A heuristic algorithm for districting problems with similarity constraints

Alex Gliesch, Marcus Ritt
Instituto de Informática
Universidade Federal do Rio Grande do Sul
Porto Alegre, Brazil
{alex.gliesch,marcus.ritt}@inf.ufrgs.br

Arthur H. S. Cruz, Mayron C. O. Moreira
Departamento de Ciência de Computação
Universidade Federal de Lavras
Lavras, Brazil
arthur.cruz@estudante.ufla.br,mayron.moreira@ufla.br

***Abstract*—Redistricting problems arise when due to evolving attributes, current districts get increasingly inefficient or even infeasible. A typical example are electoral districts of roughly the same population, which after changes in its distribution, become imbalanced. Since the cost of creating entirely new districts, in general, is too high, redistricting problems have two conflicting objectives: to find an improved or optimal solution while keeping the changes to existing districts small. In this paper we propose an efficient heuristic for redistricting problems, which can consider similarity to existing districts together with several other constraints, such as balance, connectedness, and compactness. In an experimental study, we evaluate the contribution of heuristic strategies, the effect of different attribute modification models, and the effect of different similarity metrics on the quality of the redefined districts.**

***Index Terms*—districting, redistricting, heuristics**

I. Introduction

Districting problems arise when a set of geographical basic units must be partitioned into districts, subject to additional constraints such as connectivity, compactness, or attribute balancing. A standard model is an undirected graph $G = (V, E)$ where the set of vertices V represents the basic units, and the edge set E neighborhood relations between them. A solution $S = (D_1, \dots, D_p)$ is a p-partition of V into p districts, each inducing a connected subgraph of V. Basic units $v \in V$ may have a set of attributes $a_v^1, \dots, a_v^l$; for each $j \in [l]$ district D has a total attribute value of $A^j(D) = \sum_{v \in D} a_v^j$. The *imbalance* of a district D with respect to attribute j is $B^j(D) = |A^j(D)/\bar{a}^j - 1|$, where $\bar{a}^j = A^j(V)/p$ is the average attribute value of a district. A district is *balanced* if, for each $j \in [l]$ its imbalance with respect to j does not exceed a given tolerance $\tau^j \geq 0$. Common districting problems aim to maximize a compactness measure over districts. The three most common such measures are the maximum diameter $d(D) = \max_{i,j \in D} d_{ij}$ of a district for given distances $d_{ij} \geq 0$, the maximum distance $c(D, c_1) = \max_{i \in D} d_{i,c_1}$ from the district's 1-center $c_1 = c_1(D) = \operatorname{argmin}_{i \in D} c(D, i)$ (in so-called p-center problems), or as the maximum total distance $m(D, m_1) = \sum_{i \in D} d_{i,m_1}$ from the district's 1-median $m_1(D) = \operatorname{argmin}_{i \in D} m(D, i)$, i.e. the basic unit that minimizes the total distance to all other units in the district (in so-called p-median problems). An example of a districting plan is shown in Figure 1. In this paper we measure compactness by

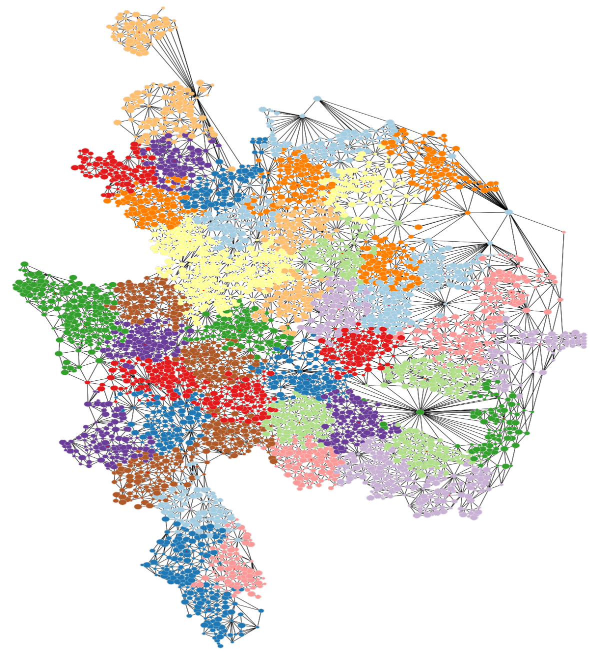

Fig. 1. Example of a districting in the city of Monterrey (instance from [1]).

the maximum total distance from the 1-medians, since this is the most common among the three above and tends to produce (but does not guarantee) connected districts.

In redistricting problems we additionally have an already existing solution $S^0 = (D_1^0, \dots, D_p^0)$, and the solution $S = (D_1, \dots, D_p)$ should be similar to S^0 according to some given similarity metric $d(S, S^0)$. In summary, the problem we address in this paper is:

$$
\begin{aligned}
&\min_{S \in \binom{V}{p}} && \sum_{D \in S} m(D, m_1(D)) && && (1)\\
&\text{subject to} && B^j(D) \leq \tau, && \forall D \in P, j \in [l], && (2)\\
& && \text{Connected}(D), && \forall D \in P, && (3)\\
& && d(S, S^0) \geq \bar{d}, && && (4)\\
& && d(D_j, D^0_{m(j)}) \geq \bar{d} && \forall i \in [p]. && (5)
\end{aligned}
$$

i.e., to maximize the p-median compactness subject to balance, connectivity, and similarity constraints over the set of all p-partitions of basic units $\left\{ {V \atop p} \right\}$, where $\bar{d}$ is an lower bound on the similarity of the new and old solution. Usually only one of constraints (4) and (5) will be active, as explained in Section II.

In this paper we propose a flexible heuristic solver for addressing districting problems with similarity requirements which extends a previous state-of-the-art heuristic for districting. In experiments we show that it is effective in handling both global (constraints (4)) and as a local (constraints (5)) similarity measures, and can treat them as either hard or soft constraint. We further propose two attribute propagation models which modify existing instances to emulate real-world redistricting scenarios, and evaluate them experimentally. In the next section we give an overview on different similarity measures, and then discuss related work in Section III. In Section IV we detail our main algorithm. We present an empirical evaluation in Section V and conclude in Section VI.

II. Similarity measures

A similarity measure quantifies by how much districts have changed over time [2]. As noted by [3], there is comparatively little literature on similarity measures. A measure usually requires a mapping $m : [p] \to [p]$ of old to new districts to evaluate changes in individual districts. Given such a mapping and a measure of similarity $d(D, D^0)$ for pairs of districts, we can then define a global similarity measure as:

$$d(S, S^0) = \sum_{i \in [p]} d(D_j, D^0_{m(j)})/p. \qquad (6)$$

This measure has range $[0, 1]$ with $d(S, S^0) = 0$ for totally dissimilar districtings, and $d(S, S^0) = 1$ for identical ones.

Most similarity measures defined in the literature fit into this model. A natural choice for such a mapping is a bijection. In this case the mapping can be defined by a perfect matching maximizing the sum of pairwise district similarities. If district similarities are symmetric, then the overall value will be, too. A simpler choice for problems where centers are defined (e.g. p-median or p-center problems) can be to use the fixed mapping defined by the district centers of the initial solution. This approach is applied, for example, by [4].

Pairwise distance measures are usually defined in terms of overlaps: for an attribute $j \in [l]$, let the overlap of two districts D, D' be $O(D, D') = A^j(D \cap D')/A^j(V)$. The choice of the attribute depends on the application and can be, for example, area or population.

Mapping m is not necessarily bijective. [5] define similarity of solutions by the sum of the maximum overlaps of every district with some original district. In this case the mapping is defined as

$$m(i) = \operatorname{argmax}_{k \in [p]} O(D_i, D^0_k).$$

[5] use the area of the basic units as main attribute, but as observed above, for some applications other attributes may be more adequate, e.g. the population for political districting. An advantage of this approach is that it also works if S and S' have a different number of districts.

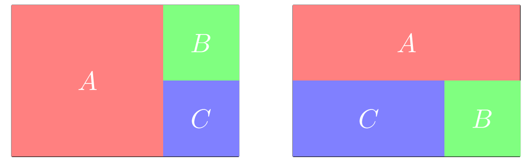

Fig. 2. A simple example with $p = 3$ districts and different similarity measures. Similarities are 0.5, 1/3, and 2/3, for matching-based mapping, center-based mapping, and the mapping of [5], respectively.

Figure 2 illustrates the effect of these different similarity measures. Observe that, when we require a high similarity (e.g. 0.8 or more) then these measures produce the same mapping m, and consequently the same similarity. In this paper, we therefore adopt a center-based fixed mapping m. We further study two constraints, a global similarity constraint

$$d(S, S^0) \geq \bar{d}$$

(used by e.g. [1]) where similarity is defined as the percentage of the attribute values that remain in their original districts, and local similarity constraints

$$d(D_j, D^0_{m(j)}) \geq \bar{d} \qquad \forall i \in [p],$$

where similarity is similarly defined as the percentage of attribute values in the district that remain in their original district.

III. Related work

Districting problems appear in a wide range of applications, such as the design of electoral [6], [7], [8], [5], police [9], [10] or commercial [11], [12], [13] districts, waste collection [14], [15], salt spreading [16], health care systems [17], and land allocation [18], [19]. Since real-world scenarios often translate to large ($\geq$ 1000 basic units) instances solution methods are usually heuristic, mainly through metaheuristics such as tabu search [5], GRASP [20], [12], genetic algorithms [19], [21] or hybrid approaches [22], [23], although several exact algorithms based on mathematical programming have been proposed that can handle smaller instances [6], [7], [24]. [25] provides an extensive overview of the most common districting domains, models and solution methods.

[6] were one of the first to propose computational methods for political redistricting, although they do not consider similarity constraints directly. They present a mathematical model based on a weighted p-median objective function and propose a heuristic based on location-allocation, which iteratively selects a center units for each district and allocates the districts optimally with respect to the fixed centers.

[5] propose a tabu search for a political redistricting problem with a single-objective model that minimizes a convex combination of several criteria, including similarity to existing plans measured for each district by its maximum overlap with some other district in a previous plan.

[26] present a broad overview of exact and heuristic solutions focused on redistricting problems in Germany. The problem considers several criteria as soft requirements, including similarity to existing plans. Because no clear legal guidelines exist the authors do not provide a mathematical measure for similarity, but state that it should be as high as possible, as long as balancing constraints are satisfied. Using adaptations of exact and heuristic approaches, the authors report that the tabu search of [5] outperforms a set of methods from literature.

[4] also consider political redistricting in Germany, and propose three solution methods via constrained geometric clustering. Since again similarity is not clearly defined by law, the authors measure it by the number of pairs of units that used to share a common district, but are assigned to different ones in the new plan. To account for similarity, the methods start optimizing from the original districts. Experimentally the authors report that an anisotropic power diagram approach finds the most similar districts overall.

[1] study a commercial territory design model that handles similarity to previous plans both as a hard constraint, by setting a maximum allowed deviation in number of units between old and new districts, and as an objective, by favoring the assignment of units to their previous districts in the objective function. The authors propose an exact algorithm that iteratively solves an integer model without connectivity constraints, and includes new connectivity constraints as cuts at each iteration until the solution becomes feasible. A similar approach has been proposed by [27], who consider commercial districting models without similarity requirements.

[28] propose a GRASP algorithm for a redistricting problem in power distribution networks. The problem considers diameter-based compactness as an objective and handles similarity by setting upper bounds on the number of basic units that can change districts.

The districting problem in the exact form studied in this paper has not been studied in the literature, to the best of our knowledge; the paper closest to our problem is [1]. For completeness, we present here an explicit mathematical model of our problem.

Let $\tilde{j}$ be the modified attribute in the redistricting process, $\tilde{a}^j$ the current attribute vector ($\tilde{a}^j_v = a^j_v, j \in [l], j \neq \tilde{j}$), and $\overline{a'}^j = (\sum_{v \in V} \tilde{a}^j_v)/p$ the average attribute value of a district. We take d_{uv} as the shortest-path distance between $u, v \in V$. Define $D_i \subseteq V$ as the set of basic units that belong to district i in the former plan. Now, let $x_{uv} \in \{0,1\}$ be a variable equal to one if basic unit v is assigned to a district with center u, and y_{iu} be a binary variable equal to one if center $u \in V$ is allocated to district i. The proposed formulation is presented as follows.

$$\min . \sum_{u,v \in V} d_{uv} x_{uv} \tag{7}$$

$$\text{s.t.} \sum_{u \in V} x_{uu} = p, \tag{8}$$

$$\sum_{u \in V} x_{uv} = 1, \qquad \forall v \in V, \tag{9}$$

$$(1-\tau^j)\overline{a'}^j x_{uu} \leq \sum_{v \in V} \tilde{a}^j_v x_{uv}, \qquad \forall u \in V, j \in [l], \tag{10}$$

$$\sum_{v \in V} \tilde{a}^j_v x_{uv} \leq (1+\tau^j)\overline{a'}^j x_{uu}, \qquad \forall u \in V, j \in [l], \tag{11}$$

$$\sum_{i \in [p]} y_{iu} = x_{uu}, \qquad \forall u \in V, \tag{12}$$

$$\sum_{u \in V} y_{iu} = 1, \qquad \forall i \in [p], \tag{13}$$

$$\sum_{v \in D_i} \tilde{a}^{\tilde{j}}_v x_{uv} \geq \overline{d} y_{iu}, \qquad \forall i \in [p], u \in V, \tag{14}$$

$$x_i(\delta(S)) \geq 1 - |S| + x_i(S), \quad \forall i \in I, S \subseteq V \setminus (\delta(i) \cup \{i\}) \tag{15}$$

$$x_{uv} \in \{0,1\}, \qquad \forall u, v \in V, \tag{16}$$

$$y_{iu} \in \{0,1\}, \qquad \forall i \in [p], \forall u \in V. \tag{17}$$

In the model (7) is the p-median objective function. Constraint (8) establishes the creation of p districts. Constraints (9) indicate that each vertex must be assigned to a single center. Inequalities (10)–(11) are the balance constraints. Coupling constraints (12) and (13) define the variables y. These equations state that a vertex u is candidate to be a center of a district if x_{uu} is one, and there is only one center per district, respectively. Constraints (14) guarantee a minimum local similarity, taking into account the former distribution of basic units over the districts. Constraints (15) guarantee connectivity and have been introduced by [6]. In this constraint $x_i(S) = \sum_{j \in S} x_{ij}$ and $\delta(S)$ is the set of all neighbors of S not in S. Since the number of these constraints is exponential in the size of the instances, this model usually is solved by cut generation. The domain of variables are indicated by Constraints (16) and (17).

IV. A HYBRID HEURISTIC ALGORITHM

In this section we present a hybrid heuristic for districting with similarity constraints. Its overall strategy is similar to an evolutionary GRASP heuristic [29] and adapts the heuristic strategy proposed by [23].

An overview is given in Algorithm 1. We use two objective functions: the total balance excess

$$E(S) = \sum_{i \in [p]} \max\{0, B^j(D_i) - \tau^j \mid j \in [l]\} \cup \{0\}$$

of solution $S = (D_1, \ldots, D_p)$ and its compactness $C(S)$, and also two lexicographic versions of the objective functions $EC(S) = (E(S), C(S))$ and $CE(S) = (C(S), E(S))$, which order solutions by balance excess followed by compactness, and vice-versa. A solution with balance excess $E(S) = 0$ is called *balanced*.

The heuristic maintains a pool of good solutions, to which it adds at the start of each iteration p new solutions created by a constructive algorithm, but keeps only up to $2p$ of the best solutions. Each iteration extracts the best solution from the pool and applies an improvement procedure. The improvement procedure alternates between improving the balance and the

Algorithm 1 A hybrid heuristic for re-districting.

$P \leftarrow \emptyset$
repeat
  for $i \in [p]$ **do**
    $P \leftarrow P \cup \{\text{construct}(S_0)\}$
  remove $\max\{0, |P| - 2p\}$ worst solutions from P
  $S \leftarrow P_0, P \leftarrow P \setminus \{P_0\}$ for best solution P_0 in P
  $S \leftarrow$ optimizeBalance(S', S)
  if $E(S) > 0$ **then**
    return S
  for $i \in [A_{\text{max}}]$ **do**
    $S' \leftarrow$ optimizeCompactness(S)
    $S \leftarrow$ optimizeBalance(S', S)
    if $E(S) > 0$ **then**
      break
until time or iteration limit reached
return S^*

compactness of the solution, starting with the balance, if the solution is not balanced initially. When improving compactness, it accepts solutions with an imbalance of up to $\alpha\,\%$. If a solution is not balanced after improving compactness, it will be balanced again in the subsequent balance step; if that fails the current iteration terminates. We also terminate an iteration early, if the solution after optimizing balance has been visited already, to avoid cycling. The alternating improvement steps are repeated at most A_{max} times. During optimization we always maintain all districts connected. The algorithm terminates after a maximum number of iterations or when a time limit is reached.

To handle similarity we have implemented two different strategies, a hard constraint and a soft constraint, that will be compared experimentally. When using hard constraints, all steps guarantee that the similarity is always kept above the minimum similarity of $\overline{d}$. This is achieved by modifying the constructive and neighborhood search heuristics, as explained below. When using soft constraints, we treat similarity as another set of balancing constraints, and use an extended balance excess function

$$E'(S) = E(S) + d(S, S^0).$$

We next describe the algorithmic components in more detail. In the procedures we allow partial solutions, where some of the vertices are not yet assigned. For a solution S, partial or complete, we denote the index of the district of a vertex $v \in V$ by $\iota(v)$, and the *boundary* of district D by $\partial D = \{v \in V \mid v \notin D \wedge \exists u \in N(v) : D_{\iota(u)} = D\}$. The neighborhood searches will use two types of moves, a *shift* $u \to k$ which assigns vertex u to district k, and a *swap* $u \leftrightarrow v$ which applies shifts $u \to \iota(v)$ and $v \to \iota(u)$.

A. Initial solutions

Initial solutions are created by a constructive greedy algorithm, following known strategies from the literature [30], [31]. For constructing a complete solution from scratch, we first initialize the p districts with p seed vertices. The seed vertices are selected to maximize the minimum distance between two seeds, and are chosen greedily, using the technique of [23]. During the construction we maintain for each district D_i a candidate vertex κ_i, which is the vertex from ∂D_i that increases $C(S)$ least. If there are multiple candidates, one is chosen randomly. We then repeatedly assign to the district D of minimum total weight $\sum_{j\in[l]} A^j(D)$ its best candidate vertex, until a complete solution has been constructed.

When redistricting the constructive heuristic also considers the similarity measure, and generates districts with a similarity above the lower bound of $\overline{d}$. To this end, a random vertex from each district is chosen as the seed, and the candidates of each district are restricted to those vertices in ∂D_i that belong to the previous district, until the district has reached a similarity of at least $\overline{d}$. After feasibility is guaranteed, the construction proceeds as described above. This procedure guarantees solutions that are feasible with respect to similarity, and variation of the districts by ignoring the previous districts after reaching the minimum similarity.

B. Improving compactness

We use a tabu search on the shift neighborhood to improve the compactness of the solution. In each iteration the best non-tabu move is selected. On applying a move, the shifted vertex is declared tabu for τ iterations, i.e. cannot participate in the next τ moves, where τ is the tabu tenure. The search stops on stagnation, defined as I_{max} iterations without improving the incumbent.

C. Improving balance

We use a tabu search on the shift and swap neighborhoods embedded into a binary search for improving balance. The tabu search selects in each iteration the best non-tabu move, and applies it. Swap moves are only considered if no improving shift move could be found. On applying a move, the participating vertices (i.e. vertex u for a shift move $u \to k$, and vertices u and v for a swap move $u \leftrightarrow v$) are declared tabu for τ iterations, as for the compactness search.

Balancing a solution may increase its compactness measure. To avoid increasing compactness too much, each tabu search runs with a maximum compactness C_m, and considers only moves that do not exceed C_m. Since the optimal value of C_m that still can be balanced is not known, the tabu search is embedded into a binary search for the smallest compactness value $C_m \in [C_l, C_u]$ that can be balanced. The upper bound C_u is set to the compactness value of the incumbent (i.e. the currently most compact, balanced solution), or to two times the current compactness, when no balanced solution has been found yet; the lower bound C_l is the compactness of the solution to be balanced.

D. Data structures for maintaining similarity

During the computation we need to maintain either the local similarity for each district, or the global similarity, under shift and swap operations. This can be done efficiently by updating

TABLE I
CHARACTERISTICS OF THE INSTANCES USED IN THE COMPUTATIONAL EXPERIMENTS.

Set	N	n	p	Attr.	Ref.
2DU60	20	60	6	Uniform	[27]
2DU80	20	80	8	Uniform	[27]
2DU100	20	100	10	Uniform	[27]
2DU120	20	120	12	Uniform	[27]
DU150	10	150	15	Uniform	[27]
DU200	10	200	20	Uniform	[27]
d500	20	500	10	Uniform	[12]
DT500	20	500	10	Symmetric	[12]
DT1000	15	1000	40	Symmetric	[32]
DU1000	30	1000	10	Symmetric	[32]

the corresponding values when executing moves. Let $\omega(v)$ be the original district of vertex v. Then, for a shift $v \to k$, if $\iota(v) = \omega(v)$ the current total attribute value (and consequently the similarity) of district $\iota(v)$ decreases, since vertex v is removed from its original district. The same holds for the global similarity. If, on the other hand, $\omega(v) = k$, then total global and local attribute values for district k increase. These values can be similarly updated for swaps $u \leftrightarrow v$. Overall we can maintain the total attribute values with cost $O(1)$ for updates, and query the total similarity in time $O(p)$ by summing over all districts.

V. Computational experiments

In this section we report on computational experiments. We present the experimental methodology in Section V-A, introduce attribute modification models in Section V-B, evaluate in Section V-C the effectiveness of the soft versus hard constraints for maintaining similarity. In Section V-E we assess the effect of different similarity metrics.

A. Experimental methodology

We use the ten instance sets shown in Table I in our experiments. For each set the table shows the total number of instances N, the number of vertices n, the number of districts p, the attribute generation method, and gives a reference to the paper that introduced the set. These sets have been selected to cover a wide range of instances of different size, number of districts, and attribute ranges. All instances have 3 attributes, which were generated with a uniform distribution over a fixed interval, or with a non-uniform, but symmetric distribution [12].

The heuristics have been implemented in C++ and compiled with GCC 7.3.0 at maximum optimization. All tests have been run on a PC with a Core i7 930 CPU running at 2.8 GHz and 12 GB of main memory, and running Ubuntu Linux 18.04. Each test has been run on a single core. Following [23] we have fixed parameters $A_{\max}$ and $I_{\max}$ at 100, and set the tabu tenure to $1.5p$. Our algorithm runs with a fixed random seed and is deterministic. The balancing tolerances τ^j were set to 0.05 for all $j \in [l]$, which is the most common value in the literature. If not specified otherwise, we have run the heuristics with a time limit of 600 seconds and 1000 iterations.

B. Attribute modification models

The main reason for applying redistricting is when the existing attribute values have changed and thus lead to inefficient districts (e.g. when the total sum of the distances of voters to their voting center gets too large) and infeasible solutions (e.g. when the current districts are violating balancing constraints).

We model changes in attribute values over time by a discrete diffusion model. Instead of the input graph $G = (V, E)$ we work here with its directed version $G^d = (V, A)$, with $A = \{(u, v) \mid u, v \in V, \{u, v\} \in E\}$, where each edge $\{u, v\} \in E$ has been expanded into two arcs (u, v) and (v, u). A diffusion model is defined by a doubly stochastic matrix $P^j = (p^j_{uv})_{u\in V, j\in V}$ for each attribute value $j \in [l]$, i.e. $\sum_{u\in V} p^j_{uv} = 1$ for all $v \in V$ and $\sum_{v\in V} p^j_{uv} = 1$ for all $u \in V$. Here p^j_{uv} is the probability that an attribute value at $u \in V$ diffuses to vertex $v \in V$. In particular p^j_{uu} is the probability of an attribute value remaining at its current vertex. Diffusion is limited to neighbors, i.e. $p^j_{uv} = 0$ for all $(u, v) \notin A$. In one step of the diffusion, values of attribute $j \in [l]$ change to $P^j a^j$ where $a^j = (a^j_v)_{v\in V}$ is the vector of the jth attribute values.

We study two diffusion models here, a uniform model and a migration model. In both we define a basic probability p_r for an attribute value to remain at the current vertex $v \in V$, and set $p^j_{vv} = p_r$ for all $j \in [l]$. In our experiments we have used $p_r = 0.95$ which models a high probability of remaining at the current vertex. We then distribute the remaining probability $1 - p_r$ over the neighbors of a vertex. In the uniform model, we define $p^j_{uv} = (1 - p_r)/|N(u)|$ for all $v \in N^+(u)$, where $N^+(u)$ are the (outgoing) neighbors of vertex u in graph G^d. This represents an unspecific diffusion to the neighbors.

In the migration model, we model the concentration of attributes in centers. This usually applies to the population. To this end we use the gravity model of migration that assumes that interaction between places u, v is proportional to $I(u, v) = {a^j_u}^{\alpha_u} {a^j_v}^{\alpha_v} / d^\gamma_{uv}$ for some attribute $j \in [l]$ [33]. Following [34] we select values $\alpha_v \in U[0.8, 0.9]$ for all vertices $v \in V$, and $\gamma \in U[0.8, 0.9]$. As before, we set $p^j_{vv} = p_r$ for all $v \in V$, and distribute the remaining probability $1 - p_r$. For each neighboring vertex $u \in N(a)$ we define a utility value of $U_v(u) = I(u, v)$. Then the probability of a attribute value a^j_v migrating to neighbor u is $p^j_{vu} = U_v(u)/\sum_{u\in N(v)} U_v(u)$.

C. Comparison of hard and soft constraints for similarity

In our first experiment we analyze whether it is better to handle the similarities as a hard or a soft constraint. When treated as a hard constraint, the similarity is always maintained above the minimum $\overline{d}$. Consequently, initial solutions that do not satisfy the minimum similarity, and moves in local searches that violate it are discarded. On the other hand, when

TABLE II
RESULTS FOR THE UNIFORM MODEL AFTER 5 AND 10 STEPS WHEN SIMILARITY IS A HARD CONSTRAINT.

	5 steps				10 steps			
Inst.	b_0 [%]	b [%]	m_1	d [%]	b_0 [%]	b [%]	m_1	d [%]
2DU60	100.0	100.0	0.0	100.0	100.0	100.0	0.0	100.0
2DU80	100.0	100.0	0.0	100.0	100.0	100.0	0.0	100.0
2DU100	90.0	100.0	1.5	100.0	85.0	100.0	2.0	100.0
2DU120	100.0	100.0	−0.8	100.0	80.0	100.0	1.5	100.0
DU150	70.0	100.0	2.1	100.0	70.0	100.0	2.3	100.0
DU200	100.0	100.0	−1.1	100.0	80.0	100.0	0.7	100.0
d500	40.0	100.0	−19.9	100.0	30.0	100.0	−17.2	100.0
DT500	90.0	100.0	−1.7	100.0	85.0	100.0	−1.7	100.0
DT1000	33.3	100.0	−21.3	100.0	0.0	100.0	−21.4	100.0
DU1000	76.7	100.0	−1.0	100.0	70.0	100.0	−1.2	100.0
Avg./Tot.	80.0	100.0	−4.2	100.0	70.0	100.0	−3.5	100.0

TABLE III
RESULTS FOR THE UNIFORM MODEL AFTER 5 AND 10 STEPS WHEN SIMILARITY IS A SOFT CONSTRAINT.

	5 steps				10 steps			
Inst.	b_0 [%]	b [%]	m_1	d [%]	b_0 [%]	b [%]	m_1	d [%]
2DU60	100.0	100.0	0.0	100.0	100.0	100.0	0.0	100.0
2DU80	100.0	100.0	0.0	100.0	100.0	100.0	0.0	100.0
2DU100	90.0	100.0	1.4	100.0	85.0	100.0	6.4	100.0
2DU120	100.0	100.0	−0.8	100.0	80.0	100.0	1.0	100.0
DU150	70.0	100.0	2.1	100.0	70.0	100.0	2.3	100.0
DU200	100.0	100.0	−1.3	100.0	80.0	100.0	3.6	100.0
d500	40.0	100.0	−18.4	100.0	30.0	100.0	−17.1	100.0
DT500	90.0	100.0	−1.6	100.0	85.0	100.0	−2.0	100.0
DT1000	33.3	100.0	−21.4	100.0	0.0	100.0	−22.4	100.0
DU1000	76.7	100.0	−1.2	100.0	70.0	100.0	−1.1	100.0
Avg./Tot.	80.0	100.0	−4.1	100.0	70.0	100.0	−2.9	100.0

treated as a soft constraint, similarity is handled as a balance constraint: we allow arbitrary low values, in particular after improving the compactness, but insist in subsequent balancing steps on a feasible solution. If a feasible solution is not found we proceed with the next iteration.

For the comparison we first create an initial districting plan by running the full heuristic. We next run the diffusion model with a uniform diffusion model for 5 and 10 steps, and then apply the redistricting to the new instances, with the districts obtained in the first steps, and a local similarity of at least 0.8. The results when using the hard constraint can be seen in Table II, and for soft constraints in Table III. Both tables show the percentage of balanced instances at the beginning b_0 and the end b, the relative deviation m_1 from the initial compactness, and the percentage d of instances that satisfy the similarity constraint.

We can see that with an increasing number of diffusion steps, the percentage of initially balanced instances decreases. Similarly the compactness decreases (i.e. the p-median increases). This is expected, since the attribute distributions get increasingly different, leading to unbalanced instances. In all

TABLE IV
RESULTS FOR THE GRAVITY MODEL AFTER 5 AND 10 STEPS WHEN SIMILARITY IS A HARD CONSTRAINT.

	5 steps				10 steps			
Inst.	b_0 [%]	b [%]	m_1	d [%]	b_0 [%]	b [%]	m_1	d [%]
2DU60	100.0	100.0	0.0	100.0	60.0	100.0	95.4	100.0
2DU80	90.0	100.0	8.9	100.0	25.0	90.0	47.0	100.0
2DU100	25.0	100.0	41.7	100.0	5.0	100.0	65.4	100.0
2DU120	35.0	100.0	40.2	100.0	0.0	75.0	249.4	100.0
DU150	50.0	100.0	24.1	100.0	0.0	100.0	190.9	100.0
DU200	30.0	100.0	21.5	100.0	0.0	80.0	144.1	100.0
d500	10.0	100.0	−4.1	100.0	0.0	100.0	37.2	100.0
DT500	10.0	100.0	3.3	100.0	0.0	100.0	11.7	100.0
DT1000	0.0	100.0	−3.9	100.0	0.0	20.0	16.8	100.0
DU1000	10.0	100.0	1.2	100.0	0.0	100.0	7.8	100.0
Avg./Tot.	36.0	100.0	13.3	100.0	9.0	86.5	86.6	100.0

cases the redistricting method found a new feasible, balanced solution. The compactness of the instances after redistricting is sometimes smaller, and the improvement increases with the instance size. This can be explained by the additional computation time invested. Comparing the compactness values in Tables II and III we can see that the hard constraints overall work slightly better, and we use this strategy in the remaining experiments.

D. Effect of different attribute modification models

In this experiment we look at the gravity diffusion model. As above we run the model for 5 and 10 steps on the initial districts from the first experiment, and then the redistricting heuristic with similarity as a hard constraint.

The results are summarized in Table IV. We can see that the percentage of initially infeasible instances with 36% and 9% after 5 and 10 steps is much lower. The reason is that the gravity model is a global model, where all basic units interact, and thus the change in attributes is higher. Furthermore, the gravity model leads to a concentration of attribute values. As a consequence these instances are more realistic but also harder to solve. After 5 steps the heuristic still is able to find a feasible solution for all instances, after 10 steps 85% of the instances can still be balanced. All found solutions satisfy the minimum similarity of 80%. Looking at the relative deviations from the p-median value of the initial district, which are considerably higher, we can see that these instances are harder to solve; the heuristic must invest more time in balancing the solutions. An exception are instance sets d500 and DT1000 after 5 steps, which improve the p-median slightly.

In Table V we show performance data for the gravity model: for 5 and 10 steps we present the average number of iterations (column “it.”), the average time find the best solution (column “t_b”, in seconds), and the total runtime (column “t”, in seconds). We can see that for the smaller instances with up to 200 vertices, the limit of 1000 iterations is reached in less than two minutes, and the best solution is found quickly, within at most 20 seconds. The larger instances also achieve close to

TABLE V
NUMBER OF ITERATIONS, TIME TO BEST AND TOTAL RUNTIME FOR THE GRAVITY MODEL AFTER 5 AND 10 STEPS WHEN SIMILARITY IS A HARD CONSTRAINT.

	5 steps			10 steps		
Inst.	it.	t_b[s]	t[s]	it.	t_b[s]	t[s]
2DU60	1000.0	0.0	3.9	1000.0	0.1	7.3
2DU80	1000.0	0.0	14.9	1000.0	0.1	20.0
2DU100	1000.0	0.2	17.2	1000.0	0.5	24.4
2DU120	1000.0	0.1	34.7	1000.0	1.6	51.6
DU150	1000.0	0.2	38.6	1000.0	2.0	64.4
DU200	1000.0	2.0	62.8	1000.0	19.6	103.0
d500	968.6	147.6	546.2	974.1	198.7	512.7
DT500	1000.0	85.3	436.6	998.8	115.1	468.9
DT1000	1000.0	216.2	429.2	1000.0	199.4	414.1
DU1000	259.8	158.3	600.0	268.5	240.0	600.0
Avg./Tot.	922.8	61.0	218.4	924.1	77.7	226.6

TABLE VI
LOCAL AND GLOBAL SIMILARITY IN THE GRAVITY MODEL AFTER 5 AND 10 STEPS.

	5 steps		10 steps	
Inst.	s_l [%]	s_g [%]	s_l [%]	s_g [%]
2DU60	100.0	100.0	96.3	97.3
2DU80	98.4	99.0	91.5	96.0
2DU100	94.0	97.7	90.9	96.1
2DU120	95.1	98.6	89.2	94.2
DU150	95.8	98.2	89.1	94.9
DU200	94.1	98.3	86.1	93.5
d500	90.0	95.1	88.3	94.0
DT500	97.2	99.0	95.2	98.4
DT1000	87.2	95.5	92.8	98.2
DU1000	97.0	98.9	95.1	98.3
Avg./Tot.	94.9	98.0	91.5	96.1

1000 iterations within 10 minutes, except for DU1000 which always hits the time limit after less than 300 iterations. The performance of the heuristics is mainly driven by instances size, and less dependent on the number of steps. We do not show results for the uniform model, which are very similar.

E. Effect of different similarity metrics

In this section we look at the difference between local and global similarity measures after running 5 and 10 steps of the gravity model. In Table VI we show the smallest local similarity s_l over all p districts and the global similarity s_g. We find that, as expected, the global similarity is always higher than the local similarity, and the similarities decrease with an increasing number of steps. Similarities are always well above the lower limit of 80%, showing that a high similarity can be maintained, and the difficulties in finding feasible solutions lie in the balancing step. As a consequence, we believe that imposing a local, per-district similarity lower bound is preferable, since global similarity alone may lead to a solution where a few districts suffer a large change.

VI. CONCLUSIONS

We have proposed a heuristic algorithm for districting problems with similarity constraints. Its main algorithmic components are a constructive algorithm that generates initially feasible solutions for a given minimum similarity, and a hybrid search strategy that alternates between improving the compactness objective and satisfying balance constraints via tabu search, maintaining feasible solutions throughout the search and updating current similarity values efficiently.

For the experimental analysis we have introduced two diffusion models for generating instances with modified attribute values. In the experiments we found that maintaining feasible solutions with respect to similarity tends to be more effective than allowing solutions that go below the minimum similarity. Global and local similarity values of solutions produced are well above the lower limit of 80%, and imposing local similarity constraints seems preferable. The heuristic finds in most cases new feasible solutions of the required similarity, and is sometimes able to improve the compactness of the solution. Overall we believe that our approach demonstrates that efficient heuristic redistricting is possible with a few selected modifications of constructive and local search algorithms.

ACKNOWLEDGEMENTS

Our research has been supported by the funding agencies CNPq (grant 420348/2016-6), FAPEMIG (grant TEC-APQ-02694-16), by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, Brasil (CAPES), Finance Code 001, and by Google Research Latin America (grant 25111). We would also like to thank the support of the Fundação de Desenvolvimento Científico e Cultural (FUNDECC/UFLA).

REFERENCES

[1] R. Z. Ríos-Mercado and J. F. López-Pérez, "Commercial territory design planning with realignment and disjoint assignment requirements," *Omega*, vol. 41, pp. 525–535, 2013.

[2] G. R. Webster, "Reflections on current criteria to evaluate redistricting plans," *Political Geography*, vol. 32, pp. 3–14, Jan. 2013.

[3] J. C. Williams, Jr., "Political districting: A review," *Regional Science*, vol. 74, no. 1, pp. 13–40, 1995.

[4] A. Brieden, P. Gritzmann, and F. Klemm, "Constrained clustering via diagrams: A unified theory and its application to electoral district design," *European Journal of Operational Research*, vol. 263, no. 1, pp. 18–34, 2017.

[5] B. Bozkaya, E. Erkut, and G. Laporte, "A tabu search heuristic and adaptive memory procedure for political districting," *European Journal of Operational Research*, vol. 144, no. 1, pp. 12–26, 2003.

[6] S. W. Hess, J. B. Weaver, H. J. Siegfeldt, J. N. Whelan, and P. A. Zitlau, "Nonpartisan political redistricting by computer," *Operations Research*, vol. 13, no. 6, pp. 998–1006, 1965.

[7] A. Mehrotra, E. L. Johnson, and G. L. Nemhauser, "An Optimization Based Heuristic for Political Districting," *Management Science*, vol. 44, no. 8, pp. 1100–1114, 1998.

[8] F. Ricca, A. Scozzari, and B. Simeone, "Political districting: from classical models to recent approaches," *Annals of Operations Research*, vol. 204, no. 1, pp. 271–299, 2013.

[9] S. J. D'Amico, S.-J. Wang, R. Batta, and C. M. Rump, "A simulated annealing approach to police district design," *Computers & Operations Research*, vol. 29, no. 6, pp. 667–684, may 2002.

[10] M. Camacho-Collados, F. Liberatore, and J. M. Angulo, "A multi-criteria police districting problem for the efficient and effective design of patrol sector," *European Journal of Operational Research*, vol. 246, no. 2, pp. 674–684, 2015.

[11] B. Fleischmann and J. N. Paraschis, “Solving a large scale districting problem: A case report,” *Computers & Operations Research*, vol. 15, no. 6, pp. 521–533, 1988.
[12] R. Z. Ríos-Mercado and E. Fernández, “A reactive GRASP for a commercial territory design problem with multiple balancing requirements,” *Comput. Oper. Res.*, vol. 36, pp. 755–776, 2009.
[13] H. Lei, G. Laporte, Y. Liu, and T. Zhang, “Dynamic design of sales territories,” *Computers and Operations Research*, vol. 56, pp. 84–92, 2015.
[14] S. Hanafi, A. Freville, and P. Vaca, “Municipal solid waste collection: An effective data structure for solving the sectorization problem with local search methods,” *INFOR: Information Systems and Operational Research*, vol. 37, no. 3, pp. 236–254, 1999.
[15] G. Ghiani, D. Lagana, E. Manni, R. Musmanno, and D. Vigo, “Operations research in solid waste management: A survey of strategic and tactical issues,” *Computers & Operations Research*, vol. 44, pp. 22–32, 2014.
[16] L. Muyldermans, D. Cattrysse, D. van Oudheusden, and T. Lotan, “Districting for salt spreading operations,” *Eur. J. Oper. Res.*, vol. 139, pp. 521–532, 2002.
[17] M. T. A. Steiner, D. Datta, P. J. Steiner Neto, C. T. Scarpin, and J. Rui Figueira, “Multi-objective optimization in partitioning the healthcare system of Parana state in Brazil,” *Omega (United Kingdom)*, vol. 52, pp. 53–64, 2015.
[18] S. Borgwardt, A. Brieden, and P. Gritzmann, “Geometric clustering for the consolidation of farmland and woodland,” *The Mathematical Intelligencer*, vol. 36, no. 2, pp. 37–44, 2014.
[19] A. Gliesch, M. Ritt, and M. C. O. Moreira, “A genetic algorithm for fair land allocation,” in *Proc. 19th Conf. Genetic Evol. Comput.*, P. A. Bosnan, Ed. Berlin: ACM Press, 2017, pp. 793–800.
[20] R. Z. Ríos-Mercado and H. J. Escalante, “GRASP with path relinking for commercial districting,” *Expert Systems with Applications*, vol. 44, pp. 102–113, 2016.
[21] F. Bação, V. Lobo, and M. Painho, “Applying genetic algorithms to zone design,” *Soft Computing*, vol. 9, no. 5, pp. 341–348, 2005.
[22] A. Butsch, J. Kalcsics, and G. Laporte, “Districting for arc routing,” *INFORMS Journal on Computing*, vol. 26, no. 4, pp. 809–824, 2014.
[23] A. Gliesch and M. Ritt, “A generic approach to districting with diameter or center-based objectives,” in *Proc. 21th Conf. Genetic Evol. Comput.*, M. López-Ibáñez, Ed. Prague: ACM Press, 2019, pp. 249–257.
[24] M. A. Salazar-Aguilar, R. Z. Ríos-Mercado, and M. Cabrera-Ríos, “New Models for Commercial Territory Design,” *Networks and Spatial Economics*, vol. 11, no. 3, pp. 487–507, 2011.
[25] J. Kalcsics and R. Z. Ríos-Mercado, “Districting Problems,” in *Location Science*. Cham: Springer International Publishing, 2019, pp. 705–743. [Online]. Available: http://link.springer.com/10.1007/978-3-030-32177-2_25
[26] S. Goderbauer and J. Winandy, “Political districting problem: Literature review and discussion with regard to federal elections in germany,” RWTH Aachen, Tech. Rep., 2018.
[27] M. A. Salazar-Aguilar, R. Z. Ríos-Mercado, and M. Cabrera-Ríos, “New models for commercial territory design,” *Netw. Spat. Econ.*, vol. 11, no. 3, pp. 487–507, 2011.
[28] L. S. de Assis, P. M. Franca, and F. L. Usberti, “A redistricting problem applied to meter reading in power distribution networks,” *Comput. Oper. Res.*, vol. 41, pp. 65–75, 2014.
[29] M. G. Resende and C. C. Ribeiro, *Optimization by GRASP*. Springer, 2016.
[30] E. Erkut, Y. Ülküsal, and O. Yeniçerioğlu, “A comparison of p-dispersion heuristics,” *Comput. Oper. Res.*, vol. 21, no. 10, pp. 1103–1113, 1994.
[31] A. Gliesch, M. Ritt, and M. C. O. Moreira, “A multistart alternating tabu search for commercial districting,” in *The 18th European Conference on Evolutionary Computation in Combinatorial Optimisation*, ser. LNCS, A. Liefooghe and M. López-Ibáñez, Eds., no. 10782. Parma, Italy: Springer, 2018, pp. 158–173.
[32] R. Z. Ríos-Mercado and J. C. Salazar-Acosta, “A GRASP with strategic oscillation for a commercial territory design problem with a routing budget constraint,” in *Mexican International Conference on Artificial Intelligence*, 2011, pp. 307–318.
[33] Q. J. Stewart, “The development of social physics,” *Amer. J. Phys.*, vol. 18, pp. 239–253, 1950.
[34] J. Poot, O. Alimi, M. P. Cameron, and D. C. Maré, “The gravity model of migration: The successful comeback of an ageing superstar in regional science,” Forschungsinstitut zur Zukunft der Arbeit / Institute for the Study of Labor, Tech. Rep., 2016.