rank_j) \land (adj_j \neq i) \land (cidx_i = cidx_j))$$

Definitions of channel variables used above:
(a) Individuals allocated to each elected official:

$$\forall j \in \{1 \ldots k\} \ : \ Let \ z_j = \frac{\sum_{i=1}^{n_1} x_{i,j}.p_i}{a_j}$$

(b) Tree to ensure cluster connectivity

$$\forall i \in \{1 \ldots n_1\} \ : \ x_{i,cidx_i} = 1$$
$$\forall i \in \{1 \ldots n_1\} \ : \ adj_i = i \Longleftrightarrow rank_i = 0$$
$$\forall i, j \in \{1 \ldots n\}, i \neq j \ : \ ((rank_i = 0) \land (rank_j = 0)) \Longrightarrow (cidx_i \neq cidx_j)$$
$$card(rank, 0) = k$$

Figure 3.2: Objective function and constraints used by Kutthoff et al. [39] to model the problem.

$$\forall a \in V \setminus \{b\}, \forall b \in V : f^{a,b}(\delta^+(b)) - f^{a,b}(\delta^-(b)) = x_{a,b} \quad (3.7)$$

$$\forall i \in V \setminus \{a, b\}, \forall a \in V \setminus \{b\}, \forall b \in V : f^{a,b}(\delta^+(i)) - f^{a,b}(\delta^-(i)) = 0 \quad (3.8)$$

$$\forall a \in V \setminus \{b\}, \forall b \in V : f^{a,b}(\delta^-(b)) = 0 \quad (3.9)$$

$$\forall a, j \in V \setminus \{b\}, \forall b \in V : f^{a,b}(\delta^-(j)) \leq x_{j,b} \quad (3.10)$$

$$\forall (i, j) \in A, \forall a \in V \setminus \{b\}, \forall b \in V : f^{a,b}_{i,j} \in \{0, 1\} \quad (3.11)$$

Figure 3.3: Constraints defined by Validi et al. [58] to add contiguity to Hess model [34].

In practice, one can interpret the direct graph as a flow network and $f^{a,b}_{i,j}$ denotes the flow passing at edge (i, j) considering b as the source and a as the sink. Moreover, let $f^{a,b}(S)$ be a shorthand for $\sum_{(i,j)\in S} f^{a,b}_{i,j}$ for a given $S \subseteq E$.

Constraint (3.7) states that if a TU $a \in V$ has another TU $b \in V$ as its district center, then the flow coming out of b to a must equal 1 and the flow coming into b is 0 (3.9). If the flow to TU a from its center b passes through any other node then two things must happen: the flow coming into that node and going out from that node must be the same (3.8) and, that node must also have b as its district center (3.10). Finally, constraint (3.11) guarantees that all $f^{a,b}_{i,j}$ variables are Boolean.

Complete methods also have some drawbacks. First of all, they are usually even more time consuming than heuristic approaches due to the huge number of possible redistricting plans to be considered. The most recent approach by Kotthoff et al., classified the problem of redistricting Ireland (5 million people) impossible to solve in useful time without further optimization [39]. Secondly, it is important to consider that even though we achieve an optimal solution, this solution is tailored only to the exact conditions and objective functions considered, thus it is important that they are chosen fairly. Otherwise, problems of voter representation by the elected officials persist.

3.3 Additional Related Work

There is extensive work in election methods and how to condition the electoral results in different voting procedures [6, 33], as well as theoretical results on its computational complexity [8, 22, 23, 24, 41].

Single and multi-objective optimization has been used in many different domains where one wants to define districts or partition a given territory. For instance, multi-objective optimization has been used to define districts for public transportation in the Paris area [57], healthcare administrative authorities over geographic areas in England [20], or census units in Canada [19].

The management of the forests and ecological areas also has an extensive work on partitioning with constraints similar to those to the electoral districting problem. For instance, several harvest scheduling in agriculture areas [14, 43, 46] or forest planning [12] also deal with contiguity constraints. The same also occurs in defining reserved areas for species [49, 50]. However, the models proposed suffer from the same drawbacks as the ones proposed in the previous section.

Chapter 4

Multi-Objective Combinatorial Optimization Model

In this chapter we present our new formulation for the PD problem as a MOCO problem and explain how we can guarantee that our solutions (electoral maps) possess the three main characteristics that are essential to good electoral maps: similar popular representation (Section 4.1), district contiguity (Sections 4.3) and district compactness (Section 4.2). Afterwards, it is made clear how to gerrymander electoral maps by maximizing or minimizing the electoral results of a party (Section 4.4). In this work, given the importance of elections for a democracy, we are mostly interested in complete methods which are capable of finding global optimum solutions, although a faster but incomplete model is presented in Section 4.3.2. Finally, optimizations to our model which can also be adapted to previous approaches are proposed in Section 4.5 and, in Section 4.6, we discuss the asymptotic growth of the size of the proposed formulations.

Like many authors before [4, 10, 31, 39, 45, 58] we rely on a contiguity graph of the Portuguese territory. A contiguity graph (or population graph) is an n-node connected weighted graph $G = (N, E)$ where the territorial units being considered correspond to the N nodes and an edge $(u, v) \in E$ between two nodes exists if, an only if, nodes u and v are neighbors (share a border).

Each edge $(u, v) \in E$ has an associated weight. In our case, the length of the border (in meters) shared between nodes u and v. On the other hand, each node $n \in N$ has associated multiple values: the number of individuals registered to vote inside the territorial unit and the number of votes received in the last election by each of the main Portuguese parties.

Let $\mathcal{K}$ denote the set of clusters, i.e., electoral districts (ED) to be created, numbered from 1 to k. That is equivalent to the number of officials we want to elect, since we are considering a system with single-member districts where each ED elects exactly one official. In this work the terms cluster and electoral district are used interchangeably.

Let the set of nodes N denote the set of territorial units (TU) numbered from 1 to n inside the area we wish to redistrict.

Let X be the k x n binary matrix that determines which territorial units are assigned to each cluster.

This is achieved by each line X_k being a binary vector (bit array) where a 1 means that that TU is part of cluster k and a 0 that it is not.

Formally:

$$x_{i,j} = 1, \text{ if cluster } i \text{ contains territorial unit } j$$
$$x_{i,j} = 0, \text{ otherwise}$$

The most trivial constraints are guaranteeing that each cluster $k \in \mathcal{K}$ is composed of at least one TU (no empty clusters – Equation 4.1) and each TU is used once, and only once, throughout all clusters (Equation 4.2). Therefore, the following PB constraints need to be defined:

$$\forall i \in \mathcal{K}: \sum_{j \in N} x_{i,j} \geq 1 \tag{4.1}$$

$$\forall j \in N: \sum_{i \in \mathcal{K}} x_{i,j} = 1 \tag{4.2}$$

This base formulation is similar to the one proposed by Kotthoff et al. [39] and we believe it is superior to the classical formulation of the PD problem by Hess et al. [34] because to match TUs with EDs it requires $n \times k$ variables instead of $n \times n$ variables and k is usually far smaller than n.

4.1 Equal Popular Representation

Ideally each individual would have exactly the same voting power, consequently each elected official should represent exactly the same number of voters. Let us call that perfect value B. The value of B is defined by the total number of individuals divided by the number of officials to be elected (which is equal to the number of electoral districts, in a FPTP system).

It is impossible to guarantee that each ED contains exactly B voters. Although, if we take into account R_j – the number of people registered to vote in each TU $j \in N$ – and a constant ε between 0 and 1, we can define the following constraint (4.3) that limits the number of people each elected official represents to a percentage around B.

$$\forall i \in \mathcal{K}: B(1-\varepsilon) \leq \sum_{j \in N} R_j x_{i,j} \leq B(1+\varepsilon) \tag{4.3}$$

Note that 4.3 is not a normalized Pseudo-Boolean (PB) constraint so we have to separate it into a lower bound and an upper bound (4.4 and 4.5, respectively).

$$\forall i \in \mathcal{K}: \sum_{j \in N} R_j x_{i,j} \geq B(1-\varepsilon) \tag{4.4}$$

$$\forall i \in \mathcal{K}: \sum_{j \in N} R_j x_{i,j} \leq B(1+\varepsilon) \tag{4.5}$$

4.2 District Compactness

Another crucial characteristic of good electoral maps is district compactness. This means that the electoral districts should avoid odd shapes like the one in Figure 2.3. In order to tackle this problem, we propose the maximization of the border length between territorial units in the same cluster as an objective function. Observe that this is equivalent to minimizing the size of the frontier between TUs in different clusters. This is an innovative approach to create compact electoral districts that we believe can yield good results efficiently and that is confirmed by the experimental results. Additionally, using real-word maps, the length of the borders is a measure easy to interpret and visualize.

Let N_j denote the set of TUs that are neighbors of TU j. If two TUs j and j' are neighbors, then $j \in N_{j'}$ and $j' \in N_j$. Let $L_{j,j'}$ be the border length between neighboring TUs $(j, j') \in N$. This value can be obtained through the analysis of the area to be redistricted using a geographic information system (GIS) software, such as ArcGIS[1] or QGIS[2].

Let $b_{i,j,j'}$ be a Boolean variable which indicates if TU j and TU $j' \in N_j$ are both in the same cluster i. Note that we only need to create such variable b between neighboring territorial units since there is no border between territorial units that are not adjacent and we do not need to create $b_{i,j',j}$ because $b_{i,j,j'} = b_{i,j',j}$.

Since our objective is to increase the compactness of our electoral districts we can add the following objective function maximizing the sum of the border length between TUs in the same cluster:

$$\text{Maximize} \sum_{i\in\mathcal{K}} \sum_{j\in N} \sum_{j'\in N_j} L_{j,j'} b_{i,j,j'} \tag{4.6}$$

We must ensure that the variable $b_{i,j,j'} = 1$ if, and only if, TU j and TU $j' \in N_j$ are both in cluster i, and $b_{i,j,j'} = 0$ otherwise. Hence, we must add the following constraint:

$$\forall i \in \mathcal{K},\ j \in N, j' \in N_j :\ \neg\,(x_{i,j} \wedge x_{i,j'}) \implies \neg\, b_{i,j,j'} \tag{4.7}$$

This constraint guarantees that variable $b_{i,j,j'} = 0$ if TU j or TU $j' \in N_j$ (or both) are not in cluster i and that would be enough to guarantee the correct objective value since we are upon a maximizing objective function and $b_{i,j,j'}$ would be equal to one whenever that is possible (all cases where Equation 4.7 does not set it to 0). However, when using constraint solvers, such as Sat4jmoco, we can significantly decrease the execution time, i.e., increase the efficiency if we also define the constraint 4.8 forcing $b_{i,j,j'}$ to equal 1 whenever neighboring TUs j and j' are both in cluster i. Running a small test in a territory with $k = 5$, $n = 16$ and $\varepsilon = 0.2$ this optimization proved successful, managing to decrease the execution time from 8.11 seconds to 1.15 seconds.

$$\forall i \in \mathcal{K},\ j \in N, j' \in N_j :\ (x_{i,j} \wedge x_{i,j'}) \implies b_{i,j,j'} \tag{4.8}$$

[1] `https://www.esri.com/en-us/arcgis/about-arcgis/overview`, accessed 22/12/2020.
[2] `https://qgis.org/en/site/`, accessed 22/12/2020.

Again, both Equation 4.7 and Equation 4.8 are not PB constraints meaning we have to normalize them as follows:

$$\begin{aligned}\forall i \in \mathcal{K},\ j \in N, j' \in N_j : x_{i,j} - b_{i,j,j'} &\geq 0 \\ x_{i,j'} - b_{i,j,j'} &\geq 0\end{aligned} \tag{4.9}$$

$$\forall i \in \mathcal{K},\ j \in N, j' \in N_j :\ -x_{i,j} - x_{i,j'} + b_{i,j,j'} \geq -1 \tag{4.10}$$

We note that if we consider the Boolean variables $b_{j,j'}$, with the same meaning as the variables $b_{i,j,j'}$ described before. It is also possible to create an equivalent objective function to (4.6) with these variables that will always yield the same results:

$$\text{Maximize} \sum_{j \in N} \sum_{j' \in N_j} L_{j,j'} b_{j,j'} \tag{4.11}$$

We only need to adapt Equations 4.7 and 4.8 to accommodate this new variables which can be done as follows:

$$\forall i \in \mathcal{K}, \forall i' \in \mathcal{K} \setminus \{i\},\ j \in N, j' \in N_j :\ (x_{i,j} \wedge x_{i',j'}) \implies \neg\, b_{j,j'} \tag{4.12}$$

$$\forall i \in \mathcal{K},\ j \in N, j' \in N_j :\ (x_{i,j} \wedge x_{i,j'}) \implies b_{j,j'} \tag{4.13}$$

It is worth noting that this implementation only creates $O(n \times \langle k \rangle)$ variables to be optimized instead of $O(k \times n \times \langle k \rangle)$ where $\langle k \rangle$ is the average degree of a node in our graph, i.e., average number of neighbors. Although, it requires more constraints since Equation 4.12 creates k times more constraints than Equation 4.7. For this reason and because the solver can infer Equation 4.14 the performances of both implementations are similar.

$$\forall j \in N, j' \in N_j :\ \sum_{i \in \mathcal{K}} b_{i,j,j'} <= 1 \tag{4.14}$$

Similarly to Figure 3.1 (for the Hess model), Figure 4.1 contains the base formulation and succinctly explains the necessary constraints in our proposed model.

4.3 District Contiguity

The last fundamental quality in electoral maps is contiguity. An electoral district is contiguous if, and only if, any pair of territorial units $(j, j') \in N$ that belong to the same cluster $i \in \mathcal{K}$ are either adjacent (neighbors) or can be connected using other TUs that are also part of cluster i. What this means is either j and j' are neighbors or there must exist a path between them passing only through other TUs inside cluster i.

In Section 4.3.1 and Section 4.3.2 we present two formulation to assure map contiguity, a tree-based

Objective:

- Maximizing compactness:

$$Maximize \sum_{i \in \mathcal{K}} \sum_{j \in N} \sum_{j' \in N_j} L_{j,j'} b_{i,j,j'}$$

Subject To:

1. Constraint to ensure each cluster has at least one TU:

$$\forall i \in \mathcal{K} \ : \ \sum_{j=1}^{n} x_{i,j} \geq 1$$

2. Each TU is assigned to 1, and only 1, cluster:

$$\forall j \in N \ : \ \sum_{i=1}^{k} x_{i,j} = 1$$

3. Each elected official represents approximately the same number of people:

$$\forall i \in \mathcal{K} \ : \ \sum_{j=1}^{n} R_j x_{i,j} \geq B(1-\varepsilon)$$

$$\forall i \in \mathcal{K} \ : \ \sum_{j=1}^{n} R_j x_{i,j} \leq B(1+\varepsilon)$$

4. Constraint to make $b_{i,j,j'} = 0$ if TU j or its neighbouring TU j' are not in cluster i:

$$\forall i \in \mathcal{K}, \ j \in N, j' \in N_j : \ \neg \left(x_{i,j} \wedge x_{i,j'}\right) \implies \neg \, b_{i,j,j'}$$

   - Optimization for constraint solvers that sets $b_{i,j,j'} = 1$ in the remaining cases:

$$\forall i \in \mathcal{K}, \ j \in N, j' \in N_j : \ \left(x_{i,j} \wedge x_{i,j'}\right) \implies b_{i,j,j'}$$

Figure 4.1: Base model with summary of constraints. Contiguity constraints in Figure 4.3.

formulation (complete) and a shortest-path formulation (incomplete). Finally, Figure 4.3 summarizes both formulations.

4.3.1 Tree-based Contiguity Formulation

Beyond the aforementioned Boolean variables, in order to guarantee contiguity, we extend the base formulation (Figure 4.1) with the following new sets of variables:

- r_j denotes if territorial unit $j \in N$ is the root of a cluster.
- $p_{j,j'}$ denotes if territorial unit $j' \in N_j$ is predecessor of TU $j \in N$
- $d_{j,l}$ denotes if territorial unit $j \in N$ is at depth l in the tree

Variables $x_{i,j}$ represent the distribution of TUs to the clusters, while variables $p_{j,j'}$ represent the connectivity of each cluster. Observe that only neighbors can be considered as predecessors of a given territorial unit j in variables $p_{j,j'}$. Since each cluster must be a set of contiguous TUs, then one can build a tree to represent the connectivity of that set. In order to represent the tree, one only needs to define the predecessor of each node and its depth ($d_{j,l}$) in the tree (equivalent to the distance of a TU to the root of its cluster). The root of a cluster, r_j, has no predecessor and a depth equal to 0.

Let $\mathcal{M} = \{0, \ldots, m\}$ denote the set of possible depths in a cluster tree where m is the maximum number of TUs to be assigned to a cluster minus 1. Considering there are limits on the number of voters in each cluster, one can a priori calculate the value of m (more details on how to calculate this value in Section 4.5.3).

With these variables in place we are now in conditions of creating the following set of constraints:

- The number of roots must be exactly the same number as the number of clusters (EDs).

$$\sum_{j \in N} r_j = k \tag{4.15}$$

- In each cluster, there can only be one root node. Hence, if the cluster already has a root, then no other node in the cluster can be root.

$$\forall i \in \mathcal{K}, j, j' \in N, j \neq j' : (x_{i,j} \wedge x_{i,j'} \wedge r_j) \implies \neg r_{j'} \tag{4.16}$$

  This constraint can be normalized as follows:

$$\forall i \in \mathcal{K}, j, j' \in N, j \neq j' : -x_{i,j} - x_{i,j'} - r_j - r_{j'} \geq -3 \tag{4.17}$$

  Meaning that if both TU $j \in N$ and its neighboring TU j' belong to cluster i then at most one of them can be the root of said cluster.

- Each TU must have a neighbor (and only one) as predecessor in the tree that represents a given cluster or be the root of the cluster itself.

$$\forall j \in N : \left(\sum_{j' \in N_j} p_{j,j'} \right) + r_j = 1 \tag{4.18}$$

- If two neighbors belong to different clusters, then they cannot have the predecessor relation.

$$\forall i \in \mathcal{K}, j \in N, j' \in N_j : (x_{i,j} \wedge \neg x_{i,j'}) \implies \neg p_{j',j} \tag{4.19}$$

  Which can be normalized as:

$$\forall i \in \mathcal{K}, j \in N, j' \in N_j : -x_{i,j} + x_{i,j'} - p_{j',j} \geq -1 \tag{4.20}$$

- For each pair of neighbors, the predecessor relation can only work in one direction. This means that either TU $j \in N$ is predecessor of TU $j' \in N_j$, TU j' is predecessor of TU j or they do not have the predecessor relation at all.

$$\forall j \in N, j' \in N_j : p_{j,j'} + p_{j',j} \leq 1 \tag{4.21}$$

- Each TU can only be assigned a depth.

$$\forall j \in N : \sum_{i=1}^{m} d_{j,i} = 1 \tag{4.22}$$

- The root node is at depth 0.

$$\forall j \in N : \ r_j \implies d_{j,0} \tag{4.23}$$

Converting to PB constraints we get:

$$\forall j \in N : \ -r_j + d_{j,0} \geq 0 \tag{4.24}$$

- The depth of a given node is one more than its predecessor.

$$\forall j \in N, j' \in N_j, l \in \mathcal{M} \setminus \{m\} : \ (p_{j,j'} \wedge d_{j',l}) \implies d_{j,l+1} \tag{4.25}$$

This constraint can be normalized as follows:

$$\forall j \in N, j' \in N_j, l \in \mathcal{M} \setminus \{m\} : \ -p_{j,j'} - d_{j',l} + d_{j,l+1} \geq -1 \tag{4.26}$$

- Finally, the nodes at depth m must be leafs of the tree, i.e. these nodes cannot be predecessors of any other nodes.

$$\forall j \in N, j' \in N_j : \ d_{j,m} \implies \neg p_{j',j} \tag{4.27}$$

Which can be normalized as:

$$\forall j \in N, j' \in N_j : \ -d_{j,m} - p_{j',j} \geq -1 \tag{4.28}$$

4.3.2 Shortest-Path Contiguity Formulation

This section presents the first formulation we developed in an attempt to guarantee contiguous districts. It turns out that although contiguity is guaranteed, this formulation is incomplete meaning that, occasionally, some feasible solutions are not considered. However, we still feel that it is relevant and worth exploring in this work because it is innovative, extremely fast due to the lower number of variables required and it is seldom that the ignored solutions belong to the Pareto front.

This formulation shares every aspect of the base formulation previously described (Figure 4.1) and creates contiguous districts by exploiting the fact that in any contiguous district i, using only territorial units belonging to i, there must exist a path between any pair of territorial units.

Consider the contiguity graph of the territorial units where a weight of 1 is defined for every edge between two adjacent territorial units. Given this weighted graph, one can easily compute the matrix D of shortest distances between all pairs of nodes (TUs) in polynomial time. An option would be using the Floyd-Warshall algorithm [15] (complexity of $O(n^3)$ where n is the number of TUs).

With that information we can create Equation 4.29 where, for all clusters, if two non-neighboring TUs j and j' belong to the same cluster i then, there must exist at least another TU j'' in that same cluster i so that j'' is at the same time neighbor of j' and the distance between TU j and TU j'' is lower than the distance between TU j and TU j' ($D_{j,j''} < D_{j,j'}$). Formally, a shortest path between any two non-neighboring TUs belonging to the same cluster i must be exist inside i.

$$\forall i \in \mathcal{K}, j, j' \in N, j \neq j', j' \notin N_j : \ (x_{i,j} \wedge x_{i,j'}) \implies \left(\bigvee_{j'' \in N_{j'}, D_{j,j''} < D_{j,j'}} (x_{i,j''}) \right) \tag{4.29}$$

Equation 4.29 can be converted into a PB constraint as follows:

$$\forall i \in \mathcal{K}, j, j' \in N, j \neq j', j' \notin N_j : \ -x_{i,j} - x_{i,j'} + \sum_{j'' \in N_{j'}, D_{j,j''} < D_{j,j'}} (x_{i,j''}) \geq -1 \tag{4.30}$$

Note that the constraint is trivially satisfied if only TU $j \in N$ or its neighbor TU j' (or none) are present in cluster i. Otherwise, at least one (but possibly multiple) other TUs $x_{i,j''}$ meeting the requirements ($j'' \in$ N$_{j'} \wedge D_{j,j''} < D_{j,j'}$) must also be in cluster i.

This formulation is incomplete because by computing the shortest distances between all TUs, only paths between TUs with that length are allowed (any shortest path). Formally, if two non adjacent TUs j and j' are in the same cluster i then all TUs that make up one shortest path between j and j' must also be in cluster i. Usually, it is extremely beneficial to maximize the compactness of the solutions to only include shortest paths between territorial units and, for that reason, it is rare that the most compact solution (according to the objective function defined) does not satisfy the shortest-path constraint. However, in Figure 4.2 we present one of those scenarios. Obviously, these situations are more frequent if we are considering an area with more territorial units or maximizing compactness is not the only objective function.

Note how, using the shortest-path formulation (left map on Figure 4.2), TUs A and B could not be in the same cluster because the distance between those TUs is 2 (passing through C). Therefore, Equation 4.29 would require at least another TU at distance 1 from A and B to also be included in that cluster and C is the only TU meeting those requirements, since it is the only one that is both neighbor of A and B. However, in this context, since C is a high population density unit, this is not feasible and the optimal solution (right map in Figure 4.2) is not considered.

The border length between territorial units in the same cluster (our compactness measure) is increased from 584402 meters using the shortest-path formulation to 614891 meters using the tree-based

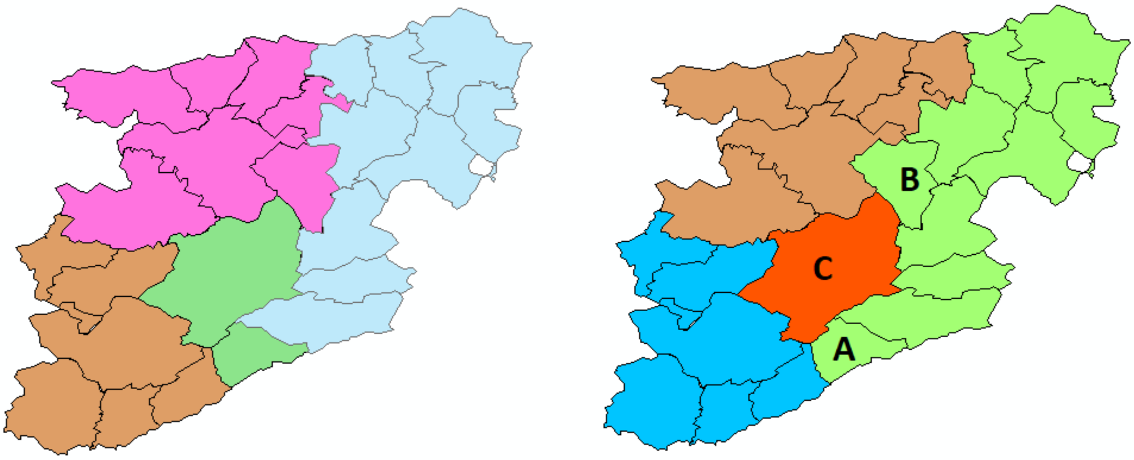


Figure 4.2: Redistricting the same area with the shortest-path constraints (left map) and the tree-based constraints (right).

formulation (right map) which represents a 5.22% increase (approximately). This a modest increase and that explains why visually it is hard to tell which solution is the most compact one.

The fact that the shortest-path formulation is simple, really efficient, usually does not preclude the global optimal solution and even in those cases manages to find pretty compact solutions, makes it a great option to use in scenarios where a complete model requires a much larger computational time. Since this incomplete contiguity formulation does not require any additional variables to guarantee contiguity (all the variables defined in Section 4.3.1) the formulas to be generated are much smaller. In Section 4.6 we compare the number of variables created by this contiguity formulation with the ones created by the tree-based formulation which helps explain its speed.

4.4 Maximizing/Minimizing Party Results

The objective function proposed in Section 4.2 is solely concerned with optimizing compactness. However, it is often the case that gerrymandering can be used to produce electoral maps beneficial towards some political party (or being less advantageous to some other party). In this section, we extend the model by considering which party is likely to win each electoral district in a first-past-the-post (FPTP) electoral system based on results from previous elections.

Let P denote the set of political parties running for the election. Let $v_{p,j}$ be the likely number of votes party $p \in P$ receives in TU $j \in N$. This value can be defined for all parties and TUs through an analysis of historical data in previous elections. Let $l_{i,p,p'}$ be a Boolean variable that denotes if political party p is likely to lose the election at ED (cluster) i to party p'.

For each political party p, the following set of constraints must be added in order to determine if a given party p has fewer votes than another party p':

Contiguity Constraints:

- Tree-based Formulation (Complete):
  1. The number of roots must be exactly the same number as the number of clusters (k):

     $$\sum_{j \in N} r_j = k$$

  2. If a cluster already has a TU has root, no other TU in the same cluster can be a root:

     $$\forall i \in \mathcal{K}, j, j' \in N, j \neq j' : (x_{i,j} \wedge x_{i,j'} \wedge r_j) \implies \neg r_{j'}$$

  3. Each TU j is either the root of a cluster or has a neighbouring TU j' as predecessor:

     $$\forall j \in N : \left(\sum_{j' \in N_j} p_{j,j'} \right) + r_j = 1$$

  4. If neighbouring TUs belong to different clusters, they cannot have the predecessor relation:

     $$\forall i \in \mathcal{K}, j \in N, j' \in N_j : (x_{i,j} \wedge \neg x_{i,j'}) \implies \neg p_{j',j}$$

  5. The predecessor relation is unidirectional, for each pair of neighbours:

     $$\forall j \in N, j' \in N_j : p_{j,j'} + p_{j',j} \leq 1$$

  6. Constraint to assign exactly one depth to each TU:

     $$\forall j \in N : \sum_{i=1}^{m} d_{j,i} = 1$$

  7. The root node is at depth 0:

     $$\forall j \in N : \; r_j \implies d_{j,0}$$

  8. The depth of a given node is one more than its predecessor:

     $$\forall j \in N, j' \in N_j, l \in \mathcal{M} \setminus \{m\} : \; (p_{j,j'} \wedge d_{j',l}) \implies d_{j,l+1}$$

  9. Nodes at final depth (m) cannot be predecessors of any other nodes:

     $$\forall j \in N, j' \in N_j : \; d_{j,m} \implies \neg p_{j',j}$$

- Shortest-Path Formulation (Incomplete):
  1. If two non-neighbouring TUs $x_{i,j}$ and $x_{i,j'}$ belong to the same cluster then, there must exist at least another TU $x_{i,j''}$ (in the same cluster i) so that $x_{i,j''}$ is at the same time neighbour of $x_{i,j'}$ and the distance (number of edges) between $x_{i,j}$ and $x_{i,j''}$ is lower than the distance between $x_{i,j}$ and $x_{i,j'}$ ($D_{j,j''} < D_{j,j'}$):

     $$\forall i \in \mathcal{K}, j, j' \in N, j \neq j', j' \notin N_j : \; (x_{i,j} \wedge x_{i,j'}) \implies \left(\bigvee_{j'' \in N_{j'}, D_{j,j''} < D_{j,j'}} (x_{i,j''}) \right)$$

Figure 4.3: Summary of contiguity constraints for the tree-based formulation (Section 4.3.1) and the shortest-path formulation (Section 4.3.2). These extend the base model in Figure 4.1.

$$\forall i \in \mathcal{K},\ \forall p, p' \in P, p \neq p' : \left(\sum_{j \in N} v_{p',j} x_{i,j} - \sum_{j \in N} v_{p,j} x_{i,j} \geq 1 \right) \iff l_{i,p,p'} \tag{4.31}$$

Note that the vote difference between party p and p' at TU j can be computed while building the formula. Hence, a more compact formulation for these constraints would be:

$$\forall i \in \mathcal{K},\ \forall p, p' \in P, p \neq p' : \left(\sum_{j \in N} (v_{p',j} - v_{p,j}) x_{i,j} \geq 1 \right) \iff l_{i,p,p'} \tag{4.32}$$

Finally, note that constraints (4.31 and 4.32) are not PB constraints. However, these can be easily converted into PB constraints as follows:

$$\forall i \in \mathcal{K}, \forall p, p' \in P, p \neq p' : K_{i,p,p'} l_{i,p,p'} + \sum_{j \in N} (v_{p,j} - v_{p',j}) x_{i,j} \geq 1 \tag{4.33}$$

$$\forall i \in \mathcal{K}, \forall p, p' \in P, p \neq p' : - K_{i,p,p'} l_{i,p,p'} + \sum_{j \in N} (v_{p',j} - v_{p,j}) x_{i,j} \geq 1 - K_{i,p,p'} \tag{4.34}$$

where for each constraint we have $K_{i,p,p'} > \sum_{j \in N} |v_{p',j} - v_{p,j}|$. In this case, since the number of registered voters plus 1 is always greater than $|v_{p',j} - v_{p,j}|$ we can safely use $K_{i,p,p'} = \sum_{j \in N} R_j + 1$.

Observe that if $l_{i,p,p'} = 1$, then the first constraint is trivially satisfied since $K_{i,p,p'}$ is always larger than the sum of the remaining literals. On the other hand, if $l_{i,p,p'}$ is assigned value 0, then we must have $\sum_{j \in N} v_{p,j} x_{i,j} > \sum_{j \in N} v_{p',j} x_{i,j}$, i.e., party p wins more votes than party p' in cluster i. The second constraint works similarly, ensuring the equivalence in Equation 4.32.

Up to this point we are considering a party p the winner against another party p' if party p gets at least one more vote than p'. Although we may also require that p receives a certain margin, M (between 0 and 1), of votes over p' to consider p the winner and update variable $l_{i,p',p}$ to 1. This is particularly useful if we want to make a distinction between electoral districts that are certain wins for a given party and toss-up districts, i.e., competitive districts where either party can be the most voted.

In order to add this functionality, we simply need to slightly modify Equation 4.32 and subtract to the vote difference between parties p and p' the margin M of all votes cast to parties $p \in P$. This can also be done while building the formula:

$$\forall i \in \mathcal{K},\ \forall p, p' \in P, p \neq p' : \left(\left(\sum_{j \in N} (v_{p',j} - v_{p,j}) - \sum_{p \in P} v_{p,j} M \right) x_{i,j} \geq 1 \right) \iff l_{i,p,p'} \tag{4.35}$$

Notice that if we have $M = 0$ then Equations 4.32 and 4.35 are equal.

Let $l_{i,p}$ be a Boolean variable that denotes if political party p is likely to lose the election at ED i against at least one of the other parties. Hence, $l_{i,p}$ can be determined as:

$$\forall i \in \mathcal{K},\ \forall p \in P : l_{i,p} \iff \bigvee_{p' \in P, p \neq p'} l_{i,p,p'} \tag{4.36}$$

Again, note that Equation 4.36 is not a PB constraint. We must convert it into the following ones:

$$\forall i \in \mathcal{K},\ \forall p \in P,\ p' \neq p : -l_{i,p} + \sum_{p \in P} l_{i,p,p'} \geq 0 \tag{4.37}$$

$$\forall i \in \mathcal{K}, \forall p \in P,\ p' \neq p : l_{i,p} - l_{i,p,p'} \geq 0 \tag{4.38}$$

If we have $l_{i,p} = 1$ it means (from Equation 4.37) that at least one other party must win against party p in cluster i. And from Equation 4.38 we have that if party p loses against any party in cluster i then it does not win the elections in cluster i ($l_{i,p} = 1$).

Therefore, in order to maximize the number of elected members for party p, one can define the following goal, that minimizes the number of ED losses:

$$\text{Minimize } \sum_{i \in \mathcal{K}} l_{i,p} \tag{4.39}$$

Observe that one such objectives can be added for any different party $p \in P$.

Analogously, if we want to minimize the number of officials elected by a certain party p, we can define the opposite objective function that maximizes the number of EDs party p loses:

$$\text{Maximize } \sum_{i \in \mathcal{K}} l_{i,p} \tag{4.40}$$

4.5 Optimizations

In this section we focus on optimizations that can be added to the proposed model in order to increase its performance. Whenever possible, we also adapt these optimizations to the Hess model [34], a classical approach to the political districting problem (described in Section 3.2.2 and Figure 3.1).

4.5.1 Adding Symmetry Constraints Between Clusters

Symmetry constraints are simply additional constraints that allow to remove equivalent solutions [60]. This is known to be very effective in some domains, but it depends on the pruning capacity and the extra effort necessary to maintain these constraints, hence, in some cases, it might not be helpful [36, 44].

Consider four TUs A, B, C, D and two clusters. Suppose that a solution is reached where A, B form cluster 1 and C, D form cluster 2. Notice that if we were to assign A, B to cluster 2 and C, D to cluster 1, then we would have essentially the same solution. In this case, we say that these two solutions are symmetric to the cluster assignment.

In order to avoid these situations, one can add constraints that cut these symmetric assignments, just allowing one of them to be a solution to our model. Observe that the model remains valid, since we are only cutting symmetric solutions. In our domain, we can add constraints such that the number of voters in each cluster is non-decreasing according to the cluster identifier. Hence, let R_j be the number of registered voters in TU j. Then, we can add the following constraints.

$$\forall i \in \mathcal{K} \setminus \{k\} : \sum_{j \in N} R_j \cdot x_{i,j} \leq \sum_{j \in N} R_j \cdot x_{i+1,j} \tag{4.41}$$

4.5.2 Adding Symmetry Constraints Inside Clusters

In the previous section we added constraints to cut symmetries between clusters. However, when using the tree-based contiguity formulation there are also symmetries inside clusters. Consider the same four TUs A, B, C, D to be divided between the same two clusters. It is equivalent assigning A and B to cluster 1 and C and D to cluster 2 with A and C as roots or the same assignment with B and D as roots and A and C as leaves.

A possible solution to avoid these scenarios is to add a constraint that would only allow the TU with the lowest ID in the cluster to be the root. It can be defined as follows:

$$\forall i \in \mathcal{K}, \forall j \in N, j' \in N_j, j < j' : \ p_{j,j'} \implies \neg r_{j'} \tag{4.42}$$

Under constraint 4.42 if TU j is preceded by a neighbor j' with higher ID then it implies that TU j' is not the root of any cluster, essentially forcing j to be the root of the cluster containing both TUs.

Equation 4.42 can be easily adapted to the Hess model [34] (described in Section 3.2.1 and Figure 3.1) in order to cut the same type of symmetries and significantly increase the performance.

$$\forall j, j' \in V, j < j' : \ x_{j,j'} \implies \neg x_{j',j'} \tag{4.43}$$

It is important to point out that this specific optimization is only possible because the objective function used for compactness (Section 4.2) does not depend on the center (root) of the cluster, allowing any territorial unit to be the root. This is not the case in many previous approaches to the political districting problem [53], including in the original Hess model [34] (Figure 3.1), meaning that Equation 4.43 is only valid using another objective function.

4.5.3 Reducing Maximum Depth

The tree-based contiguity formulation (Section 4.3.1) defines each cluster as a tree with depth m. A trivial upper bound on the maximum depth of each tree that represents a cluster is to define $m = n - k$. In this worst case, there is a cluster with m TUs, while the remaining $k - 1$ clusters only have one TU. However, this is only expected to occur if there are TUs with enough voters to be a cluster on its own. Otherwise, tighter bounds can be defined.

Another trivial approach is to consider that the cluster with most TUs is composed of the TUs with fewer voters. Let U denote the maximum number of voters to be assigned to a cluster and let R_j denote the number of people registered to vote in TU j. Furthermore, let O define the ordered list of the n TUs indexes in a non-decreasing order of voters. Therefore, one can define m as follows:

$$m = max\{u : \sum_{j=1}^{u} R_{O[j]} \leq U\} \tag{4.44}$$

Clearly, there can not exist a cluster with $u + 1$ TUs, since it would not satisfy constraint (4.5). However, a limitation of this method is that the TUs with fewer voters might not be contiguous.

In the worst case, the units form a graph that corresponds to a linked list. Notwithstanding, in this case, if we consider the root of a cluster to be a node in the middle of the list, there is still always a solution since our constraints defined in Section 4.3.1 do not preclude it. Hence, we can safely divide m by 2 without removing any feasible clusters.

4.5.4 Removing Impossible Pairs

Drawing from Validi et al. [58], it is also possible to determine that some pairs of TUs cannot be in the same cluster. Consider the following weighted directed graph $G_w = (N, E)$ defined as follows:

- For each TU $j \in N$ there is a corresponding node $j \in N$
- For each pair of adjacent TUs j and j', we define two edges (j, j') and (j', j) in E such that $w(j, j') = R_{j'}$ and $w(j', j) = R_j$

Let U denote the upper population limit in an ED and $\delta(j, l)$ denote the shortest path from j to l in G_w. If $R_j + \delta(j, l) > U$, then TUs j and l cannot be in the same cluster. Note that it would require more voters than the upper limit U in order for these two TUs to be in the same contiguous cluster. Hence, any of the following constraints could be safely added to our model:

$$\forall i \in \mathcal{K} : x_{i,j} \implies \neg x_{i,l} \tag{4.45}$$

$$\forall i \in \mathcal{K} : x_{i,l} \implies \neg x_{i,j} \tag{4.46}$$

Which translate to the PB constraint:

$$\forall i \in \mathcal{K} : -x_{i,j} - x_{i,l} \geq -1 \tag{4.47}$$

The same principle can also be applied in the Hess model, resulting in Equation 4.48 requiring TUs j and l to always be in different clusters.

$$\forall i \in N : x_{j,i} \implies \neg x_{l,i} \tag{4.48}$$

4.6 Complexity

In this section we dive deeper into the complexity of both the complete and incomplete models based on the number of variables they require which is related with the search space that needs to be explored. We start with the base model and then explain the variables that need to be added in the tree-based contiguity version, estimating an approximate value and providing an upper bound on the total number of variables for each case.

The number of $x_{i,j}$ variables is $O(k \times n)$ with $2 \leq k < n$, but we expect k to be much smaller than n. The number of $b_{i,j,j'}$ variables is $O(k \times n \times \langle k \rangle)$ where $\langle k \rangle$ is the average degree of a node in our graph, i.e., average number of neighbors. In the worst case scenario $\langle k \rangle = n - 1$ because all nodes connect to every other node, although we only need to consider the connection in one direction since $b_{i,j,j'} = b_{i,j',j}$. However, we found that in real-world scenarios we have small values of $\langle k \rangle$, 5.31 in graphs of Portuguese parishes and 3.74 in graphs of Portuguese municipalities. Hence, we can expect the number of variables in the incomplete model to be around $k \times n + k \times n \times \langle k \rangle \div 2$ (the number of $x_{i,j}$ variables plus the number of $b_{i,j,j'}$ variables, respectively). In the absolute worst case scenario, we can guarantee that the number of variables is bounded by $O(n^3)$.

Using the tree-based contiguity formulation (complete), beyond those variables we also have to consider the number of $p_{j,j'}$, r_j and $d_{j,l}$ variables. The number of $p_{j,j'}$ variables is $O(n \times \langle k \rangle)$, the number of nodes times the average degree of a node. The number of r_j variables is simply n, one per TU. Finally, the number of $d_{j,l}$ variables is $O(n \times m)$. In the worst case, we have $m = (n - k)/2$ resulting in $(n^2 - nk)/2$ variables, but one can expect m to be much smaller, since the number of voters is not usually so concentrated. As a result, in the worst case scenario, the number of variables in the complete model is also always bounded by $O(n^3)$.

However, in both contiguity formulations we strongly expect the number of variables to be much lower than $O(n^3)$ since k and $\langle k \rangle$ will be much lower values than n. Our data shows that even the tree-based formulation usually grows as n^2.

Finally, if we are also maximizing or minimizing the results of a political party we also need to add the variables $l_{i,p}$ and $l_{i,p,p'}$ to the model. Note that these variables depend on the number of parties P being considered. The number of $l_{i,p}$ is $n \times P$ and the number of $l_{i,p,p'}$ is $n \times P^2$. In the end, since the value P is typically really small (≤ 4, in our particular case) the total number of variables is not strongly impacted.

On the number of constraints, for both cases, it is also bounded by $O(n^3)$ because Equation 4.7 requires $O(k \times n \times \langle k \rangle)$ constraints. Additionally, for contiguity, in the shortest-path method, Equation 4.29 creates $O(k \times n^2)$ new constraints. Meanwhile, in the tree-based method, Equation 4.16 introduces $O(k \times n^2)$ constraints, Equation 4.19 creates $O(k \times n \times \langle k \rangle)$ and Equation 4.25 introduces $O(n \times \langle k \rangle \times m)$ more constraints where m is the maximum depth. These are all bounded by $O(n^3)$. However, once again, because k and $\langle k \rangle$ will be much lower values than n we expect the number of constraints to be well smaller than n^3.

Chapter 5

Experimental Procedure and Results

In this chapter we put our model to the test. We are interested in studying the Portuguese case following the propositions to change the electoral system in place. Therefore, multiple scenarios are tested focusing on the territory of continental Portugal. Unbiased electoral maps and biased electoral maps towards each of the main Portuguese parties are created and compared. Note that we refer to a map as biased or unbiased if previous electoral results are taken, or not, into account, respectively. However, unintentional gerrymandering is still a possibility [13].

The set of rules followed to generate the instances is the typical for redistricting:

1. The number of people registered to vote in each electoral district (ED) must not diverge more than 25% from the theoretical best value B. This means that ϵ in Equation 4.3 is set to 0.25.

2. The new electoral districts should be as compact as possible.

3. All electoral districts must be contiguous.

4. There must be conformity to administrative boundaries, i.e., new electoral maps must respect the current administrative divisions. The largest possible administrative divisions should be kept whenever possible without disregarding the first rule.

The population margin ϵ is difficult to set. The bigger the maximum margin to a theoretical ideal value, the less equal is the voting power between electoral districts. Some countries such as the United States of America prefer a margin as low as possible, although that is only possible because administrative boundaries are ignored in favor of census tracts. Other countries set the limit at values such as 10% (Italy, Australia or Ukraine).[1] In this work, the maximum margin value is set at 25%, as used in countries such as Canada or Germany (it was also briefly in the United Kingdom). Moreover, it was the maximum margin value presented in one of the propositions for Portugal back in 1998 [4, 17].

[1] `http://aceproject.org/main/english/bd/bdb05a.htm`, accessed on 19/12/2020.

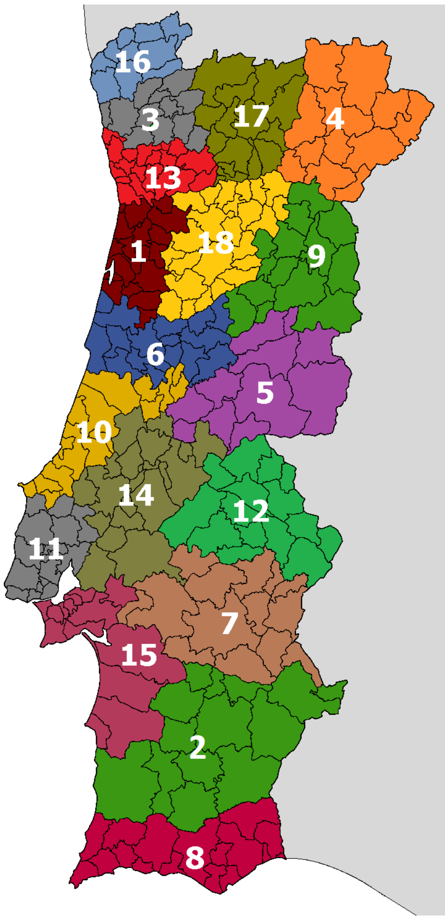


Figure 5.1: Map of Portuguese regions and respective identifier (ID).

ID	Name	Elected
01	Aveiro	16
02	Beja	3
03	Braga	19
04	Bragança	3
05	Castelo Branco	4
06	Coimbra	9
07	Évora	3
08	Faro	9
09	Guarda	3
10	Leiria	10
11	Lisboa	48
12	Portalegre	2
13	Porto	40
14	Santarém	9
15	Setúbal	18
16	Viana do Castelo	6
17	Vila Real	5
18	Viseu	8
Total		215

Table 5.1: Identifiers, names and current number of elected officials for each Portuguese region.

The formulations proposed in Sections 4.2 and 4.3 guarantee district compactness and contiguity, respectively. Although, the rule of conformity to administrative boundaries is the hardest to satisfy. Portugal will not be considered as a whole, instead we look at each first level administrative subdivision individually.[2] This is equivalent to the problem in the United States of America where each state is redistricted separately. The number of mandates, that is, the number of officials each Portuguese region elects for the National Assembly, is defined by the Comissão Nacional de Eleições[3] (an independent body whose mission is to make sure that elections are fair) and calculated using the D'Hondt method [21] with the populations of each region. The usage of this method guarantees that, independently of the region an individual lives on, its voting power is approximately the same across the country. The current number of parliament members for each region (calculated using the most recent population data) is presented in Figure 5.1 and Table 5.1.

Whenever possible, second level administrative divisions are preserved (the smaller subdivisions in each Portuguese region in Figure 5.1, called municipalities[4]). However, that is impossible in some cases, either because the number of elected officials is larger than the number of municipalities, or

[2]In Portuguese, the first level administrative subdivision (each of the map regions in Figure 5.1) is named a *"distrito"*. In order to avoid confusion with electoral districts, in this work we use the term region to designate these areas.

[3]Available at (In Portuguese): `http://www.cne.pt/`, accessed on 19/12/2020.

[4]In Portuguese, the municipalities are named *"municípios"* or *"concelhos"*. A complete list of Portuguese municipalities can be found at: `https://en.wikipedia.org/wiki/List_of_municipalities_of_Portugal`, accessed on 19/12/2020.

because the population differences between municipalities does not allow joining them while respecting the population equality rule. In the aforementioned cases, the first step is to split the region in as many different areas as possible at the municipality level. Next, each of these areas is redistricted at the civil parish level[5] (the third and lowest level of Portuguese administrative divisions) to find the final new electoral map of a region.

This approach has two major advantages. Firstly, it avoids splitting municipalities between EDs as much as possible. Secondly, it does not make the redistricting at the parish level overcomplicated since the areas to be redistricted at this level will contain less territorial units each and also elect fewer officials (lower values of n and k, respectively, the biggest factors in complexity). Most regions have over 100 parishes (values of n over 100) which creates a huge search space. Hence, we find that this is a great approach to obtain global optimal solutions in acceptable amounts of time while at the same time preserving second-level administrative divisions as much as possible.

Figure 5.2 exemplifies the approach when there is the need to redistrict at the parish level. Consider the region of Aveiro (region number 01) and its 19 municipalities (map A in Figure 5.2). Furthermore, consider also that we want to elect 8 officials to represent this region. When we try to redistrict at the municipality level, the constraints are unsatisfiable because there is a municipality with 125534 registered voters and the upper population boundary is 100883 (with a 25% population margin). Therefore, we first take these 19 municipalities as our territorial units and split them into 4 areas. Through the maximization of the compactness of each area map B is obtained. We proceed to consider the civil parishes inside these areas our new territorial units (map C). Finally, districting each of these 4 areas into 2 single-member districts (again maximizing for compactness) yields our final electoral map for the region of Aveiro (map D in Figure 5.2).

Considering that the smallest territorial units possibly used in this work are the current Portuguese parishes (census tracts or census blocks are never used) we are meeting the criterion of conformity to administrative boundaries set before.

We focus on continental Portugal, meaning that we do not redistrict the archipelagos of Açores and Madeira since it is impossible to meet the contiguity and population constraints, particularly in the islands of Açores. This is compatible with one of the 1998 propositions [17] where the Portuguese archipelagos would each have their regional circle and elect member to the national assembly using party-list proportional representation. However, one way of doing it would be to create a virtual border between the closest islands where the territorial units of one island would connect to the territorial units of the other. The constraints would then be satisfiable and the procedure previously described could be used to also redistrict those regions.

The political districting problem is one such problem where finding global optimal solutions is of the utmost importance. Hence, in the following scenarios we always use a complete model, the base model previously detailed (Figure 4.1) with the tree-based contiguity constraints (4.3.1). The time limit is set to three hours and the maximum population margin to the theoretical best value (B) is set to 25%. Whenever the time limit is exceeded before finding the global optimal solution, the instance is also run

[5]In Portuguese, the civil parishes are named *"freguesias"*. A complete list of Portuguese civil parishes can be found at (in Portuguese): `https://pt.wikipedia.org/wiki/Lista_de_freguesias_de_Portugal`, accessed on 19/12/2020.

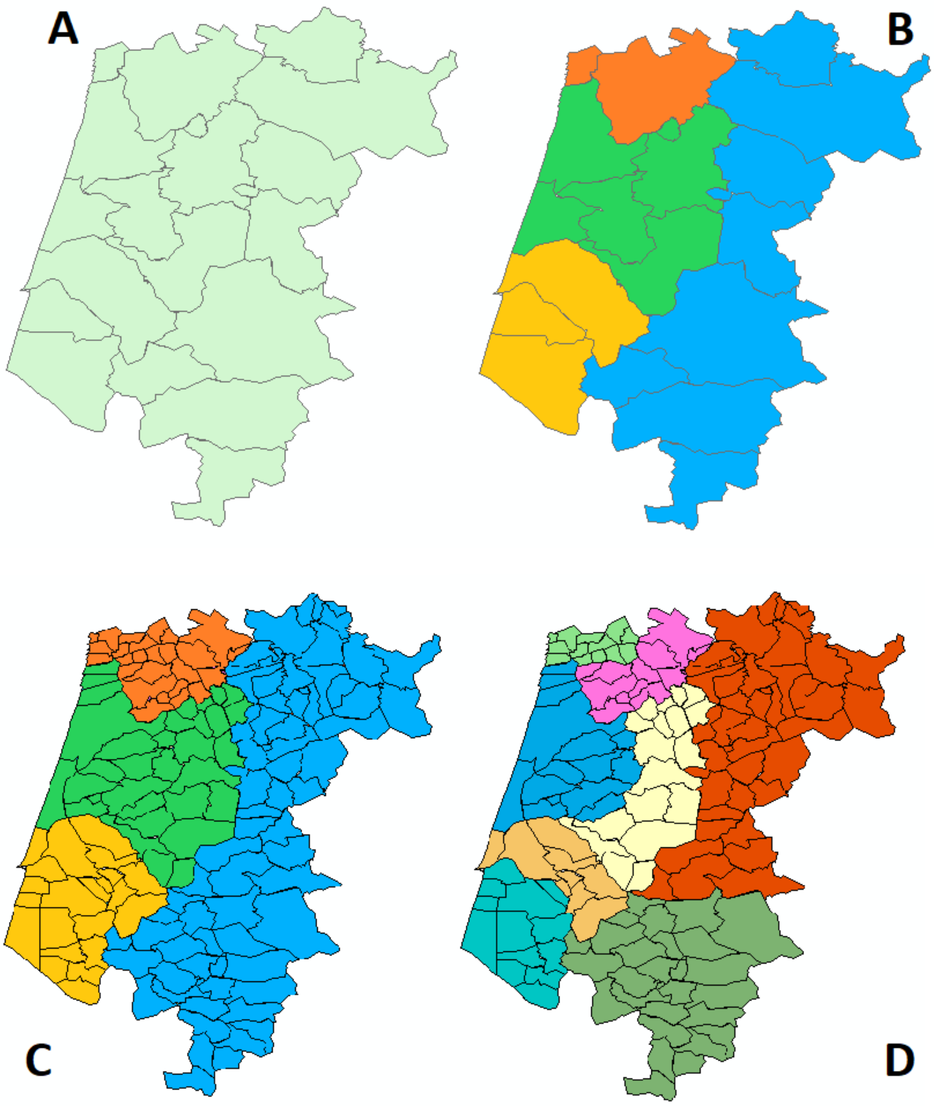


Figure 5.2: Steps to redistrict at the parish level.

using the shortest-path contiguity formulation. The presented solution is the best one found with either formulation but is always presented striped in the following figures with our results to differentiate from certain global optimal solutions.

Regarding the computational infrastructure, all results are obtained on four Intel Xeon Silver 4110 processor (total of 32 cores) running Debian Linux with 64GB of RAM. In single-objective optimization instances the solver used is the CPLEX Optimizer[6] (version 12.6.0), a commercial solver, allowing us to exploit the full potentiality of our system. Multi-objective optimization problems are solved using Sat4jMoco[7], an open-source library in Java.

[6] `https://www.ibm.com/analytics/cplex-optimizer`, accessed on 20/12/2020.
[7] `https://gitlab.ow2.org/sat4j/moco`, accessed on 20/12/2020.

5.1 Scenario 1: Unbiased Redistricting

The first question we are interested in answering is: what does a fair redistribution of the Portuguese electoral map using a FPTP voting system look like? Therefore, we must split the territory of Portugal into 215 single-member districts. Each region is partitioned separately according to the values in Table 5.1. The redistricting of each Portuguese region is done without using any gerrymandering techniques, meaning that the only objective function is to maximize the compactness of our new single-member districts (described in Section 4.2). This will prove the capabilities of the model and create completely unbiased electoral maps.

In our results, each color represents an electoral district and notice that they are always contiguous, i.e., a change in color means a change in ED. The regions of Porto and Lisboa are impossible to redistrict in this particular scenario because both contain parishes with populations over the upper population boundary and for that reason are crossed out from the following map with our results (Figure 5.3). Whenever the presented EDs in our maps are striped it means that the solution shown may not be the global optimal solution because. There are two possibilities, it was either obtained using the shortest-path contiguity formulation or the time limit was exceeded using the tree-based formulation.

The achieved results are good, we can find global optimal solutions for most of the territory of continental Portugal following all the previously established rules. Concerning the compactness, there are not major issues across regions and odd shapes are scarce. Probably the most unusual shape is the pink ED in the region of Vila Real (labelled as A on the left map in Figure 5.4) and that was due to that region being one of the 6 redistrictings at the municipality level which means that there are less possible solutions to the problem. Combined with the fact that the blue district (labelled as B) over district A has a population close to the upper population limit making it impossible to join with other territorial units, forces district A to end with an odd shape. In order to prove the capabilities of our objective function to generate compact solutions, we decided to redistrict the region of Vila Real at the parish level (the TUs are now the parishes instead of the municipalities and the number of clusters remains equal to 5). The results are presented on the right map in Figure 5.4 and prove that the potential of the objective function proposed in Section 4.2 because the solution is clearly more compact. A more compact solution comes at the cost of municipality boundaries no longer being maintained and the solution found probably not being the global optimal solution (note that all EDs are striped) because the number of parishes considered is 197 ($n = 197$) to be divided into 5 EDs ($k = 5$), creating a large search space.

5.2 Scenario 2: Using another voting system

The first-past-the-post system (FPTP) is not the only voting system using single-member districts. Some countries in Europe such as Italy or Hungary (Figure 2.1), but also around the world (Japan or Mexico, for example) use the parallel voting system. This was precisely the system proposed by the two major Portuguese parties in 1998 [17] making it a strong candidate for a future electoral system revision in Portugal and definitely worth exploring in this work.

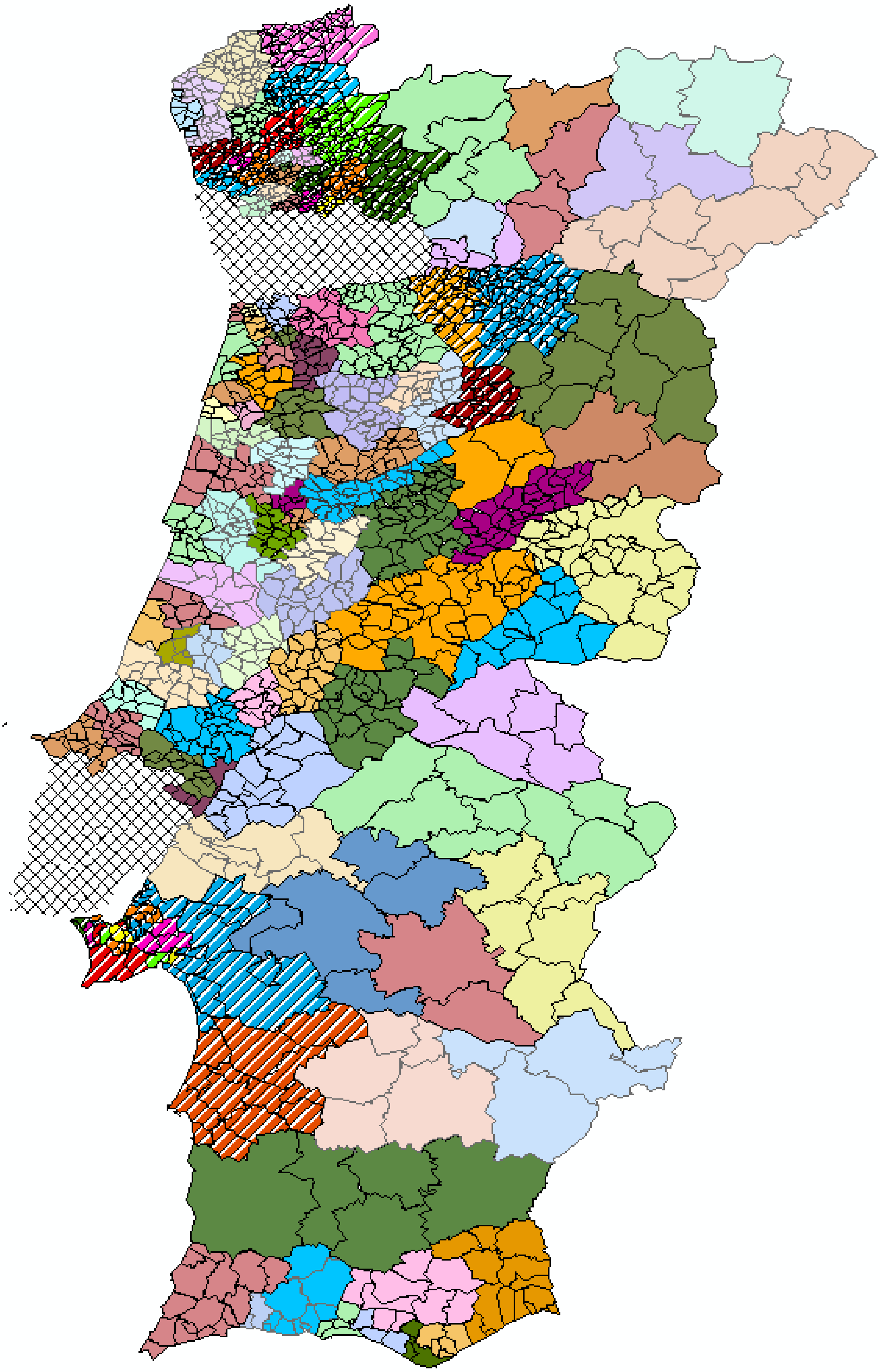


Figure 5.3: Complete electoral map of continental Portugal using the first-past-the-post system.

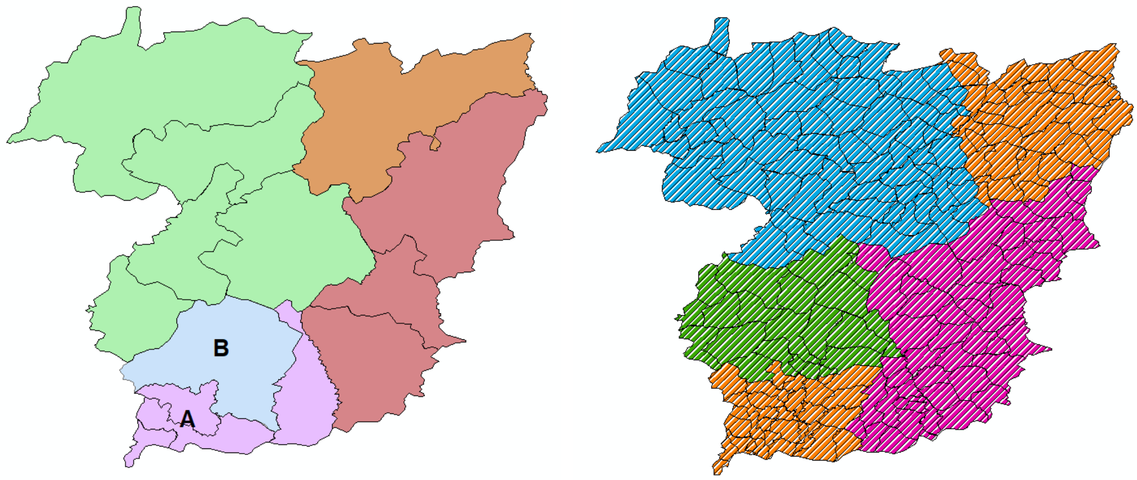


Figure 5.4: Results for the region of Vila Real (17) at the municipality level and at the parish level.

In this electoral system, a percentage of the votes is distributed using single-member districts and the rest using a party-list proportional representation method (most commonly, the D'Hondt method) at the national level. In the election day, a voter casts two different votes, one for an individual in its single-member district and the other one for a party at the national level. The idea behind it is bringing the voters closer to politics (through single-member districts) by voting directly to elect a parliament member in the national assembly who is typically more concerned and connected with their local electoral area, while maintaining proportionality through the national circle at the party level (avoiding the tendency to create a two-party system of a simple FPTP voting system).

Under a parallel voting system, we consider that the number of single-member districts to be created in each Portuguese region is half the current number of elected officials (see Table 5.1), rounded up. Hence bringing the number of officials per region to a level where the constraints are satisfiable, even in the most populous regions while maintaining, approximately, the same voting power between regions. The total number of single-member districts to be created becomes 112 in continental Portugal (Table 5.2). These 112 seats would be awarded through a FPTP system while the remaining seats would be awarded through party-list proportional representation at the national level. Note that the percentage of seats awarded through single-member districts and party-list proportional representation varies greatly between countries that use a parallel voting system.

Once again, the focus is only on continental Portugal and considering only one objective function: maximize the compactness. Hence, totally fair electoral maps for Portugal are created. Since the instances to solve are single-objective optimization instances, the solver used is the CPLEX Optimizer.

Since we believe this is a strong possibility in a future Portuguese electoral revision, this is also the electoral system that we will use in future scenarios. Hence, the results in this scenario will work as a baseline to compare with the following ones where instead of creating unbiased distributions of Portugal we deliberately try to make the redistricting advantageous to a certain party.

For comparison with following scenarios where we are applying gerrymandering techniques, we will track how many electoral districts each of the two main Portuguese parties (PS and PSD) wins. The

ID	Name	Electoral Districts
01	Aveiro	8
02	Beja	2
03	Braga	10
04	Bragança	2
05	Castelo Branco	2
06	Coimbra	5
07	Évora	2
08	Faro	5
09	Guarda	2
10	Leiria	5
11	Lisboa	24
12	Portalegre	1
13	Porto	20
14	Santarém	5
15	Setúbal	9
16	Viana do Castelo	3
17	Vila Real	3
18	Viseu	4
Total		112

Table 5.2: Identifiers, names and number of electoral districts for each region under a parallel voting system.

party that gets the most votes inside an electoral district is the winner (independently of the margin) just like in a FPTP system. These results are presented in the unbiased column of Table A.1 of the Appendix.

Dividing the Portuguese territory in 112 electoral districts according to Table 5.2 (using the algorithm described in Figure 5.2) results in the 35 instances characterized in Table 5.3.

Figure 5.5 shows the complete map of continental Portugal after redistricting each of its regions individually following the algorithhm described in Figure 5.2. Figure 5.6 and Figure 5.7 show close-ups of the more densely populated regions of Portugal (the metropolitan areas of Lisbon and Porto) where the redistricting had to be done in two steps (the first at the municipality level and the second one at the parish level). In this scenario, given that the number of officials to elect in each region (Table 5.2) is lower than in the previous one, we were able to redistrict 11 out of the 18 regions while respecting the municipality boundaries. There are only 5 instances (out of 35) comprising 17 EDs (out of 112) that could not be solved within the time limit established. These 5 instances are also solved using the shortest-path contiguity formulation and the best (most compact) solutions are the ones presented in our results. As an example, in the region of Porto, one instance with $k = 4$ and $n = 149$ is striped in Figure 5.5 because the presented solution is the one found with the shortest-path model as the tree-based model could not even find a solution after 3 hours.

In the previous scenario (Section 5.1), we already showed the capabilities of our model to generate compact solutions and once again it delivers excellent results without any major compactness issues, even in instances where the time limit was exceeded which may not be global optimal ones (striped EDs).

The number of seats each of the main Portuguese parties (PS and PSD) would get under this scenario, in each region, is presented in Table A.1 of the Appendix (the unbiased columns). You may notice that PS wins in 84 out of 112 electoral districts (75%) while PSD wins the remaining districts which

Instance Number	Portuguese Region (ID)	Territorial Units (n)	Electoral Districts (k)
1	01	22	2
2	01	25	2
3	01	33	2
4	01	67	2
5	02	14	2
6	03	34	2
7	03	37	2
8	03	53	2
9	03	70	2
10	03	153	2
11	04	12	2
12	05	11	2
13	06	64	2
14	06	91	3
15	07	14	2
16	08	16	5
17	09	14	2
18	10	57	2
19	10	53	3
20	11	15	6
21	11	15	6
22	11	24	6
23	11	80	6
24	12	15	1
25	13	11	4
26	13	15	4
27	13	25	4
28	13	43	4
29	13	149	4
30	14	21	5
31	15	12	4
32	15	43	5
33	16	10	3
34	17	14	3
35	18	24	4

Table 5.3: Correspondence between instance number, Portuguese region, the number of territorial units and districts to be created.

might seem too big of a difference. The justification behind these results lays on two main factors. First, the data used to calculate the electoral results is from the 2019 legislative elections where PS received 36.65% of the national vote while PSD only received 27.90%.[8] Second, the fact that the FPTP voting system used in single-member districts tends to over-represent the most voted parties [25, 54], particularly if the votes are geographically relatively well distributed. However, it is important to mention that these results would be balanced with the other half of the seats in parliament being awarded under a party-list proportional representation method at the national level evening the electoral results and attributing seats to least voted parties.

[8] Available at (in Portuguese): `https://www.eleicoes.mai.gov.pt/legislativas2019/territorio-nacional.html`, accessed on 22/12/2020.

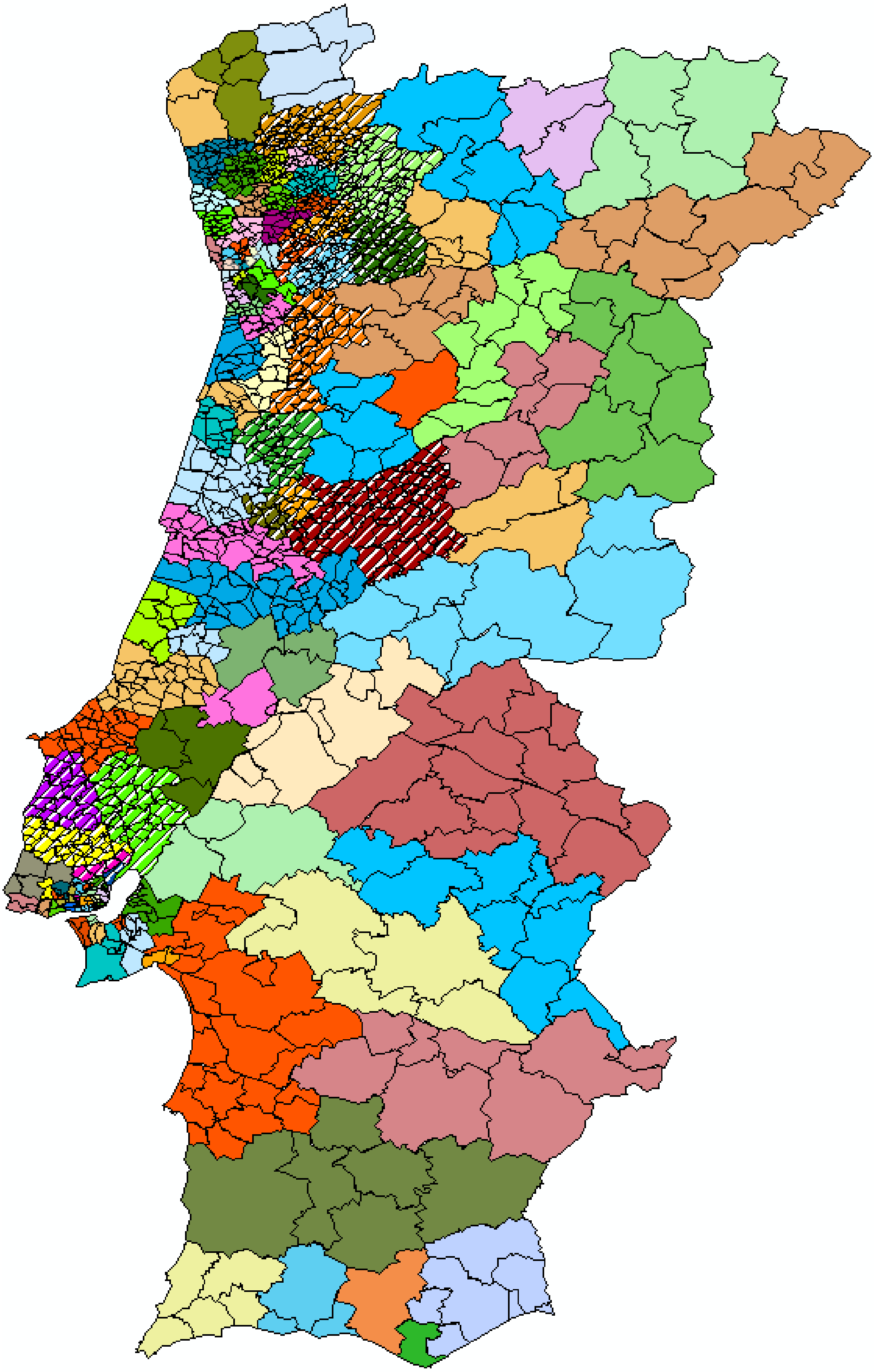


Figure 5.5: Complete electoral map of continental Portugal using a parallel voting system.

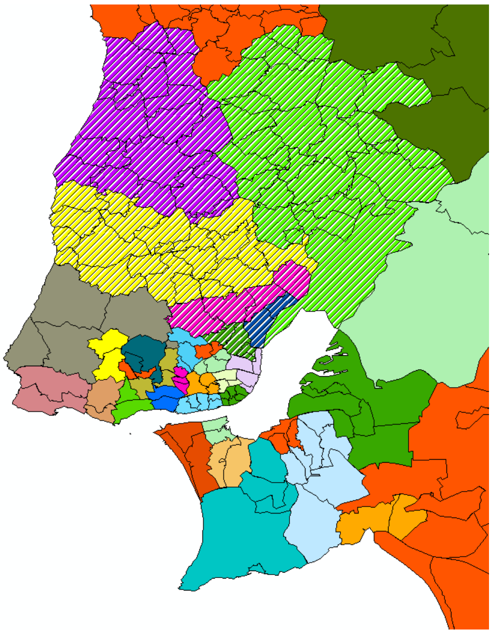


Figure 5.6: Close-up of Lisbon and Setúbal areas.

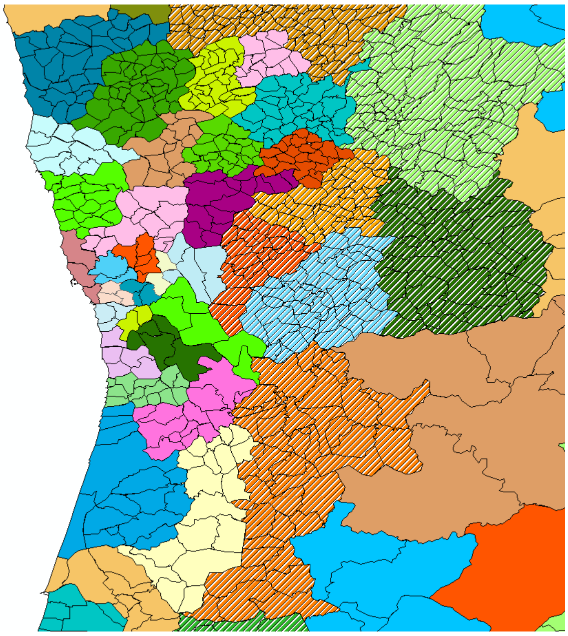


Figure 5.7: Close-up of Braga, Porto and Aveiro areas.

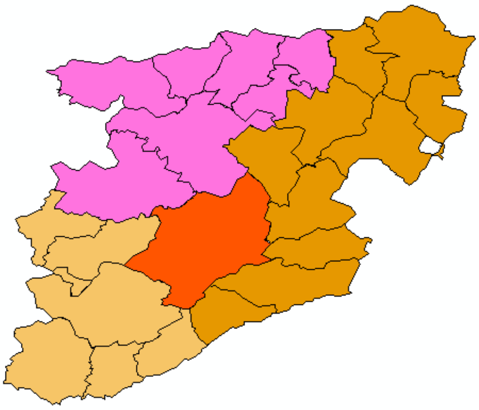
5.3 Scenario 3: Gerrymandering to Maximize Party Results

In this section, we deliberately generate maps that maximize the district wins of a party while still maximizing the compactness, meaning that we are now upon a multi-objective combinatorial optimization problem instead of a single-objective one. This can be done by adding Equation 4.39 in Section 4.4 to our compactness objective function. The generated maps are favorable towards the main parties (PS and PSD) and a comparison of the electoral results with the ones from a fair redistricting (previous scenario) is done.

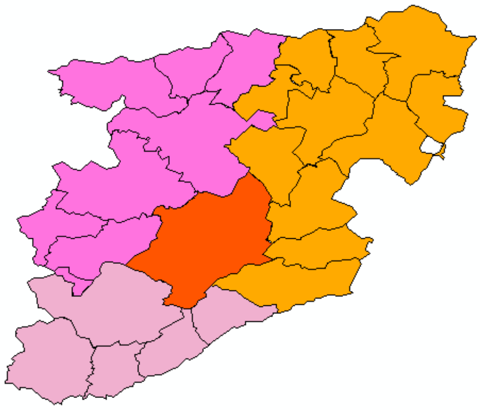
The only variable changed is adding one more objective function (minimizing party losses, i.e., maximizing party wins) so that a comparison of the results with the ones obtained in the previous scenario is possible. Hence, we are also considering a maximum margin of 25% to B, the first level administrative divisions will be considered separately and municipalities (second level divisions) will be kept whenever possible. The solver used is Sat4jMoco, an open-source solver that solves multi-objective optimization problems.

The districting was done, for all Portuguese regions, and despite a complete map not being presented, the complete electoral results (number of seats won by each party) for each objective is presented in Table A.1 of the Appendix. In instances where the Pareto front contains multiple solutions, the results presented in Table A.1 are of the solution that is the most advantageous towards the party currently being maximized.

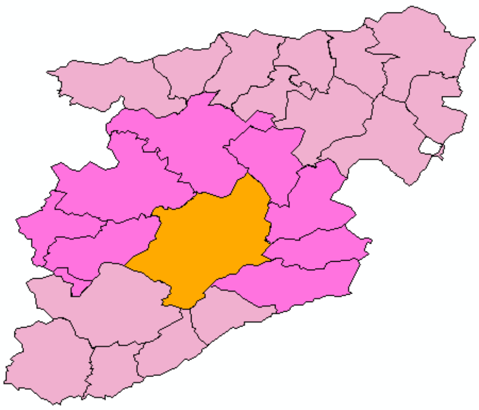
Considering that the associated colours of parties PS and PSD are, respectively, pink and orange, in Figures presenting our results (also valid in the following Sections) EDs colored in shades of pink are wins for party PS and colored in shades of orange are wins for party PSD. Just like in the previous scenario, striped districts in our results mean that the time limit of three hours was exceeded before we



(a)



(b)

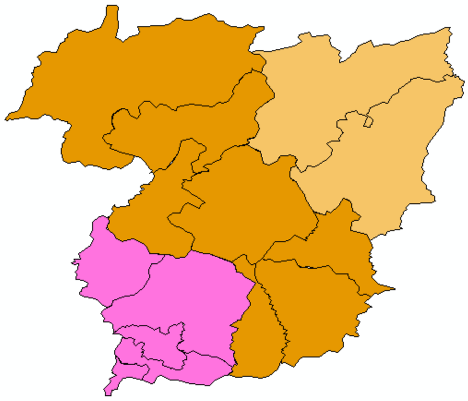


(c)

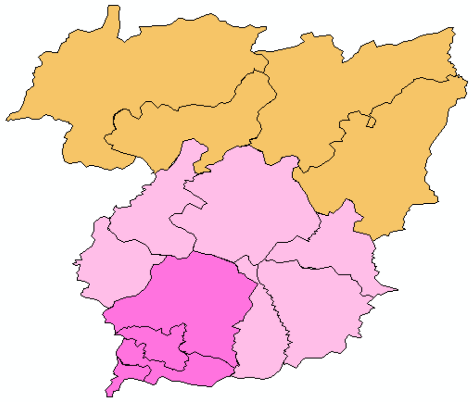
Figure 5.8: Pareto front for the region of Viseu (18) maximizing compactness along with PS wins. In shades of pink, EDs won by PS, in orange EDs won by PSD.

could find a global optimal solution with the complete model. The presented EDs are the best solution found with either the tree-based or the shortest-path contiguity models which might not be the global optimal solutions (notice that the colours still represent the winning party in that ED).

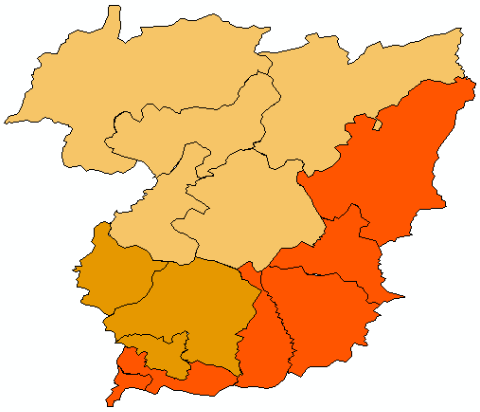
Figure 5.8 presents the three possible redistricting options for the region of Viseu (region number 18) at the municipality level. These three points of the Pareto front represent the trade-off between electoral district wins for party PS and compactness (border length). The top left map (Figure 5.8a) is the unbiased distribution and also the most favorable towards PSD. On this map, the total border length between TUs in the same cluster is 614891 meters and it goes down to 611282 meters as the number of electoral district wins for party PS increases (Figure 5.8b). In the most advantageous map towards PS, (Figure 5.8c), the total border length is further reduced to 594286 meters. These values represent a 0.59% and a 3.35% decrease in compactness to the unbiased map. It is interesting to note that between the map in Figure 5.8a and the map in Figure 5.8c the electoral results were essentially inverted – PSD wins 3 EDs in Figure 5.8a while PS wins 3 EDs in Figure 5.8c – exposing the potential of gerrymandering to change the course of an election. However, in Figure 5.8c the pink districts start presenting some odd shapes with the centre one twirling around the orange district which is the first



(a)



(b)



(c)

Figure 5.9: Results for the region of Vila Real (17). In pink EDs won by party PS and in orange EDs won by party PSD.

indicator of gerrymandering in action.

For the region of Vila Real the results are presented in Figure 5.9. The unbiased distribution (5.9a) has the orange party winning 2 districts and a total compactness measure of 440210 meters. If we are also trying to maximize the wins of the pink party (Figure 5.9b), it is possible to make PS win 2 out of the 3 EDs but the length of the border between TUs in the same cluster goes down to 426060 meters (a 3.21% decrease). Finally, the map in 5.9c, shows a redistricting option where PSD would win all the seats but we can immediately notice that one of the districts has an extremely odd shape and that is confirmed by our compactness measure now being only 371788 meters (a 15.54% decrease to the unbiased version).

Figure 5.10 shows the results for the region of Aveiro (region number 01) which is one of the cases where a redistricting only at the municipality level is not possible. Therefore, following the algorithm presented in Figure 5.2 results in 4 instances (each containing 2 electoral districts). In the unbiased distribution (Figure 5.10a), the pink party wins 5 EDs while the orange party wins the remaining 3. In Figure 5.10b, we are maximizing the ED wins of PS while also maximizing compactness and it is

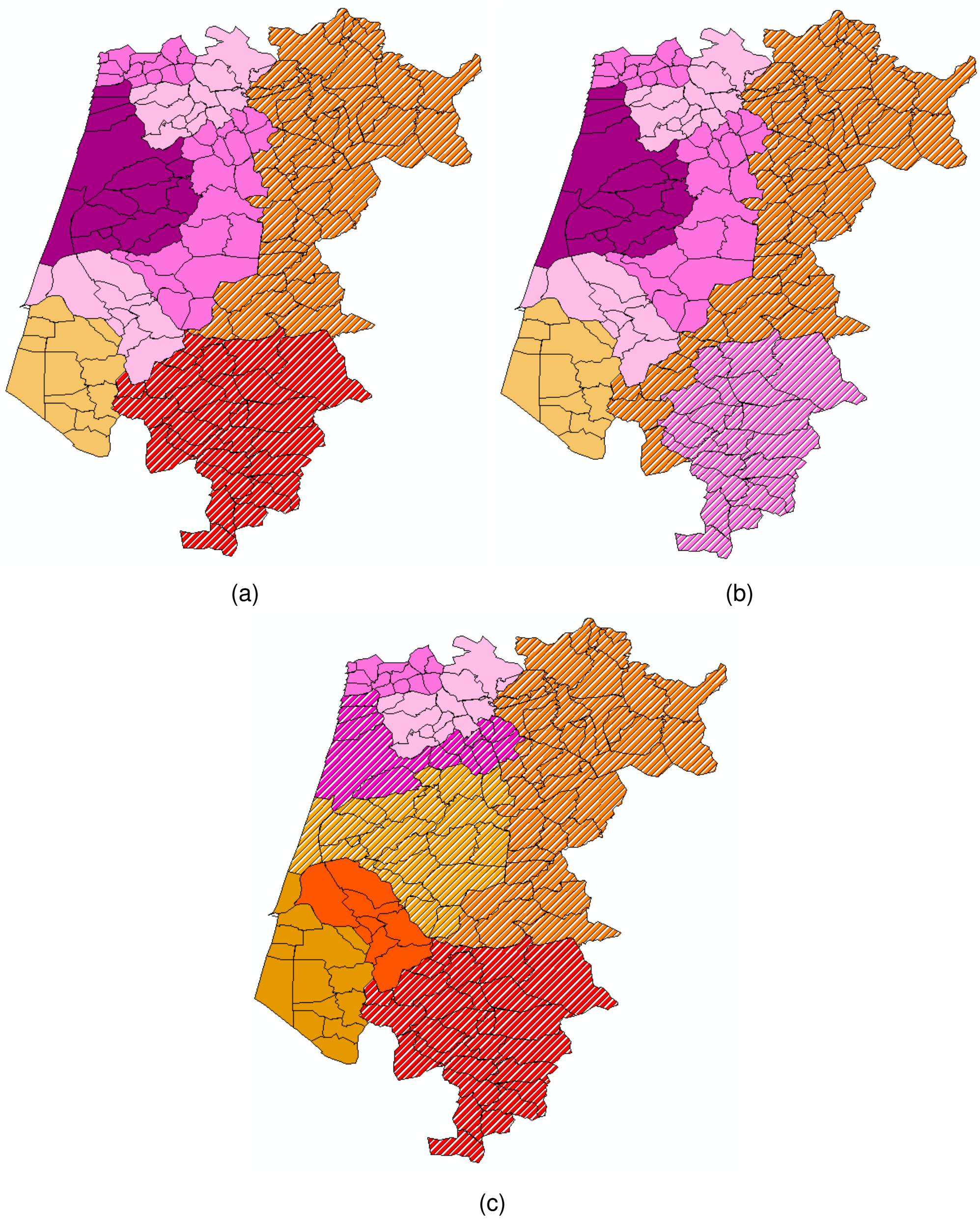
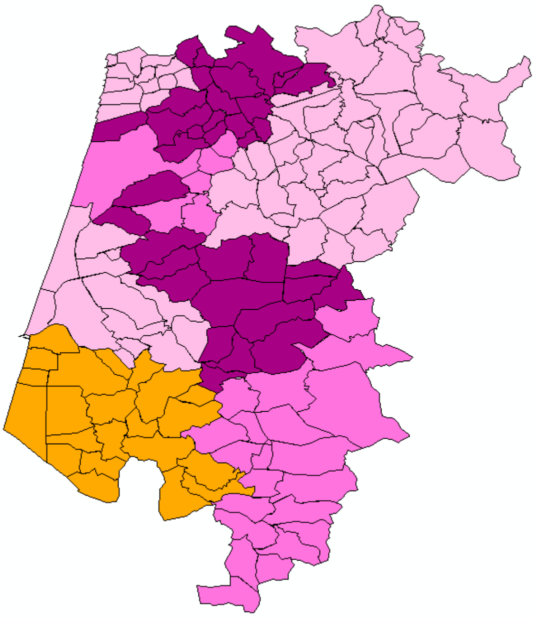
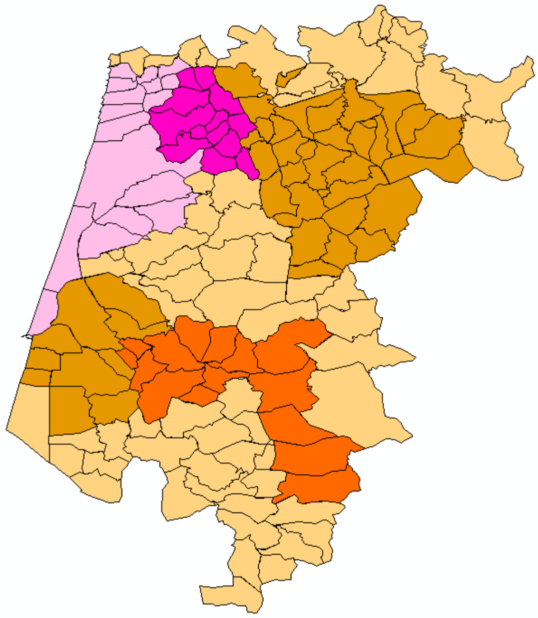


Figure 5.10: Different results for the region of Aveiro (01) at the parish level with a first division at the municipality level.

possible to make PS win one more district through packing the opposition voters in the odd shaped orange district (the striped one). Finally, in Figure 5.10c, we are maximizing the wins of PSD along with compactness and the results show that it is possible to make PSD win 5 EDs while PS only wins 3 out of the 8 single-member districts. Comparing it with the unbiased map in 5.10a, we conclude that gerrymandering is, once more, inverting the results of the elections. Although it is impossible to guarantee that the presented solutions are the global optimal ones, these results (only a lower bound) are enough to prove that it is definitely possible to gerrymander a territory with consequential effects. Focusing on the most advantageous maps towards each party (Figures 5.10b and 5.10c) we note that it possible to have party PS clearly win the elections in a certain region with 6 out of 8 seats (75%) or clearly lose winning only 3 seats out of 8 (37.5%), a 3 seat change.



(a)



(b)

Figure 5.11: Extreme possible results for the region of Aveiro at the parish level ignoring the boundaries of municipalities (no prior division) when maximizing party PS (left) or party PSD (right).

Observe that in Figure 5.10 a first division at the municipality level is performed in order to preserve second-level administrative subdivisions as much as possible before moving to the parish level. However, it is interesting to study how far could the results be skewed if we were to ignore these administrative subdivisions and consider all parishes at the same time. This test creates more liberty for the electoral districts to be gerrymandered because the number of territorial units considered at the same time is larger. Consequentially, the number of possible solutions is also much larger and, although global optimal solutions could not be found (even after increasing the time limit to 10 hours) in Figure 5.11 we present possible solutions that are a lower bound on the number of possible wins for each party. Each one of the solutions is extremely favorable to one of the parties (hence unfavorable to the other) and shows extremely well how gerrymandering can be used to give a huge advantage to a certain party when both receive a similar number of votes (in the region of Aveiro PS received 34.31% of the votes and PSD 33.54%, in the 2019 elections).

On the electoral map on the left of Figure 5.11, maximizing the wins of party PS (along with compactness) it ends up winning 7 of the 8 electoral districts (87.5%). It is clear that some districts seem to extend themselves into the neighboring ones, creating some odd shapes (particularly in the center) which cracks the opposition voters. On the map presented on the right we are trying to maximize the ED wins of PSD (along with compactness) and it is possible to make it win 6 out of 8 districts (75%). Remember that under an unbiased districting PSD would only win 3 (Figure 5.10a). In Figure 5.11b it is extremely clear that some of the districts are immensely elongated, again cracking and packing the opposition. Notice that some EDs are stretched out to a point where they span the whole region of Aveiro from side to side and the TUs within seem to only be connected to other parishes by few meters.

Obviously, applying gerrymandering techniques translates into a decrease in compactness. The compactness measure (border length shared between TUs on the same ED) goes down from 1448842 meters in the unbiased distribution (Figure 5.10a) to 1358548 meters and 1273997 meters, respectively, on the maps of Figure 5.11. This represents a 6.23% decrease in compactness in the case where we are maximizing the pink party and a 12.07% decrease when we are maximizing the orange party.

You may notice that one of the orange EDs in Figure 5.11b seems to be violating the contiguity constraint but that only happens because it is actually an exclave[9] of another territorial unit that is connected to the remaining ones in that electoral district. This is an extremely rare situation because it is highly prejudicial to the maximization of the border length between TUs in the same ED (the objective function maximized) to include exclaves since they do not share any border with the remaining TUs in that ED. However, one possible (and simple) solution to prevent this situations would be to force both TUs – the one with the exclave (let us call it a) and the one in-between (let us call it b) – to be in the same cluster, by adding the following constraint:

$$\forall i \in \mathcal{K}, a, b \in N : \; x_{i,a} \iff x_{i,b} \tag{5.1}$$

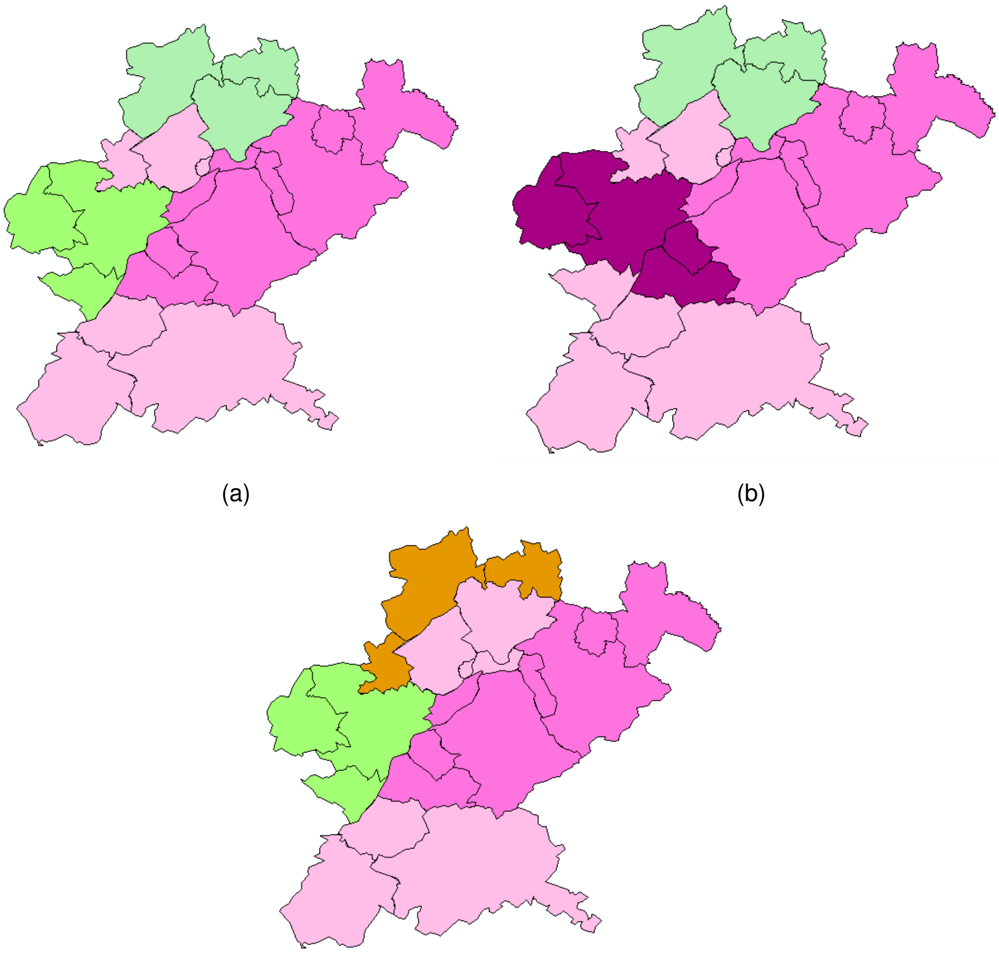
In the 2019 elections (data used for the biased distributions) PS received 36.65% of the national vote while PSD received 27.90% meaning that, at the national level, PS is expected to be the party with the most parliament members under a FPTP system. This prediction is confirmed by the complete electoral results at the national level which are presented in Table A.1 of the Appendix. They show that under an unbiased distribution PS would win 84 out of 112 seats (75%) and PSD the remaining 28 seats (25%). Using gerrymandering, in the most favorable distribution towards PS, it is possible to make it win 13 more seats, bringing its total to 97 (86.61%). Contrarily, under an unfavorable distribution it can also lose 8 seats to the opposing party leaving it with 76 members in parliament out of the 112 distributed using the FPTP voting system (67.86%).

5.4 Scenario 4: Securing Electoral District Wins

This scenario is similar to the previous one, where we are interested in maximizing the results of a certain party. However, just like in real-world applications of gerrymandering, we want to be sure that the party we are trying to favor definitely wins in some electoral districts. Therefore, instead of considering a party the winner of an ED if it receives at least one more vote than all the other parties, we will be requiring at least 15% more votes than all the other parties. If no party achieves a victory by this margin we can label that district as a toss-up district, i.e., a competitive district where either party has a chance of winning. Toss-up districts are colored in shades of green in the following figures with our results while the colours used for each of the main parties are maintained.

From the point of view of a party it makes sense not only to maximize its results but also to minimize the district wins of its direct rival, with the objective of bringing some EDs to a toss-up level instead of

[9] `https://www.merriam-webster.com/dictionary/exclave`, accessed on 22/10/2020.



(a) (b) (c)

Figure 5.12: Different results for the region of Santarém (14) at the municipality level. Green are toss-up districts, in pink, EDs won by PS and, in orange, EDs won by PSD.

being safe wins to an opposing party. This is exactly what we simulate in this scenario, we try maximize the ED wins of a party and minimize the ED wins of the opposing party (a toss-up district is neither a win nor a loss to any party). Obviously, compactness is still maximized along these two cost functions, hence we are now optimizing three objective functions at the same time.

Figure 5.12 shows how easily it is to change the outcome of the elections by making small changes to the electoral maps. Hence, it is possible that an objective function that simply tries to minimize the number of changes to current electoral maps to conform to new census data (such as the one proposed by Browdy [11]) might not be the best option. The unbiased distribution (Figure 5.12a) and the most favorable distribution to party PSD (Figure 5.12c) only have two TUs swapped, although that is enough to make PSD safely win one ED in a district that would otherwise be a toss-up. The length of the borders is reduced from 513793 meters to 458230 meters, a 10.81% decrease. Similarly, if we imagine that PS controls the redistricting process and tries to maximize its electoral results, with a couple of changes to the electoral maps they can secure one more district and win 4 out of 5 confidently.

The solutions that make up the Pareto front (all non dominated solutions) for the region of Faro when maximizing PS and minimizing PSD are presented in Figure 5.13. Once again, it demonstrates how

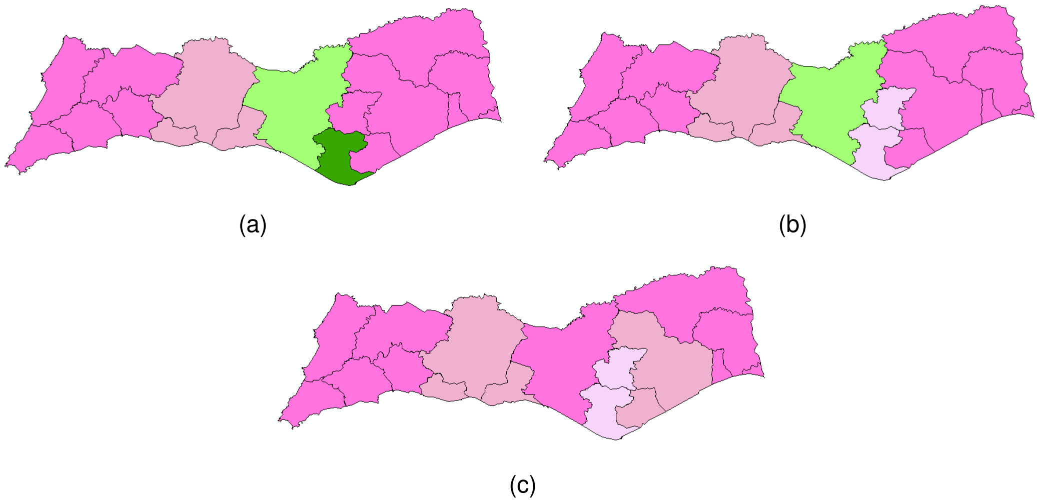


Figure 5.13: Pareto front for the region of Faro (08) at the municipality level when maximizing PS and minimizing PSD along with compactness. Green are toss-up districts, in pink, districts won by PS.

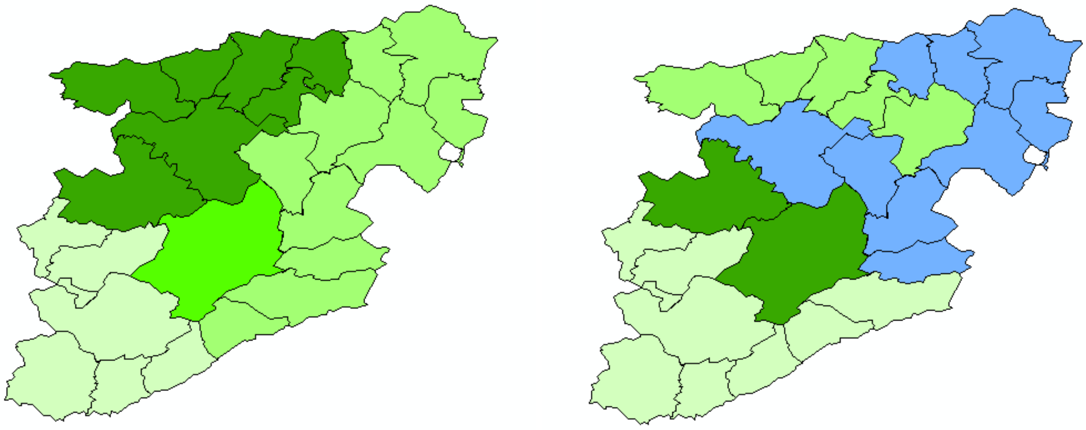
easily one party can favor itself if it gets the power to control the redistricting process. Gerrymandering would allow PS to win all EDs by cracking the opposition voters between EDs where PS already has a strong preference. The result is the creation of an extremely odd ED that is only marginally contiguous (Figure 5.13c, the most biased towards PS) and a total border length of 316975 meters. Note that this value represents a resounding 23.56% decrease in compactness to the unbiased map.

Once again, the redistricting was performed in all Portuguese regions and the complete results for all regions of continental Portugal are presented in Table A.2 of the Appendix. Under an unbiased distribution, PS would win 41 EDs and PSD 3 EDs leaving the remaining 68 as toss-ups. It is interesting to notice how the biased redistrictings can totally skew the electoral results at a national level, towards any of the parties. If PS controlled the redistricting process, then it could possibly get 53 certain electoral district wins (12 more) while its rival (PSD) gets 0 safe seats because the remaining 59 EDs are all toss-ups. On the opposite spectrum, a redistricting controlled by PSD, maximizing its performance and minimizing PS, still gives PSD good chances in the elections since it is possible to increase the number of safe seats from 4 to 7 while also increasing the number of toss-up districts by 2 (to a total of 70).

5.5 Scenario 5: Two-Party System with Gerrymandering

From the Duverger's Law [16, 54] we know that a first-past-the-post system favors the creation of a two-party system. In this section we propose a scenario where we are precisely interested in studying the formation of a two-party system in Portugal.

It is easy to divide the Portuguese political spectrum into left and right because the last two governments were a right wing coalition and a left wing coalition, respectively. Moreover, in parliament, these two blocks often vote in a similar fashion in many policies. We start by grouping the votes of the left wing parties (under a party we name as Left) and the votes of the right wing parties (under party named Right) and then create electoral maps that maximize one of the sides and minimize the other. In this



(a) (b)

Figure 5.14: Pareto front when for the region of Viseu (18) at the municipality level. In shades of green are toss-up districts and in blue the district won by the Right party.

scenario, unlike the previous ones, we are not only taking into account the results of the last elections (2019), but also from the 2015 elections. Unfortunately, it was impossible to also include the results of the 2015 election in Sections 5.3 and 5.4 because in the 2015 elections PSD ran on a coalition with another right wing party and it is not clear how many votes they would get without the coalition. For each territorial unit, we considered the arithmetic mean of the votes for each party in the last elections.[10] It is harder to consider electoral results prior to the 2015 elections because there was as an administrative reorganization of all civil parishes in 2013[11], which created new civil parishes by joining and eliminating others.

We are simulating a scenario similar to the United States of America, where we only have two parties and each one is trying to maximize its results while minimizing the results of the opposition. The only difference is that we allow a 25% population deviation to the best value (B) while in the United States of America it is expected to be as close to B as possible which comes at the cost of losing conformity to administrative boundaries. Our proposed model could be adapted to such scenario by decreasing the maximum population margin to B and, eventually, adding another objective function minimizing the population differences between electoral districts.

Once again, a district will be considered a toss-up district if the difference between the number of votes received by each party is less than 15% of the total of votes cast in that cluster. Toss-up districts are presented in shades of green in the following maps with our results. Electoral districts that are certain wins for the Left party (left wing coalition) are presented in shades of red while shades of blue denote districts that are Right party (right wing coalition) wins. Striped EDs still represent clusters where the presented solution might not be the global optimal one since the time limit set for our optimization (3 hours) was exceeded.

[10] Other methods such as a weighted arithmetic mean could be used. For instance, if we consider that the results will be similar to the ones of the last election we can increase its weight.

[11] Available at (in Portuguese): `https://dre.pt/web/guest/legislacao-consolidada/-/lc/107679275/202008121053/diploma`, accessed on 22/12/2020.

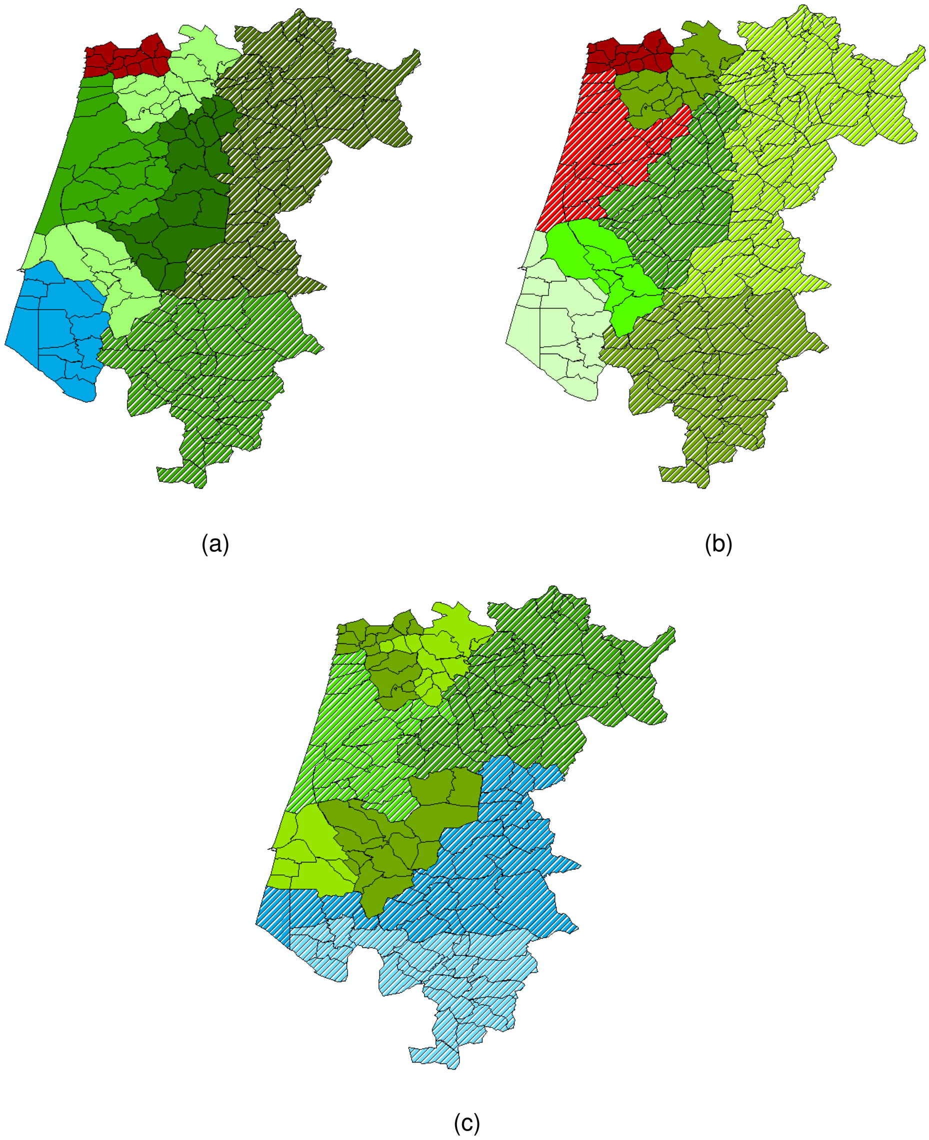


Figure 5.15: Possible results for the region of Aveiro (01) at the parish level. Green are toss-up districts, in red districts won by the Left party and in blue districts won by the Right party. Striped districts may not be global optimal solutions.

Figure 5.14 presents results for the region of Viseu when, along with compactness, the results of the Right party are maximized and the Left party minimized. Under an unbiased distribution (also the most beneficial towards the Left party) all EDs in the region of Viseu would be toss-up districts since the margin between the votes of the left wing parties and the right wing parties is smaller than 15% in all districts (Figure 5.14a). However, when we gerrymander the electoral maps (Figure 5.14b) it is possible to create an ED that is a certain win for the Right party. Visually it is immediately noticeable that the shape of the blue district is extremely unusual, and seems to encircle an already elongated district. The compactness measure is, obviously, affected and the total length of the border between TUs in the same ED is reduced from 614891 meters to 468806 meters, a staggering 23.76% reduction between the unbiased and the biased electoral maps.

Figure 5.15 is yet another example of gerrymandering in action. The electoral map in Figure 5.15a (unbiased) sees the Left party and the Right party win in 1 ED each while the remaining 6 are toss-up districts. However, when trying to minimize the results of the opposition and increase its own results at the same time, it is possible for any party to certainly win in 2 EDs and leave the opposition with no certain victories. Some side effects of gerrymandering are clear, particularly on the map favoring the Right party (Figure 5.15c), where many electoral districts are elongated to a point where some territorial units are only loosely connected to the remaining ones, a clear example of cracking.

Just like in the previous scenarios, electoral maps were created for all regions in continental Portugal. The complete electoral results are presented in Table A.3 of the Appendix and they show that, under an unbiased distribution (only compactness is maximized without looking at previous electoral results) a left wing coalition would win a majority of the seats – 65 out of 112 (approximately 58%) – awarded through a FPTP system. Note that the left wing parties were the most voted in both the 2015[12] and the 2019 legislative elections, with approximately 52% and 56% of the votes, respectively. Given that the data used was precisely from these two elections, it is not odd that under a FPTP system the Left would win more seats. Gerrymandered maps to favor the Left party could boost its results even more and make it win 78 EDs (69.64%) while leaving the opposition with no certain victories because the remaining 34 EDs are all toss-ups. On the other hand, drawing the maps in such way that it favors the Right party manages to increase its certain wins to 9 EDs and decrease certain Left party wins by 11 (when compared to the unbiased distribution).

5.6 Comparing Approaches

In this final section we are interested in studying the complexity of the problem and the quality of the proposed models in Chapter 4. In the following tests, the same benchmark instances (described in Table 5.3) created by three different models – the Hess model, our complete and incomplete models – are solved with two different solvers, the open-source Sat4jMoco and the commercial CPLEX Optimizer.

For simplicity reasons, in this section, we refer to the original Hess model [34] (presented in Figure 3.1) with the added contiguity constraints by Validi et al. [58] (presented in Figure 3.3) simply as *the Hess model*. Similarly, our proposed base formulation (Figure 4.1) with the added tree-based contiguity constraints (complete version) is referred to as *the tree-based model* while the base formulation with the shortest-path contiguity constraints (incomplete version) is referred to as *the shortest-path model*.

The experimental tests run in this section hopefully allow answering the following research questions:

1. Can we improve previous models with the additional techniques presented in Section 4.5?
2. How does an open-source solver (Sat4jMoco) measure against a commercial solver (the CPLEX Optimizer) in political districting problems?
3. Is our proposed tree-based model better than the Hess model?
4. The shortest-path based formulation can just provide an approximation. However, how far are these results from the optimal? What about the performance?

[12] Available at (in Portuguese): `https://www.eleicoes.mai.gov.pt/legislativas2015/`, accessed on 22/12/2020.

The instances used are the same as in the previous scenarios where the parallel voting system is used. Therefore, experimental results are obtained from real-world maps. Dividing the Portuguese regions in 112 electoral districts (see Table 5.2) using the algorithm described in Figure 5.2 results in the 35 benchmark instances characterized in Table 5.3. Using different models and solvers on the same instances allows us to answer our research questions.

In order to assess whether the proposed optimizations in Section 4.5.2 and Section 4.5.4 can improve the performance of the Hess model, we run the same single-objective optimization (SOO) instances with and without the optimizations. To allow the use of the optimizations in Section 4.5.2 and for comparison with our model, the original objective function in the Hess model (see Figure 3.1) was changed to the one proposed in Section 4.2, the maximization of the border length between territorial units in the same cluster. These instances were run using version 12.6.0 of the the CPLEX Optimizer solver. The time limit was set to three hours (10800 seconds) and whenever that limit was exceeded the respective cell in Table 5.4 is highlighted in orange. The maximum population margin to the theoretical best value was set to 25% (just like in previous scenarios). Note that depending on the population margin to the best value being considered the number of impossible pairs in the same district – and consequentially the efficacy of the optimization cutting those pairs – varies. Nevertheless, cutting symmetries inside the clusters proved to be more effective than removing impossible pairs.

The results are presented in Table 5.4 and comparing the performances with and without the optimizations the differences are obvious. Not only can we reduce the number of times the time limit of three hours is exceeded (instances highlighted in orange) from 18 to 11 but also greatly improve the performance in the remaining instances. Observe that the performance never worsens with the added optimizations and high speedup values are obtained for most instances. Particularly, in instances 2, 22, 27, 30 and 35 the performance is improved by two orders of magnitude when the optimizations are added. In instances where both versions exceed the time limit the performance is sometimes also improved (a better solution is found after 3 hours) with the added optimizations. Taking instance number 8 as an example, without optimizations the best solution found after 3 hours has an objective value of 251324 meters (length of the border between territorial units in the same district). However, with the optimizations, this value is increased by 16.54% to 292899 meters. Given the high number of variables generated by the Hess model it is also common that no solution is reached within the time limit.

Table 5.5 compares the number of variables created and the execution time of two different complete models, the Hess model (with the added optimizations that we already established as useful) and the tree-based model, using two different solvers, the CPLEX Optimizer and Sat4jMoco. These results aid us in answering which is the best solver in this type of problems and which is the best complete model between the ones studied.

Evaluating the results in Table 5.5, specifically the columns with the time spent by each solver to find an optimal solution, we can clearly see that CPLEX (a commercial solver) performs better than Sat4jMoco (an open-source solver) independently of the model used. The number of instances where the time limit of three hours is exceeded (highlighted cells) is significantly reduced (especially in the tree-based model) allowing finding more global optimal solutions. Analyzing instances where both solvers

Instance Number	Time without Optimizations	Time with Optimizations
1	768.84	9.47
2	10800	30.49
3	10800	148.31
4	10800	10800
5	8.73	0.56
6	10800	501.70
7	10800	525.94
8	10800	10800
9	10800	10800
10	10800	10800
11	3.68	0.66
12	0.94	0.05
13	10800	10800
14	10800	10800
15	8.74	0.51
16	6.87	0.51
17	9.20	0.85
18	10800	2241.33
19	10800	10800
20	6.59	0.66
21	4.41	0.15
22	10800	46.14
23	10800	10800
24	0.20	0.04
25	1.25	0.12
26	252.93	4.33
27	5486.67	7.33
28	10800	10800
29	10800	10800
30	8106.49	7.52
31	3.72	0.85
32	10800	10800
33	1.78	0.76
34	14.27	0.91
35	10800	4.60

Table 5.4: Comparison between the execution times (in seconds) of the Hess model [34] with the contiguity constraints by Validi et al. [58] with and without the optimizations proposed in Section 4.5.

can find the optimum solution within the time limit, except for some small instances without much significance (such as number 11) the execution time is also decreased dramatically and instances number 2 and 35 are perfect examples. At first one can think that this is only due to the fact that a constraint solver, such as Sat4jMoco, cannot be parallelized meaning that it is only using 1 thread even though we are running on a machine with 32 cores (the same used in previous scenarios). However, when we set CPLEX to only run on 1 thread (sequentially) the performance obviously decreases but is never worse than with Sat4jMoco. An example of such would be instance number 30 under the tree-based model which running sequentially on CPLEX takes 61.56 seconds. This is a huge increase from the 4.89 seconds of the parallel version but still 2.69 times faster than using Sat4jMoco. These results repeat in other instances and in the Hess model leading us to believe that CPLEX is more efficient in this type of problems than Sat4jMoco. However, the later can be used in multi-objective combinatorial optimization (MOCO) problems, one of the focuses of this work (Sections 5.3, 5.4 and 5.5)

Instance Number	Hess Model Optimized			Tree-Based Model		
	Variables	Time CPLEX	Time Sat4jMoco	Variables	Time CPLEX	Time Sat4jMoco
1	46996	9.47	171.61	478	**0.47**	**13.54**
2	70681	30.49	3657.52	549	**0.82**	**1688.98**
3	160156	148.31	10800	820	**8.62**	10800
4	1396234	10800	10800	2630	10800	10800
5	10021	0.56	2.60	226	**0.38**	**0.63**
6	183883	501.7	10800	962	**5.47**	10800
7	236930	525.94	10800	1047	**1.34**	10800
8	733279	10800	10800	1792	**26.32**	10800
9	1788682	10800	10800	3178	**1882.42**	10800
10	18797828	10800	10800	11396	10800	10800
11	7080	0.66	1.89	192	**0.56**	**0.54**
12	4252	0.05	0.77	145	**0.04**	**0.39**
13	1249432	10800	10800	2528	**575.96**	10800
14	3751519	10800	10800	5134	10800	10800
15	10021	0.51	3.10	226	**0.38**	**0.61**
16	14620	**0.51**	3.14	356	1.01	**0.89**
17	10807	0.85	3.05	234	**0.42**	**0.65**
18	870865	2241.33	10800	2015	**20.07**	10800
19	688327	10800	10800	1988	**35.2**	10800
20	13755	**0.66**	3.45	375	5.63	**1.96**
21	11500	**0.15**	1.53	350	0.69	**0.46**
22	62838	**46.14**	10800	696	53.26	10800
23	2630605	10800	10800	4120	10800	10800
24	13304	0.04	1.61	222	**0.01**	**0.35**
25	4981	**0.12**	0.70	197	0.40	**0.36**
26	14206	4.33	**4.71**	321	**1.32**	6.71
27	53167	7.33	16.54	602	**3.97**	**5.20**
28	371749	10800	10800	1546	**711.26**	10800
29	17961013	10800	10800	9278	10800	10800
30	39293	7.52	228.08	539	**4.89**	**165.57**
31	7080	0.85	1.47	240	**0.48**	**0.99**
32	371749	10800	10800	1560	**225.03**	10800
33	3718	0.76	0.85	160	**0.49**	**0.44**
34	10021	0.91	3.07	251	**0.41**	**0.90**
35	58226	4.60	695.33	588	**2.20**	**286.41**

Table 5.5: Comparison between the number of variables created and the execution times (in seconds) of the optimized Hess model and the proposed Tree-Based model using the CPLEX Optimizer and the Sat4jMoco solvers. Highlighted in orange are time limits exceeded using CPLEX and, in yellow, using Sat4jMoco. Results in bold highlight the faster model for each instance and solver.

Moving on to the third research question, is our proposed tree-based model better than the Hess model (even with the added optimizations)? To answer this question we need only to analyze Table 5.5, and compare, for each instance and solver, the execution time of both complete models. In order to aid the visualization, highlighted in bold is the fastest time between the models, for each solver and instance.

Using the CPLEX Optimizer, we can see that there are only 5 out of 35 instances – numbers 16, 20, 21, 22 and 25 – where the execution time is worse (speedup values below 1) using the tree-based model. Although 4 of these 5 instances (numbers 16, 20, 21 and 25) are relatively easy to solve (the optimal solution can be found in a few seconds) and the performance between both models is similar in instance 22. On the other hand, the tree-based model performs decisively better (over 100 times faster) on instances number 7, 8, 18 and 19. The fact that the tree-based model performs better on instances which are harder to solve (high values of territorial units) makes its results even more significant because in small instances where the optimal solution can be found in mere seconds it is relatively insignificant if it takes 1 second or 5 seconds to reach the optimal solution. Last but not least, the tree-based model only exceeds the time limit in 5 instances instead of 11.

Using the Sat4jMoco solver, the tree-based formulation is faster in all instances except in instance number 26. The speedup values are lower than when using CPLEX but significant increases in performance can be noticed in numbers 1, 2 and 35. Given the time limit of three hours both models finish solving the same number of instances (19 instances).

The large discrepancy in execution times in instances where the number of territorial units is higher can be attributed to the number of binary variables necessary for each model. Comparing the columns with the number of variables created by each model we can see that the Hess model creates two to three orders of magnitude more variables than the tree-based model. Remember that the number of variables created for contiguity in the Hess model equal $n^2 \times A$ (the flow variables in Figure 3.3), where n is the number of territorial units and A is double the number of edges E. Furthermore, notice that $E \geq n - 1$ in contiguous maps. Meanwhile, the number of variables in the tree-based model was estimated in Section 4.6 to be closer to n^2 and we are now in position to confirm that the estimate was correct. A higher number of variables also requires a higher number of constraints (in the same order of magnitude) hence creating large branch and cut trees which may also cause memory problems. Therefore, from the obtained results we can conclude that tree-based model performs better than the Hess model, even with the added optimizations.

The last topic we must evaluate is the performance of the shortest-path model against the tree-based model. Since the shortest-path model is an incomplete model (as explained in Section 4.3.2) we already know that occasionally it does not produce the global optimal solution. However, we are interesting in studying how often that is the case and how far is it from the optimum is such cases. On the other hand, concerning execution times, it performs better than complete models because the search space is smaller.

We start by running the same instances using the CPLEX Optimizer for both models, tracking the number of binary variables created, the execution time and the objective value achieved. The results are presented in Table 5.6 with the objective value increase when using a complete model, for each experimental instance. We are also interested in studying the different performances in multi-objective combinatorial optimization (MOCO) problems since it is possible that in more complex problems the results differ. Therefore, the same instances are generated with one more objective function: maximizing the results of a party (along with compactness). These MOCO instances are solved using Sat4jMoco (albeit faster, CPLEX can only be used in SOO problems) and the execution times as well as the hypervolume increase/decrease to a static strictly dominated reference point using the tree-based model is registered in Table 5.7, for each instance.

Analyzing the results in Table 5.6 one difference is immediately obvious: the shortest-path model does not exceed the three hour time limit in any instance, using the CPLEX Optimizer. Meanwhile, the

Instance Number	Tree-Based Model		Shortest-Path Model		Compactness Increase
	Variables	Time CPLEX	Variables	Time CPLEX	
1	478	0.47	140	0.48	0.00%
2	549	0.82	162	0.30	0.00%
3	820	8.62	212	0.52	**1.33%**
4	2630	10800	444	1.96	TLE
5	226	0.38	78	0.26	0.00%
6	962	5.47	226	0.64	0.00%
7	1047	1.34	246	0.75	0.00%
8	1792	26.32	366	1.61	0.00%
9	3178	1882.42	504	3.28	**1.80%**
10	11396	10800	1108	4.80	TLE
11	192	0.56	72	0.34	0.00%
12	145	0.04	56	0.03	**5.15%**
13	2528	575.96	432	1.77	**1.29%**
14	5134	10800	951	18.44	TLE
15	226	0.38	78	0.01	**6.13%**
16	356	1.01	220	0.81	0.00%
17	234	0.42	82	0.02	0.00%
18	2015	20.07	380	0.74	0.00%
19	1988	35.20	525	5.20	**0.67%**
20	375	5.63	270	0.71	0.00%
21	350	0.69	240	0.70	0.00%
22	696	53.26	468	11.42	0.00%
23	4120	10800	1710	8939.96	TLE
24	222	0.01	44	0.01	0.00%
25	197	0.40	124	0.66	0.00%
26	321	1.32	184	0.37	0.00%
27	602	3.97	268	0.98	0.00%
28	1546	711.26	572	6.08	**0.72%**
29	9278	10800	2212	1839.54	TLE
30	539	4.89	325	1.28	0.00%
31	240	0.48	144	0.24	0.00%
32	1560	225.03	715	19.21	**0.56%**
33	160	0.49	84	0.32	0.00%
34	251	0.41	117	0.42	0.00%
35	588	2.20	296	0.53	**5.22%**

Table 5.6: Comparison between the execution times (in seconds) and the quality of the results produced by the tree-based model (complete) and the shortest-path model (incomplete) for SOO problems using the CPLEX Optimizer. Highlighted in orange are instances where the time limit was exceeded. Highlighted in bold is the compactness increase, whenever existing.

tree-based model, finds the global optimal solution for 30 out of 35 instances. As far as the execution times are concerned, these are usually sharply reduced using the shortest-path model. The most noticeable cases are in instances 9, 13 and 28 (over 100 times faster using the shortest-path formulation) along with instances 4, 10 and 14 where the tree-based model exceeds the time lime but the shortest-path method manages to find its optimum solutions swiftly. Unfortunately, the low execution times come with a loss in objective value (compactness) in 30% of the instances, i.e., in 9 instances out of the 30 that could be solved, the solution is more compact using complete methods. Without disregarding this percentage, the maximum increase in compactness using the tree-based model is only 6.13% (instance 15) meaning that the shortest-path model is still producing reasonable solutions. As an example, the second largest gap (instance 35) is the one presented in Figure 4.2 and, visually, it is extremely hard to

Instance Number	Tree-based Time	Shortest-Path Time	Hypervolume Increase
1	10.57	0.61	0.00%
2	4910.57	0.71	0.00%
3	10800	0.76	1.33%
4	10800	2.53	0.10%
5	0.55	0.30	0.00%
6	10800	0.40	13.42%
7	10800	0.75	0.20%
8	10800	3.48	-1.50%
9	10800	2.03	0.76%
10	10800	7.05	-0.28%
11	0.51	0.31	0.00%
12	0.31	0.31	5.15%
13	10800	2.75	0.71%
14	10800	10800	-3.54%
15	0.58	0.31	6.13%
16	0.91	0.49	0.00%
17	0.53	0.31	0.00%
18	10800	0.95	0.00%
19	10800	413.98	-2.00%
20	2.74	1.08	0.00%
21	0.56	0.55	0.00%
22	10800	10800	-0.26%
23	10800	10800	-15.58%
24	0.29	0.29	0.00%
25	0.33	0.43	0.00%
26	7.99	0.54	0.00%
27	9.44	0.45	0.00%
28	10800	778.92	-4.03%
29	10800	10800	-100.00%
30	221.93	10.41	0.00%
31	1.05	0.48	0.00%
32	10800	10800	-1.82%
33	0.37	0.31	0.00%
34	0.90	0.36	0.51%
35	288.05	0.45	22.24%

Table 5.7: Comparison between the execution times (in seconds) and the quality of the results produced by the tree-based model (complete) and the shortest-path model (incomplete) for MOO problems using Sat4jMoco. Highlighted in yellow are instances where the time limit was exceeded. Highlighted in green or red are upgrades or downgrades, respectively, to the hypervolume using the tree-based model.

tell which one is the most compact. In the five instances where the time limit is exceeded it is impossible to compare the quality of the solutions found because CPLEX could not even find a solution to the problem within the time limit. Contrarily to the previous test with the Hess model, the time differences can no longer be attributed to the number of variables since the difference is not so pronounced and significantly lower than between the tree-based model and the Hess model (see Table 5.5). The most likely explanation is that the shortest-path model by only allowing the shortest-paths between territorial units in the same cluster ends up cutting (too) many pairings that must still be considered in the tree-based model.

The results for MOCO instances, solved using Sat4jMoco, are presented in Table 5.7. For each instance, the execution time of each model is registered. The dominated area (hypervolume) to the same strictly dominated point (the point at (0, 8)) is also calculated and the hypervolume increase using the tree-based model presented in the last column of Table 5.7. The number of time limits exceeded (cells highlighted in yellow) is significantly reduced from 16 to 5 using the shortest-path model and, once again, the execution times are drastically lower. Just as in SOO scenarios there are instances where the time limit is exceeded using the complete model but an optimum solution is found using the incomplete model in mere seconds (such as in instances 3, 6 or 7). Curiously, comparing the dominated area (hypervolume) of each contiguity method to the same strictly dominated point we see that many of the instances where the tree-based model fails to finish within 3 hours and the execution time is seriously reduced using the shortest-path model (instances 3, 4, 6, 9) are precisely the ones where the quality of the solutions is improved using the tree-based model (cells highlighted in green). These results suggest that the shortest-path model is only so much faster because it is cutting many solutions that end up constituting the Pareto front. On the other hand, when both approaches exceed the time limit (instances 14, 22, 23, 29 and 32) the shortest-path model always manages to find the best solution (cells highlighted in red) implying that, although it may be cutting the global optimal solutions, it is still considering good solutions and finding them faster than the tree-based model, proving to be the best option in the hardest to solve instances. Instance 29 is the only where, even using a constraint solver, a possible solution cannot be found within the time limit established using the tree-based model, hence the 100% decrease in hypervolume.

In conclusion, the tree-based model proves to be the best option in most instances and given enough time would always yield better results. However, and particularly in the most difficult to solve instances (such as 23 and 29) containing large numbers of territorial units to be divided between many clusters (large values of n and k) incomplete methods prove to be the best option achieving significantly better results both in SOO and MOCO problems.

Chapter 6

Conclusions and Future Work

The political districting problem is a computationally hard one that has been studied since the 1960s. Solving this problem has the potential of eradicating gerrymandering in single-member districts, fulfilling the purpose of elections in democracies: let the voters choose the politicians and not the politicians choose the voters. The electoral districting problem is NP-Complete meaning that the larger the number of territorial units considered and the number of districts to create, the harder it is to find the global optimal solution to the problem. However, we believe that the importance of fair elections in democracies justifies the computational power necessary to find such solutions.

Recent developments in hardware and algorithms to solve optimization problems increased the number of instances that can be solved using complete methods. Yet, just like many authors before, we still face largely densely populated areas that are extremely hard to redistrict. An option in such scenarios is to use incomplete methods that although incapable of finding global optimums, manage to produce solutions that still create compact, contiguous and similarly populated electoral districts.

As Governor of California, Ronald Reagan said: *"There is only one way to do reapportionment – feed into the computer all the factors except political registration"*.

In this work that is exactly the objective, through the development of a novel and efficient multi-objective combinatorial optimization model with the ability of creating electoral districts that are: compact, contiguous and give voters similar popular representation. Two formulations to create contiguous districts are presented, a complete one (capable of finding global optimal solutions) and an incomplete, but dramatically faster version. The incomplete version justifies its use in the most difficult to solve instances since our experimental results show that it is able to find good solutions in cases where complete methods sometimes fail to even produce a feasible solution. Optimizations to the model that can also be adapted to classical approaches to the redistricting problem are also presented and managed to drastically improve its computational performance. The developed model is versatile and can also be used to create gerrymandered maps using data from previous elections, allowing the study of the effects of gerrymandering in electoral maps. Experimental results comparing the proposed model with a classical approach to the political districting problem, show that our model significantly improves the performance, even when the proposed optimizations are added.

Following the propositions from the main parties to change the Portuguese electoral system to one using single-member districts, we test our model in different scenarios using only official geographical data. The first complete electoral maps for Portugal using single-member districts under two different voting systems (the first-past-the-post and the parallel voting) are created and the results show that the model can deliver compact and contiguous solutions within the population boundaries while conforming with current administrative divisions. Furthermore, an algorithm to redistrict large densely populated areas by making a first partition using existing administrative divisions is proposed and helps to find solutions in the hardest to solve problems.

Additionally, using the gerrymandering capacities of the model and official electoral results from previous legislative elections, the impact of gerrymandering is studied. Testing different scenarios, contemplating toss-up districts and bipartisan systems, the results always show that gerrymandering significantly affects the compactness of the districts, creating the famous odd shapes that coined the term. Moreover, it demonstrates how the electoral results can be greatly affected once a party tries to maximize its electoral results, altering the course of elections and damaging the pillars of democracy.

6.1 Future Work

Since most countries define tighter population boundaries to the theoretical best number of individuals in each electoral district, it would be interesting to study how would the results change if this value was set to 10% or even 5%, instead of the 25% considered in our experimental tests. Additionally, it would also allow studying the effects of such change on the compactness of the electoral districts.

Our formulation of the problem introduces a new method to generate compact districts by maximizing the length of the borders between territorial units in the same district and the tests carried showed exceptional results. Using a compactness measure, such as the Reock score[47], one can compare the compactness of the electoral districts created using our proposed method and more classical ones, such as minimizing the sum of the Euclidean distances between the centroid of each territorial unit and the root of the electoral district it belongs to.

The problem of redistricting electoral areas in the United States of America (USA) is by far where research most work has been conducted in the area. This occurs in part because the USA is the country where most cases of gerrymandering have surfaced and electoral results have been changed because of it. Remember that conformity to administrative subdivisions is not a requirement and census tracts (the smallest territorial divisions) are used in order to create the most equal districts in terms of population as possible. This characteristics compel researchers to deal with more territorial units, a major factor in the complexity of the problem. Contrarily, as tighter population boundaries can be defined, the search space is reduced. Applying our model to this particular problem would allow for a better comparison with previous approaches.

Furthermore, in the United Stated of America where electoral maps were proven to have been gerrymandered (Figure 2.3) it would be extremely interesting to use the gerrymandering capacities of the model and compare the results with those electoral maps. For instance, could one replicate or even further skew the results of the 2018 House of Representatives elections in North Carolina?

Most proposed models in the area are single-objective ones and a weight has to be assigned in order to maximize compactness while minimizing population differences between electoral districts. This is a typical objective function in electoral districting problems, especially when studying the problem of the United States of America. A multi-objective combinatorial model such as the one proposed, allows defining an additional objective function minimizing the population differences between electoral districts at the same time as compactness is maximized. Using the proposed model one could obtain the Pareto front for scenarios in the United States of America and compare the results with single-objective ones.

Finally, more optimizations are still to be explored with the goal of increasing the performance, consequentially, reducing the number of instances that cannot be solved in useful time.

Bibliography

[1] *Cooper v. Harris, 581 U.S. (2017)*, volume 581. Supreme Court, 2017. URL https://supreme.justia.com/cases/federal/us/581/15-1262/.

[2] M. Altman. Is automation the answer: the computational complexity of automated redistricting. *Rutgers Computer and Law Technology Journal*, 23:81–142, 1997. URL http://maltman.hmdc.harvard.edu/pubpapers.shtml.

[3] J. Argelich, D. L. Berre, I. Lynce, J. P. M. Silva, and P. Rapicault. Solving linux upgradeability problems using boolean optimization. In *Proceedings First International Workshop on Logics for Component Configuration, LoCoCo 2010, Edinburgh, UK, 10th July 2010*, pages 11–22, 2010. doi: 10.4204/EPTCS.29.2. URL https://doi.org/10.4204/EPTCS.29.2.

[4] F. Bação, V. S. Lobo, and M. Painho. Applying genetic algorithms to zone design. *Soft Comput.*, 9(5):341–348, 2005. doi: 10.1007/s00500-004-0413-4. URL https://doi.org/10.1007/s00500-004-0413-4.

[5] S. Bangia, C. V. Graves, G. Herschlag, H. S. Kang, J. Luo, J. C. Mattingly, and R. Ravier. Redistricting: Drawing the line, 2017.

[6] J. J. Bartholdi III, C. A. Tovey, and M. A. Trick. How hard is it to control an election? *Mathematical and Computer Modelling*, 16(8-9):27–40, 1992.

[7] P. K. Bergey, C. T. Ragsdale, and M. Hoskote. A decision support system for the electrical power districting problem. *Decision Support Systems*, 36(1):1 – 17, 2003. ISSN 0167-9236. doi: https://doi.org/10.1016/S1344-6223(02)00033-0. URL http://www.sciencedirect.com/science/article/pii/S1344622302000330.

[8] N. Betzler and J. Uhlmann. Parameterized complexity of candidate control in elections and related digraph problems. *Theor. Comput. Sci.*, 410(52):5425–5442, 2009. doi: 10.1016/j.tcs.2009.05.029. URL https://doi.org/10.1016/j.tcs.2009.05.029.

[9] A. Billionnet. Designing an optimal connected nature reserve. *Applied Mathematical Modelling*, 36(5):2213 – 2223, 2012. ISSN 0307-904X. doi: https://doi.org/10.1016/j.apm.2011.08.002. URL http://www.sciencedirect.com/science/article/pii/S0307904X11005026.

[10] L. Bodin. A districting experiment with a clustering algorithm. *Annals of the New York Academy of Sciences*, 219:209 – 214, 12 2006. doi: 10.1111/j.1749-6632.1973.tb41400.x.

[11] M. H. Browdy. Simulated annealing: An improved computer model for political redistricting. *Yale Law & Policy Review*, 8(1):163–179, 1990. ISSN 07408048. URL `http://www.jstor.org/stable/40239326`.

[12] R. Carvajal, M. Constantino, M. Goycoolea, J. P. Vielma, and A. Weintraub. Imposing connectivity constraints in forest planning models. *Oper. Res.*, 61(4):824–836, 2013. doi: 10.1287/opre.2013.1183. URL `https://doi.org/10.1287/opre.2013.1183`.

[13] J. Chen and J. Rodden. Unintentional gerrymandering: Political geography and electoral bias in legislatures. *Quarterly Journal of Political Science*, 8, 06 2013. doi: 10.1561/100.00012033.

[14] M. Constantino, I. Martins, and J. G. Borges. A new mixed-integer programming model for harvest scheduling subject to maximum area restrictions. *Operations Research*, 56(3):542–551, 2008.

[15] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Second Edition*. The MIT Press and McGraw-Hill Book Company, 2001. ISBN 0-262-03293-7.

[16] G. W. Cox. *Making votes count: strategic coordination in the world's electoral systems*. Cambridge University Press, 1997.

[17] M. B. da Cruz. A revisão falhada do sistema eleitoral. *Análise Social*, 35(154/155):45–53, 2000. ISSN 00032573, 21822999. URL `http://www.jstor.org/stable/41011635`.

[18] S. J. D'Amico, S. Wang, R. Batta, and C. M. Rump. A simulated annealing approach to police district design. *Computers & OR*, 29(6):667–684, 2002. doi: 10.1016/S0305-0548(01)00056-9. URL `https://doi.org/10.1016/S0305-0548(01)00056-9`.

[19] D. Datta, J. Malczewski, and J. R. Figueira. Spatial aggregation and compactness of census areas with a multiobjective genetic algorithm: a case study in canada. *Environment and Planning B: Planning and Design*, 39(2):376–392, 2012.

[20] D. Datta, J. Figueira, A. Gourtani, and A. Morton. Optimal administrative geographies: an algorithmic approach. *Socio-Economic Planning Sciences*, 47(3):247–257, 2013.

[21] V. D'Hondt. *Système pratique et raisonné de représentation proportionnelle*. librairie C. Muquardt, Merzbach et Falk, éditeurs, 1882. doi: https://doi.org/10.3931/e-rara-39876.

[22] G. Erdélyi, M. R. Fellows, J. Rothe, and L. Schend. Control complexity in bucklin and fallback voting: A theoretical analysis. *J. Comput. Syst. Sci.*, 81(4):632–660, 2015. doi: 10.1016/j.jcss.2014.11.002. URL `https://doi.org/10.1016/j.jcss.2014.11.002`.

[23] G. Erdélyi, E. Hemaspaandra, and L. A. Hemaspaandra. More natural models of electoral control by partition. In T. Walsh, editor, *Algorithmic Decision Theory - 4th International Conference, ADT 2015, Lexington, KY, USA, September 27-30, 2015, Proceedings*, volume 9346 of *Lecture Notes in*

Computer Science, pages 396–413. Springer, 2015. doi: 10.1007/978-3-319-23114-3_24. URL https://doi.org/10.1007/978-3-319-23114-3_24.

[24] P. Faliszewski, E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Llull and copeland voting computationally resist bribery and constructive control. *J. Artif. Intell. Res.*, 35:275–341, 2009. doi: 10.1613/jair.2697. URL https://doi.org/10.1613/jair.2697.

[25] D. M. Farrell. *Electoral systems: A comparative introduction*. Macmillan International Higher Education, 2011.

[26] J. A. Ferland and G. Guénette. Decision support system for the school districting problem. *Operations Research*, 38(1):15–21, 1990. doi: 10.1287/opre.38.1.15. URL https://doi.org/10.1287/opre.38.1.15.

[27] E. Forrest. Apportionment by computer. *American Behavioral Scientist*, 8(4):23–23, 1964. doi: 10.1177/000276426400800407. URL https://doi.org/10.1177/000276426400800407.

[28] M. Gallagher. Proportionality, disproportionality and electoral system. *Electoral studies*, 10(1): 33–51, 1991.

[29] R. S. Garfinkel and G. L. Nemhauser. Optimal political districting by implicit enumeration techniques. *Management Science*, 16(8):B495–B508, 1970. ISSN 00251909, 15265501. URL http://www.jstor.org/stable/2628656.

[30] B. Grofman and G. King. The future of partisan symmetry as a judicial test for partisan gerrymandering after lulac v. perry. *Election Law Journal: Rules, Politics, and Policy*, 6(1):2–35, 2003. URL https://dash.harvard.edu/handle/1/4214910.

[31] O. Guest, F. J. Kanayet, and B. C. Love. Gerrymandering and computational redistricting. *Journal of Computational Social Science*, 2(2):119–131, Jul 2019. ISSN 2432-2725. doi: 10.1007/s42001-019-00053-9. URL https://doi.org/10.1007/s42001-019-00053-9.

[32] D. Hayes and S. Mckee. The participatory effects of redistricting. *American Journal of Political Science*, 53:1006 – 1023, 10 2009. doi: 10.1111/j.1540-5907.2009.00413.x.

[33] E. Hemaspaandra, L. A. Hemaspaandra, and J. Rothe. Anyone but him: The complexity of precluding an alternative. *Artif. Intell.*, 171(5-6):255–285, 2007. doi: 10.1016/j.artint.2007.01.005. URL https://doi.org/10.1016/j.artint.2007.01.005.

[34] S. W. Hess, J. B. Weaver, H. J. Siegfeldt, J. N. Whelan, and P. A. Zitlau. Nonpartisan political redistricting by computer. *Operations Research*, 13(6):998–1006, 1965. ISSN 0030364X, 15265463. URL http://www.jstor.org/stable/167658.

[35] C. Hunt. When does redistricting matter? changing conditions and their effects on voter turnout. *Electoral Studies*, 54, 05 2018. doi: 10.1016/j.electstud.2018.05.007.

[36] C. Jefferson and K. E. Petrie. Automatic generation of constraints for partial symmetry breaking. In *Principles and Practice of Constraint Programming – CP 2011*, pages 729–743, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg. ISBN 978-3-642-23786-7.

[37] C. L. Johnson. Statistics of the congressional election from official sources for the election of november 6, 2018. `https://history.house.gov/Institution/Election-Statistics`, February 2019.

[38] M. Kennedy and A. Hurt. Supreme court rejects 2 congressional districts in north carolina, May 2017. URL `https://www.npr.org/sections/thetwo-way/2017/05/22/529503045/supreme-court-rejects-2-congressional-districts-in-north-carolina`. Accessed: 2020-11-17.

[39] L. Kotthoff, B. O'Sullivan, S. S. Ravi, and I. Davidson. Complex Clustering Using Constraint Programming: Modelling Electoral Map Creation. In *14th International Workshop on Constraint Modelling and Reformulation*, Sept. 2015.

[40] Z. Li, R.-S. Wang, and Y. Wang. A quadratic programming model for political districting problem. In *Proceedings of The First International Symposium on Optimization and Systems Biology (OSB'07)*, 12 2019.

[41] H. Liu, H. Feng, D. Zhu, and J. Luan. Parameterized computational complexity of control problems in voting systems. *Theor. Comput. Sci.*, 410(27-29):2746–2753, 2009. doi: 10.1016/j.tcs.2009.04.004. URL `https://doi.org/10.1016/j.tcs.2009.04.004`.

[42] D. Lublin. *The paradox of representation: Racial gerrymandering and minority interests in Congress*. Princeton University Press, 1999.

[43] M. E. McDill, S. A. Rebain, and J. Braze. Harvest scheduling with area-based adjacency constraints. *Forest Science*, 48(4):631–642, 2002.

[44] C. Mears, M. Garcia de la Banda, M. Wallace, and B. Demoen. A novel approach for detecting symmetries in csp models. In *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, pages 158–172, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-68155-7.

[45] A. Mehrotra, E. L. Johnson, and G. L. Nemhauser. An optimization based heuristic for political districting. *Management Science*, 44(8):1100–1114, 1998. ISSN 00251909, 15265501. URL `http://www.jstor.org/stable/2634689`.

[46] A. T. Murray. Spatial restrictions in harvest scheduling. *Forest Science*, 45(1):45–52, 1999.

[47] R. G. Niemi, B. Grofman, C. Carlucci, and T. Hofeller. Measuring compactness and the role of a compactness standard in a test for partisan and racial gerrymandering. *The Journal of Politics*, 52 (4):1155–1181, 1990. ISSN 00223816, 14682508. URL `http://www.jstor.org/stable/2131686`.

[48] J. Oehrlein and J.-H. Haunert. A cutting-plane method for contiguity-constrained spatial aggregation. *Journal of Spatial Information Science*, 2017, 12 2017. doi: 10.5311/JOSIS.2017.15.379.

[49] H. Önal and R. A. Briers. Optimal selection of a connected reserve network. *Oper. Res.*, 54(2): 379–388, 2006. doi: 10.1287/opre.1060.0272. URL `https://doi.org/10.1287/opre.1060.0272`.

[50] H. Önal, Y. Wang, S. T. M. Dissanayake, and J. D. Westervelt. Optimal design of compact and functionally contiguous conservation management areas. *Eur. J. Oper. Res.*, 251(3):957–968, 2016. doi: 10.1016/j.ejor.2015.12.005. URL `https://doi.org/10.1016/j.ejor.2015.12.005`.

[51] E. C. Reock. A note: Measuring compactness as a requirement of legislative apportionment. *Midwest Journal of Political Science*, 5(1):70–74, 1961. ISSN 00263397. URL `http://www.jstor.org/stable/2109043`.

[52] F. Ricca and B. Simeone. Political districting: Traps, criteria, algorithms and tradeoffs. *Ricerca Operativa AIRO*, 27:81–119, 01 1997.

[53] F. Ricca, A. Scozzari, and B. Simeone. Political districting: From classical models to recent approaches. *Annals of Operations Research*, 204:271–299, 04 2013. doi: 10.1007/s10479-012-1267-2.

[54] W. H. Riker. The two-party system and duverger's law: An essay on the history of political science. *American Political Science Review*, 76(4):753–766, 1982. doi: 10.1017/S0003055400189580.

[55] J. E. Schwartzberg. Reapportionment, gerrymanders, and the notion of compactness. *Minnesota Law Review*, 50(1701):443, 1966. URL `https://scholarship.law.umn.edu/mlr/1701/`.

[56] R. Swamy, D. King, and S. Jacobson. Multi-objective optimization for political districting: A scalable multilevel approach. *Optimization Online*, 2019.

[57] F. Tavares-Pereira, J. R. Figueira, V. Mousseau, and B. Roy. Multiple criteria districting problems. *Ann. Oper. Res.*, 154(1):69–92, 2007. doi: 10.1007/s10479-007-0181-5. URL `https://doi.org/10.1007/s10479-007-0181-5`.

[58] H. Validi, A. Buchanan, and E. Lykhovyd. Imposing contiguity constraints in political districting models. 2020.

[59] W. Vickrey. On the prevention of gerrymandering. *Political Science Quarterly*, 76(1):105–110, 1961. ISSN 00323195. URL `http://www.jstor.org/stable/2145973`.

[60] T. Walsh. General symmetry breaking constraints. In *International Conference on Principles and Practice of Constraint Programming*, pages 650–664. Springer, 2006.

[61] H. P. Young. Measuring the compactness of legislative districts. *Legislative Studies Quarterly*, 13 (1):105–115, 1988. ISSN 03629805. URL `http://www.jstor.org/stable/439947`.

[62] Y. Yuan and W. Banzhaf. ARJA: automated repair of java programs via multi-objective genetic programming. *CoRR*, abs/1712.07804, 2017. URL `http://arxiv.org/abs/1712.07804`.

[63] E. Zitzler. Evolutionary algorithms for multiobjective optimization : Methods and applications. *PhD Thesis, Swiss Federal Institute of Technology, ETH Zurich, Switzerland*, November 1999.

[64] A. A. Zoltners and P. Sinha. Sales territory alignment: A review and model. *Management Science*, 29(11):1237–1256, 1983. ISSN 00251909, 15265501. URL http://www.jstor.org/stable/2630904.

Appendix A

Electoral Results

Electoral Results - Section 5.3 (Scenario 3)						
Instance Number	Unbiased		Max PS		Max PSD	
	PS	PSD	PS	PSD	PS	PSD
1	1	1	1	1	0	2
2	2	0	2	0	2	0
3	2	0	2	0	1	1
4	0	2	1	1	0	2
(01)	5	3	6	2	3	5
5 (02)	2	0	2	0	2	0
6	1	1	2	0	1	1
7	1	1	2	0	1	1
8	2	0	2	0	2	0
9	0	2	0	2	0	2
10	1	1	1	1	1	1
(03)	5	5	7	3	5	5
11 (04)	0	2	0	2	0	2
12 (05)	2	0	2	0	2	0
13	2	0	2	0	1	1
14	3	0	3	0	3	0
(06)	5	0	5	0	4	1
15 (07)	2	0	2	0	2	0
16 (08)	5	0	5	0	5	0
17 (09)	2	0	2	0	1	1
18	1	1	2	0	1	1
19	1	2	2	1	0	3
(10)	2	3	4	1	1	4
20	6	0	6	0	6	0
21	5	1	5	1	5	1
22	5	1	6	0	3	3
23	6	0	6	0	6	0
(11)	22	2	23	1	20	4
24 (12)	1	0	1	0	1	0
25	2	2	3	1	2	2
26	4	0	4	0	4	0
27	3	1	3	1	3	1
28	3	1	4	0	3	1
29	3	1	4	0	3	1
(13)	15	5	18	2	16	4
30 (14)	4	1	4	1	4	1
31	4	0	4	0	4	0
32	5	0	5	0	5	0
(15)	9	0	9	0	9	0
33 (16)	1	2	2	1	1	2
34 (17)	1	2	2	1	0	3
35 (18)	1	3	3	1	1	3
TOTAL	84	28	97	15	76	36

Table A.1: Complete electoral results for the districting in scenario 3 (Section 5.3). Highlighted in gray are the sums of all electoral districts won in each Portuguese region (ID between parentheses), per party. The cells highlighted in yellow represent instances where the time limit was exceeded, hence the presented results are for the best solution found after 3 hours (which might not be the global optimum).

Electoral Results - Section 5.4 (Scenario 4)									
Instance Number	Unbiased			Max PS			Max PSD		
	PS	PSD	Toss-Up	PS	PSD	Toss-Up	PS	PSD	Toss-Up
1	0	1	1	0	0	2	0	1	1
2	0	0	2	0	0	2	0	0	2
3	0	0	2	0	0	2	0	0	2
4	0	0	2	0	0	2	0	1	1
(01)	0	1	7	0	0	8	0	2	6
5 (02)	2	0	0	2	0	0	2	0	0
6	0	0	2	0	0	2	0	0	2
7	0	0	2	0	0	2	0	0	2
8	1	0	1	2	0	0	1	0	1
9	0	1	1	0	0	2	0	1	1
10	0	0	2	0	0	2	0	0	2
(03)	1	1	8	2	0	8	1	1	8
11 (04)	0	0	2	0	0	2	0	0	2
12 (05)	1	0	1	2	0	0	1	0	1
13	1	0	1	1	0	1	0	0	2
14	2	0	1	2	0	1	1	0	2
(06)	3	0	2	3	0	2	1	0	4
15 (07)	2	0	0	2	0	0	2	0	0
16 (08)	3	0	2	5	0	0	3	0	2
17 (09)	0	0	2	0	0	2	0	0	2
18	0	0	2	1	0	1	0	0	2
19	0	1	2	1	0	2	0	2	1
(10)	0	1	4	2	0	3	0	2	3
20	4	0	2	5	0	1	3	0	3
21	4	0	2	4	0	2	4	0	2
22	2	0	4	2	0	4	0	0	6
23	4	0	2	6	0	0	4	0	2
(11)	14	0	10	17	0	7	11	0	13
24 (12)	1	0	0	1	0	0	1	0	0
25	0	0	4	0	0	4	0	0	4
26	1	0	3	2	0	2	0	0	4
27	1	0	3	1	0	3	1	0	3
28	0	0	4	0	0	4	0	1	3
29	0	0	4	1	0	3	0	0	4
(13)	2	0	18	4	0	16	1	1	18
30 (14)	3	0	2	4	0	1	3	1	1
31	4	0	0	4	0	0	4	0	0
32	5	0	0	5	0	0	5	0	0
(15)	9	0	0	9	0	0	9	0	0
33 (16)	0	0	3	0	0	3	0	0	3
34 (17)	0	0	3	0	0	3	0	0	3
35 (18)	0	0	4	0	0	4	0	0	4
TOTAL	41	3	68	53	0	59	35	7	70

Table A.2: Complete electoral results for the districting in scenario 4 (Section 5.4). Highlighted in gray are the sums of all electoral districts won in each Portuguese region (ID between parentheses), per party (or toss-up). The cells highlighted in yellow represent instances where the time limit was exceeded, hence the presented results are for the best solution found after 3 hours (which might not be the global optimum).

Electoral Results - Section 5.5 (Scenario 5)									
Instance Number	Unbiased			Max Left			Max Right		
	Left	Right	Toss-Up	Left	Right	Toss-Up	Left	Right	Toss-Up
1	0	1	1	0	0	2	0	0	2
2	1	0	1	1	0	1	0	0	2
3	0	0	2	1	0	1	0	2	0
4	0	0	2	0	0	2	0	0	2
(01)	1	1	6	2	0	6	0	2	6
5 (02)	2	0	0	2	0	0	2	0	0
6	0	0	2	1	0	1	0	0	2
7	1	0	1	2	0	0	1	0	1
8	2	0	0	2	0	0	1	0	1
9	0	1	1	0	0	2	0	1	1
10	0	0	2	0	0	2	0	0	2
(03)	3	1	6	5	0	5	2	1	7
11 (04)	0	0	2	0	0	2	0	0	2
12 (05)	1	0	1	2	0	0	1	0	1
13	1	0	1	2	0	0	1	0	1
14	2	0	1	3	0	0	2	0	1
(06)	3	0	2	5	0	0	3	0	2
15 (07)	2	0	0	2	0	0	2	0	0
16 (08)	5	0	0	5	0	0	5	0	0
17 (09)	0	0	2	0	0	2	0	0	2
18	0	0	2	1	0	1	0	0	2
19	0	1	2	1	0	2	0	2	1
(10)	0	1	4	2	0	3	0	2	3
20	6	0	0	6	0	0	6	0	0
21	5	0	1	5	0	1	4	0	2
22	4	0	2	6	0	0	2	0	4
23	5	0	1	6	0	0	4	0	2
(11)	20	0	4	23	0	1	16	0	8
24 (12)	1	0	0	1	0	0	1	0	0
25	3	0	1	3	0	1	2	0	2
26	4	0	0	4	0	0	3	0	1
27	3	0	1	3	0	1	1	0	3
28	3	0	1	4	0	0	2	0	2
29	0	0	4	1	0	3	0	0	4
(13)	13	0	7	15	0	5	8	0	12
30 (14)	4	0	1	4	0	1	4	1	0
31	4	0	0	4	0	0	4	0	0
32	5	0	0	5	0	0	5	0	0
(15)	9	0	0	9	0	0	9	0	0
33 (16)	1	0	2	1	0	2	1	1	1
34 (17)	0	0	3	0	0	3	0	1	2
35 (18)	0	0	4	0	0	4	0	1	3
TOTAL	65	3	44	78	0	34	54	9	49

Table A.3: Complete electoral results for the districting in scenario 5 (Section 5.5). Highlighted in gray are the sums of all electoral districts won in each Portuguese region (ID between parentheses), per coalition (or toss-up). The cells highlighted in yellow represent instances where the time limit was exceeded, hence the presented results are for the best solution found after 3 hours (which might not be the global optimum).