Optimizing the Layout of Proportional Symbol Maps

Guilherme Kunigami[1,⋆], Pedro J. de Rezende[1,⋆⋆], Cid C. de Souza[1,⋆⋆⋆], and Tallys Yunes[2]

[1] Institute of Computing, State University of Campinas, Campinas, SP, Brazil 13084-852
kunigami@gmail.com, {rezende,cid}@ic.unicamp.br
[2] Department of Management Science, University of Miami Coral Gables, FL, USA 33124-8237
tallys@miami.edu

Abstract. Proportional symbol maps are a cartographic tool to assist in the visualization and analysis of quantitative data associated with specific locations (earthquake magnitudes, oil well production, temperature at weather stations, etc.). Symbol sizes are proportional to the magnitude of the quantities that they represent. We present a novel integer programming model to draw opaque disks on a map with the objective of maximizing the total visible border of all disks (an established measure of quality). We focus on drawings obtained by layering symbols on top of each other, known as *stacking drawings*. We introduce decomposition techniques, and several new families of facet-defining inequalities, which are implemented in a cut-and-branch algorithm. We assess the effectiveness of our approach through a series of computational experiments using real demographic and geophysical data. To the best of our knowledge, we are the first to provide provably optimal solutions to some of those problem instances.

Keywords: Computational Geometry, Symbol Maps, Integer Linear Programming, Cartography.

1 Introduction

Proportional symbol maps (PSMs) are a cartographic tool to assist in the visualization and analysis of quantitative data associated with specific locations (e.g. earthquake magnitudes, oil well production, temperature at weather stations, etc.). At each location, a symbol is drawn whose size is proportional to the numerical data collected at that point on the map (see [1,2]). For our purposes, the

⋆ Supported by CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico – Grant #830510/1999-0.
⋆⋆ Partially supported by CNPq Grants #472504/2007-0, 483177/2009-1, 473867/2010-9, FAPESP – Fundação de Amparo à Pesquisa do Estado de São Paulo – Grant #07/52015-0 and a Grant from FAEPEX/Unicamp.
⋆⋆⋆ Partially supported by CNPq Grants #301732/2007-8, 472504/2007-0, 473867/2010-9 and FAPESP Grant #07/52015-0.

B. Murgante et al. (Eds.): ICCSA 2011, Part III, LNCS 6784, pp. 1–16, 2011.

symbols are scaled opaque disks (typically preferred by users [7]), and we focus on drawings obtained by layering symbols on top of each other, also known as *stacking drawings*. Because of overlapping, a drawing of the disks on a plane will expose some of them (either completely or partially) and potentially obscure the others. Although there have been studies about symbol sizing, it is unclear how much the symbols on a PSM should overlap (see [5,12]). The quality of a drawing is related to how easily the user is able to correctly judge the relative sizes of the disks. Intuitively, the accuracy of such a judgment is proportional to how much of the disk borders are visible. As a consequence, the objective function consists of maximizing one of two alternative measures of quality: the minimum visible border length of any disk (the *Max-Min* problem) – which emphasizes the local perception, or the total visible border length over all disks (the *Max-Total* problem) – which benefits the global awareness. For n disks, Cabello et al. [1] show that the Max-Min problem can be solved in $O(n^2 \log n)$ in general, or in $O(n \log n)$ if no point on the plane is covered by more than $O(1)$ disks. The complexity of the Max-Total problem for stacking drawings is open.

The contributions of this work are: (i) proposing a novel integer linear programming (ILP) formulation for the Max-Total problem; (ii) introducing decomposition techniques, as well as several new families of facet-defining inequalities; and (iii) implementing a cut-and-branch algorithm to assess the effectiveness of our approach through a series of computational experiments on a set of instances that includes real geophysical data from NOAA's National Geophysical Data Center [11]. To the best of our knowledge, we are the first to provide provably optimal solutions to some of the Max-Total instances studied in [1,2]. We are unaware of other attempts at using ILP to solve this problem.

In Section 2, we describe the problem more formally and introduce some basic terminology. We present the ILP model in Section 3, and perform a polyhedral study of the formulation in Section 4. We describe new families of facet-defining inequalities in Section 5, and introduce decomposition techniques in Section 6. The computational results obtained with our cut-and-branch algorithm appear in Section 7.

2 Problem Description and Terminology

Let $S = \{1, 2, \ldots, n\}$ be a set of disks with known radii and center coordinates on the Euclidean plane. Let the *arrangement* $\mathcal{A}$ be defined as the picture formed by the borders of all the disks in S. A point at which two or more disk borders intersect is called a *vertex* of $\mathcal{A}$. A portion of a disk border that connects two vertices, with no other vertices in between, is called an *arc*. An area of $\mathcal{A}$ that is delimited by arcs is called a *face*. A *drawing* of S is a subset of the arcs and vertices of $\mathcal{A}$ that is drawn on top of the filled interiors of the disks in S (see Figure 1).

A *canonical face* is a face that contains no arcs in its interior. A set of arcs on the boundary of a canonical face that belong to the same disk constitutes a *canonical arc*. In Figure 2, the boundary of face f is made up of canonical arcs

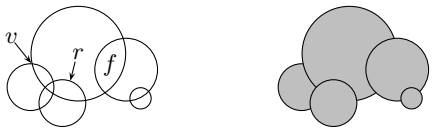

Fig. 1. Arrangement with vertex v, arc r, and face f (left), and a drawing (right)

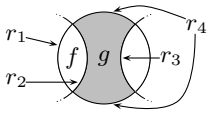

Fig. 2. Three single-piece canonical arcs r_1, r_2, r_3; a multi-piece canonical arc r_4

r_1 and r_2. The boundary of face g is made up of three canonical arcs: r_2, r_3 and r_4. Note that canonical arc r_4 is composed of two pieces. From now on, arcs and faces are assumed to be canonical, unless noted otherwise.

Given an arrangement, many drawings are possible, but not all of them represent a sensible, physically feasible, placement of symbols. A *stacking drawing* is obtained by assigning disks to levels (a stacking order) and drawing them, in sequence, from the lowest to the highest level.

3 An Integer Linear Programming Model

Let $G_S = (V, E)$ be an undirected graph with one vertex for every disk $i \in S$ (denoted $V(i)$) and one edge for every pair of vertices whose corresponding disks overlap. Moreover, let $m-1$ be the length of the longest simple path in G_S, and let $\mathcal{K}$ be the set of all maximal cliques of G_S.

Proposition 1. *The Max-Total problem for stacking drawings has an optimal solution that uses at most m levels.*

Proof. Assume that a given solution assigns levels to all disks using more than m levels. Create a directed graph G'_S such that $V(G'_S) = V(G_S)$ and arc (i, j) is directed from i to j if disk i is at a level below disk j. Because the given solution is a stacking drawing, G'_S contains no directed cycles and hence admits a topological ordering of its vertices. Note that this ordering induces the same stacking order as the given solution. Because the length of the longest directed path in G'_S is at most $m-1$, the greatest label used in the topological ordering is less than or equal to m. □

Even though it may seem, at first glance, that an optimal solution might require at most as many levels as the size of the largest clique in G_S, it is easy to see that in the case where G_S is a simple path with $n > 2$ vertices, its largest clique has size 2, while an optimal solution may require n levels.

Our ILP model uses the following data, which can be calculated in polynomial time from the set S:

- $R \equiv$ set of all arcs;
- $\ell_r \equiv$ length of arc $r \in R$ (total length if r has multiple pieces);
- $d_r \equiv$ disk that contains arc r in its border;
- $S_r^I \equiv$ set of disks that contain arc r in their interior.

For each $r \in R$, let the binary variable x_r be equal to 1 if arc r is visible in the drawing, and equal to 0 otherwise. Then, the objective is to maximize $\sum_{r \in R} \ell_r x_r$. We assume that $m \geq 2$ because it is trivial to find the optimal solution when $m = 1$. For each disk $i \in S$, let the binary variable y_{ip} be equal to 1 if disk i is at level p ($1 \leq p \leq m$), and equal to 0 otherwise. A stacking drawing has to satisfy the following constraints:

$$\sum_{p=1}^{m} y_{ip} \leq 1, \qquad \forall\, i \in S, \tag{1}$$

$$x_r - \sum_{p=1}^{m} y_{d_r p} \leq 0, \qquad \forall\, r \in R, \tag{2}$$

$$\sum_{i\,:\,V(i) \in K} y_{ip} \leq 1, \qquad \forall\, 1 \leq p \leq m,\ K \in \mathcal{K}, \tag{3}$$

$$\sum_{a=1}^{p} y_{d_r a} + \sum_{b=p}^{m} y_{ib} + x_r \leq 2, \qquad \forall\, r \in R,\ i \in S_r^I,\ 1 \leq p \leq m, \tag{4}$$

$$x_r \in \{0, 1\}, \qquad \forall\, r \in R, \tag{5}$$

$$y_{ip} \in \{0, 1\}, \qquad \forall\, i \in S,\ 1 \leq p \leq m. \tag{6}$$

We refer to the convex hull of feasible integer solutions to (1)–(6) as P. Constraint (1) states that each disk is assigned to at most one level. Constraint (2) states that a disk with a visible arc must be assigned to a level, and (3) says that overlapping disks can not be at the same level. Constraint (4) ensures that arc r is only visible if d_r is above all other disks that contain r.

4 Polyhedral Study of P

In this section, we obtain the dimension of P and determine which inequalities in the original formulation (1)–(6) define facets. For the sake of brevity, we omit the proofs of Propositions 2 to 5, which are based on the *direct method*, that is, they essentially enumerate affinely independent points belonging to a given polytope to establish its dimension. For those proofs, see [8]. We include here, however, the proofs that inequalities are facet-defining whenever they employ the *indirect method*. Both direct and indirect methods are discussed in Theorem 3.6, Part I.4 of [10].

Proposition 2. *The dimension of P is $nm + |R|$.*

Proposition 3. *Given an arc $r \in R$, the inequality $x_r \geq 0$ defines a facet of P, whereas the inequality $x_r \leq 1$ does not.*

The inequality $x_r \leq 1$ is not facet-defining for P because it is implied by the combination of (1) and (2).

Proposition 4. *Given a disk $i \in S$, and a level $1 \leq p \leq m$, the inequality $y_{ip} \geq 0$ defines a facet of P, whereas the inequality $y_{ip} \leq 1$ does not.*

The inequality $y_{ip} \leq 1$ does not define a facet of P because it is implied by (1).

Proposition 5. *Given a disk $i \in S$, (1) defines a facet of P.*

Proposition 6. *Given an arc $r \in R$, (2) defines a facet of P.*

Proof. We use the indirect method. Let $\mathbf{x} = (y, x)$ and let $\pi\mathbf{x} \leq \pi_0$ be a valid inequality for P whose induced face contains the face F induced by (2). We will show that $\pi\mathbf{x} \leq \pi_0$ is a scalar multiple of (2). Because the origin is a feasible solution that satisfies (2) as an equality, we have that $\pi_0 = 0$. Let $1 \leq p \leq m$ and $\mathbf{x}_{rp}$ satisfy $y_{d_r p} = x_r = 1$, with all other variables equal to zero. It is easy to see that $\mathbf{x}_{rp}$ is feasible and satisfies (2) as an equality. Then,

$$\pi\mathbf{x}_{rp} = \pi_{d_r p} + \pi_r = \pi_0 = 0 \ , \tag{7}$$

where $\pi_{d_r p}$ is the component of vector π that multiplies variable $y_{d_r p}$ in $\mathbf{x}_{rp}$, and π_r is the component that multiplies x_r. Therefore, $\pi_{d_r p} = -\pi_r$. By varying the value of p, (7) implies that

$$\pi_{d_r 1} = \pi_{d_r 2} = \cdots = \pi_{d_r m} = -\pi_r = \alpha_r \ . \tag{8}$$

To complete the proof, we need to show that all remaining components of π are equal to zero.

Let $r' \in R \setminus \{r\}$ with $d_{r'} = d_r$. Consider the vector $\mathbf{x} = \mathbf{x}_{rp} + e_{nm+r'}$, whose components are all zero except $y_{d_r p}$, x_r and $x_{r'}$ which have value one. Clearly, $\mathbf{x}$ is feasible and belongs to F. Therefore, we have $\pi_{r'} = 0$. From now on, let us assume that $d_{r'} \neq d_r$. For any $p \in \{1, \ldots, m\}$, by setting $y_{d_{r'} p} = 1$ and all other variables equal to zero, we obtain a feasible vector $\mathbf{x}$ that lies on F. As a consequence, $\pi\mathbf{x} = \pi_0$, implying that $\pi_{d_{r'} p} = 0$ for all $r' \neq r$ and all p. Similarly, choosing $\mathbf{x}$ such that $y_{d_{r'} p} = x_{r'} = 1$ with all the remaining components set to zero, we generate a feasible point in F which yields $\pi_{r'} = 0$ for all $r' \neq r$. □

Proposition 7. *Given $1 \leq p \leq m$ and $K \in \mathcal{K}$, (3) defines a facet of P.*

Proof. We use the indirect method. Let $\mathbf{x} = (y, x)$ and let $\pi\mathbf{x} \leq \pi_0$ be a valid inequality for P whose induced face F contains the face of P induced by (3). We will show that $\pi\mathbf{x} \leq \pi_0$ is a scalar multiple of (3). In this proof, the components of vector π are identified as in Proposition 6.

First let us partition the variables into five classes: (i) y_{jp} with $V(j) \in K$; (ii) y_{jq} with $V(j) \in K$, and $q \neq p$; (iii) y_{jq} with $V(j) \notin K$; (iv) x_r with $V(d_r) \in K$; and (v) x_r with $V(d_r) \notin K$. We now exhibit feasible points that satisfy (3) as an equality to determine the values of the coefficients of vector π for each class of variables defined above. For each choice of $\mathbf{x}$ given below, undefined variables are assumed to be equal to zero. (i) Let $\mathbf{x}$ have $y_{ip} = 1$. Then, $\pi\mathbf{x} = \pi_{ip} = \pi_0$; (ii) Let $i \in S$ be such that $V(i) \in K$, and let $\mathbf{x}$ have $y_{jq} = y_{ip} = 1$. Then, $\pi\mathbf{x} = \pi_{jq} + \pi_{ip} = \pi_0$, which implies $\pi_{jq} = 0$ because of (i); (iii) There exists

$i \in S$ with $V(i) \in K$ such that $V(j)$ is not adjacent to $V(i)$ (otherwise, $V(j)$ would be a vertex of K). For each $1 \leq q \leq m$, let $\mathbf{x}$ have $y_{jq} = y_{ip} = 1$. Then, as in (ii), $\pi_{jq} = 0$; (iv) If $\mathbf{x}$ satisfies $y_{d_r p} = x_r = 1$, we have $\pi\mathbf{x} = \pi_{d_r p} + \pi_r = \pi_0$, which implies $\pi_r = 0$; (v) As in (iii), we can find an $i \in S$ with $V(i) \in K$ such that $V(d_r)$ is not adjacent to $V(i)$. Let $\mathbf{x}$ have $y_{d_r 1} = y_{ip} = x_r = 1$. Then, $\pi\mathbf{x} = \pi_{d_r 1} + \pi_{ip} + \pi_r = \pi_0$, which implies $\pi_r = 0$. □

Proposition 8. *Given an arc $r \in R$, $i \in S_r^I$ and $1 \leq p \leq m$, (4) does not define a facet of P, but* (9) *does if $1 \leq p < m$.*

$$\sum_{a=1}^{p} y_{d_r a} + \sum_{b=p}^{m} y_{ib} + x_r \leq 1 + \sum_{a=1}^{m} y_{d_r a} \tag{9}$$

Proof. We first show that inequality (4) does not define a facet of P. To this end, let F denote the face defined by (4) in P. Now, we claim that all feasible points in F satisfy inequality (1) at equality for $i = d_r$ (otherwise d_r is not assigned to a level, x_r is zero because of (2), and the left-hand side of (4) is at most one). Since the P is full-dimensional, F cannot be a facet of it.

Notice that, by defining the binary variable $z = \sum_{a=1}^{m} y_{d_r a}$ and lifting this variable in (4), we obtain inequality (9). We now prove that the latter inequality is facet defining for P under the assumptions made in the proposition.

Initially, we observe that (9) is not facet-defining for P when $p = m$ because it is clearly dominated by (11) or (14), depending on what kind of arc r is. Moreover, for convenience, we rewrite (9) as:

$$\sum_{b=p}^{m} y_{ib} - \sum_{a=p+1}^{m} y_{d_r a} + x_r \leq 1 \ . \tag{10}$$

We use the indirect method. Let $\mathbf{x} = (y, x)$ and let $\pi\mathbf{x} \leq \pi_0$ be a valid inequality for P whose induced face F contains the face of P induced by (10). We will show that $\pi\mathbf{x} \leq \pi_0$ is a scalar multiple of (10). In this proof, the components of vector π are identified as in Proposition 6. We partition the variables into ten classes and establish the appropriate corresponding coefficients in vector π. For each choice of $\mathbf{x}$ given below, undefined variables are assumed to be equal to zero and the vector is easily shown to be feasible and to lie on F. (i) y_{il} for $p \leq l \leq m$: Let $\mathbf{x}$ have $y_{il} = 1$. Then, $\pi\mathbf{x} = \pi_{il} = \pi_0$. (ii) y_{jm} for all $j \in S \setminus \{d_r, i\}$: Let $\mathbf{x}$ have $y_{i(m-1)} = y_{jm} = 1$. Then, $\pi\mathbf{x} = \pi_{i(m-1)} + \pi_{jm} = \pi_0$ which, from the previous result, implies that $\pi_{jm} = 0$. (iii) y_{jl} for all $j \in S \setminus \{d_r, i\}$ and $1 \leq l \leq m-1$: Let $\mathbf{x}$ have $y_{im} = y_{jl} = 1$. Then, $\pi\mathbf{x} = \pi_{im} + \pi_{jl} = \pi_0$ which, from (i), implies that $\pi_{jl} = 0$. (iv) $y_{d_r l}$ for $1 \leq l \leq p$: Let $\mathbf{x}$ have $y_{im} = y_{d_r l} = 1$. Then, $\pi\mathbf{x} = \pi_{im} + \pi_{d_r l} = \pi_0$ which, from (i), implies that $\pi_{d_r l} = 0$. (v) x_r: Let $\mathbf{x}$ have $y_{d_r p} = x_r = 1$. Then, $\pi\mathbf{x} = \pi_{d_r p} + \pi_r = \pi_0$ which, from (iv), implies that $\pi_r = \pi_0$. (vi) y_{il} for $1 \leq l \leq p-1$: Let $\mathbf{x}$ have $y_{d_r p} = x_r = y_{il} = 1$. Then, $\pi\mathbf{x} = \pi_{d_r p} + \pi_r + \pi_{il} = \pi_0$ which, from (iv) and (v), implies that $\pi_{il} = 0$. (vii) x_q for all $j \in S \setminus \{d_r, i\}$ and all arcs q of disk j: Let

$\mathbf{x}$ have $y_{i(m-1)} = y_{jm} = x_q = 1$. Then, $\pi\mathbf{x} = \pi_{i(m-1)} + \pi_{jm} + \pi_q = \pi_0$ which, from (i) and (ii), implies that $\pi_q = 0$. (viii) x_q for all arcs q of disk i: Let $\mathbf{x}$ have $y_{im} = x_q = 1$. Then, $\pi\mathbf{x} = \pi_{im} + \pi_q = \pi_0$ which, from (i), implies that $\pi_q = 0$. (ix) x_q for all arcs q of disk d_r except arc r: Let $\mathbf{x}$ have $y_{d_r p} = x_r = x_q = 1$. Then, $\pi\mathbf{x} = \pi_{d_r p} + \pi_r + \pi_q = \pi_0$ which, from (iv) and (v), implies that $\pi_q = 0$. (x) $y_{d_r l}$ for $p{+}1 \leq l \leq m$: Let $\mathbf{x}$ have $y_{ip} = y_{d_r l} = x_r = 1$. Then, $\pi\mathbf{x} = \pi_{ip} + \pi_{d_r l} + \pi_r = \pi_0$ which, from (i) and (ix), implies that $\pi_{d_r l} = -\pi_0$. □

5 Strengthening the ILP Formulation

The geometric nature of PSMs enables us to obtain new valid inequalities by observing that certain groups of arcs cannot be visible simultaneously due to a physical impossibility. In the sequel, $\mathcal{A}$ is an arrangement of disks on a plane. We use the following additional data sets:

- $D_f \equiv$ set of disks that contain face f.
- $B_f \equiv$ set of arcs that form the boundary of face f. $B_f^+ = \{r \in B_f \mid d_r \in D_f\}$ and $B_f^- = B_f \setminus B_f^+$.
- $I_f \equiv$ set of disks whose borders contain an arc in B_f.
- $C_f \equiv$ set of disks that contain face f in their interior ($C_f = D_f \setminus I_f$).

Consider the arrangement in Figure 2. The boundary of face g is formed by arcs r_2, r_3, and r_4. We have $B_g = \{r_2, r_3, r_4\}$, $D_g = \{d_{r_4}\}$, $B_g^+ = \{r_4\}$, $B_g^- = \{r_2, r_3\}$, $I_g = \{d_{r_2}, d_{r_3}, d_{r_4}\}$, and $C_g = \emptyset$. In the arrangement of Figure 3, the boundary of face f is formed by arcs r_1, r_2, and r_3. Therefore, we have $B_f = B_f^+ = \{r_1, r_2, r_3\}$, $D_f = \{d_{r_1}, d_{r_2}, d_{r_3}, d_{r_4}\}$, $I_f = \{d_{r_1}, d_{r_2}, d_{r_3}\}$, and $C_f = \{d_{r_4}\}$. If one of the arcs in B_f is visible in a drawing, the other two cannot appear. Moreover, if d_{r_4} is assigned to the topmost level, f will not appear. This leads to the valid inequality $y_{d_{r_4} m} + x_{r_1} + x_{r_2} + x_{r_3} \leq 1$. In general, we have the following result:

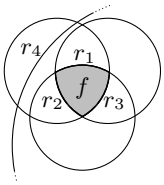

Fig. 3. Arcs r_1, r_2, and r_3 of face f cannot be visible simultaneously

Proposition 9. *Let f be a face of $\mathcal{A}$ with $|B_f^+| \geq 1$. If $|C_f| \geq 1$ or $|B_f^+| \geq 2$, then (11) defines a facet of P.*

$$\sum_{i \in C_f} y_{im} + \sum_{r \in B_f^+} x_r \leq 1 \tag{11}$$

Proof. To prove validity, note that for every arc $r \in B_f^+$, all the arcs in $B_f^+ \setminus \{r\}$ are in the interior of d_r. Therefore, if r is visible, no other arc of $B_f^+ \setminus \{r\}$ can be visible, which implies $\sum_{r \in B_f^+} x_r \leq 1$. Moreover, if a disk in C_f is at the top level (m), we must have $\sum_{r \in B_f^+} x_r = 0$, so it suffices to show that $\sum_{i \in C_f} y_{im} \leq 1$. Because all the disks in C_f contain f, the corresponding vertices in G_S form a clique. Hence, at most one of those disks can be assigned to level m because of (3). If $|B_f^+| = 0$, (11) is dominated by (3). If $|C_f| = 0$ and $|B_f^+| = 1$, (11) reduces to $x_r \leq 1$, which is not facet-defining due to Proposition 3.

To prove that (11) is facet-defining for P under the assumptions stated above, we use the indirect method. Let $\mathbf{x} = (y, x)$ and let $\pi\mathbf{x} \leq \pi_0$ be a valid inequality for P whose induced face contains the face F induced by (11). We will show that $\pi\mathbf{x} \leq \pi_0$ is a scalar multiple of (11). As usual, this is done by exhibiting several vectors that can be easily shown to be feasible and lying on F. Moreover, the components of vector π are also identified as in Proposition 6.

Let $r \in B_f^+$, $1 \leq p \leq m$, and let $\mathbf{x}_{rp}$ satisfy $y_{d_r p} = x_r = 1$, with all other variables equal to zero. Clearly, $\mathbf{x}_{rp}$ satisfies (11) as an equality, $\mathbf{x}_{rp} \in P$, and

$$\pi\mathbf{x}_{rp} = \pi_{d_r p} + \pi_r = \pi_0 \ . \tag{12}$$

By varying the value of p, (12) implies that, for any $r \in B_f^+$,

$$\pi_{d_r 1} = \pi_{d_r 2} = \cdots = \pi_{d_r m} = \alpha_r \ . \tag{13}$$

Let $r \in B_f^+$ and $q \notin B_f^+$. If $p_q < p_r \leq m$, let $\mathbf{x}_{rqp_r p_q}$ satisfy $y_{d_r p_r} = y_{d_q p_q} = x_r = 1$, with all other variables equal to zero. This gives $\pi\mathbf{x}_{rqp_r p_q} = \pi_{d_r p_r} + \pi_{d_q p_q} + \pi_r = \pi_0 + \pi_{d_q p_q}$ (using (12)), which implies $\pi_{d_q p_q} = 0$. If $p_r < p_q = m$, there are two cases: (i) $d_q \notin C_f$: we can still set $y_{d_r p_r} = y_{d_q m} = x_r = 1$, which yields $\pi_{d_q m} = 0$ as above; (ii) $d_q \in C_f$: setting $y_{d_q m} = 1$ and all remaining variables equal to zero, we conclude that $\pi_{d_q m} = \pi_0$.

We now deal with coefficients of π corresponding to x variables associated with arcs outside B_f^+. Let $q \notin B_f^+$. There are two cases to consider: (i) $d_q \in C_f$: let $\mathbf{x}_{qm}$ satisfy $y_{d_q m} = x_q = 1$, with all other variables equal to zero. Then, $\pi\mathbf{x}_{qm} = \pi_{d_q m} + \pi_q = \pi_0 + \pi_q = \pi_0$. Therefore, $\pi_q = 0$; (ii) $d_q \notin C_f$: Take $r \in B_f^+$ and let $\mathbf{x}_{qr21}$ satisfy $y_{d_q 2} = y_{d_r 1} = x_q = x_r = 1$ (even if $q \in B_f^-$, both q and r will be visible). Then, $\pi\mathbf{x}_{qr21} = \pi_{d_q 2} + \pi_{d_r 1} + \pi_q + \pi_r = \pi_0 + \pi_q = \pi_0$. Hence, $\pi_q = 0$.

If $|B_f^+| \geq 2$, let $p_1 > p_2$, r_1 and $r_2 \in B_f^+$, and let $\mathbf{x}_{r_1 r_2 p_1 p_2}$ satisfy $y_{d_{r_1} p_1} = y_{d_{r_2} p_2} = x_{r_1} = 1$, with all other variables equal to zero. Then, $\pi\mathbf{x}_{r_1 r_2 p_1 p_2} = \pi_{d_{r_1} p_1} + \pi_{d_{r_2} p_2} + \pi_{r_1} = \alpha_{r_1} + \alpha_{r_2} + \pi_{r_1} = \pi_0$, yielding $\alpha_r = 0$ for all r, because of (12) and (13). Consequently, $\pi_r = \pi_0$ for all $r \in B_f^+$. To achieve the same results when $|B_f^+| = 1$, we assume $|C_f| \geq 1$. Let $\mathbf{x}_{qrm}$ satisfy $y_{d_q m} = y_{d_r(m-1)} = 1$, where $d_q \in C_f$ and $B_f^+ = \{r\}$. Then, $\pi\mathbf{x}_{qrm} = \pi_{d_q m} + \pi_{d_r(m-1)} = \pi_0 + \pi_{d_r(m-1)}$, which implies $\pi_{d_r(m-1)} = 0$. Consequently, because of (13), $\pi_{d_r p} = 0$ for all p, and $\pi_r = \pi_0$. □

Proposition 10. *Let f be a face of $\mathcal{A}$ with $|B_f^-| \geq 1$. For each $r \in B_f^-$, (14) defines a facet of P.*

$$\sum_{i \in D_f} y_{im} + x_r \leq 1 \tag{14}$$

Proof. The inequality is clearly valid. To prove that (14) is facet-defining for P under the assumptions stated above, we use the indirect method as in the proof of Proposition 9. Let $1 \leq p \leq m$, and let $\mathbf{x}_{rp}$ satisfy $y_{d_r p} = x_r = 1$, with all other variables equal to zero. Clearly, $\mathbf{x}_{rp}$ satisfies (14) as an equality, $\mathbf{x}_{rp} \in P$, and

$$\pi \mathbf{x}_{rp} = \pi_{d_r p} + \pi_r = \pi_0 \ . \tag{15}$$

By varying the value of p, (15) implies that

$$\pi_{d_r 1} = \pi_{d_r 2} = \cdots = \pi_{d_r m} = \alpha_r \ . \tag{16}$$

Let $q \neq r$. If $p_q < p_r \leq m$, let $\mathbf{x}_{rqp_r p_q}$ satisfy $y_{d_r p_r} = y_{d_q p_q} = x_r = 1$, with all other variables equal to zero. This gives $\pi \mathbf{x}_{rqp_r p_q} = \pi_{d_r p_r} + \pi_{d_q p_q} + \pi_r = \pi_0 + \pi_{d_q p_q}$ (using (15)), which implies $\pi_{d_q p_q} = 0$. If $p_r < p_q = m$, there are two cases: (i) $d_q \notin D_f$: we can still set $y_{d_r p_r} = y_{d_q m} = x_r = 1$, which yields $\pi_{d_q m} = 0$ as above; (ii) $d_q \in D_f$: setting $y_{d_q m} = 1$ and all remaining variables equal to zero, we conclude that $\pi_{d_q m} = \pi_0$.

We now deal with coefficients of π corresponding to x variables associated with arcs $q \neq r$. There are two cases to consider: (i) $d_q \in D_f$: let $\mathbf{x}_{qm}$ satisfy $y_{d_q m} = x_q = 1$, with all other variables equal to zero. Then, $\pi \mathbf{x}_{qm} = \pi_{d_q m} + \pi_q = \pi_0 + \pi_q$. Therefore, $\pi_q = 0$; (ii) $d_q \notin D_f$: Let $\mathbf{x}_{qr21}$ satisfy $y_{d_q 2} = y_{d_r 1} = x_q = x_r = 1$. Then, $\pi \mathbf{x}_{qr21} = \pi_{d_q 2} + \pi_{d_r 1} + \pi_q + \pi_r = \pi_0 + \pi_q$. Hence, $\pi_q = 0$.

Finally, let $d_q \in D_f$ and let $\mathbf{x}_{qrm}$ satisfy $y_{d_q m} = y_{d_r (m-1)} = 1$. Then, $\pi \mathbf{x}_{qrm} = \pi_{d_q m} + \pi_{d_r (m-1)} = \pi_0 + \pi_{d_r (m-1)}$, which implies $\pi_{d_r (m-1)} = 0$. Consequently, because of (16), $\alpha_r = 0$ and $\pi_r = \pi_0$. □

A vertex of an arrangement is *non-degenerate* if it is an intersection point of exactly two disks or, equivalently, four arcs, as shown in Figure 4(i). Since each arc can be either visible or not, there are 16 potential assignments of values to their respective x variables. In a feasible solution, however, only the five assignments shown in Figure 4(ii)–(vi) are possible (dashed arcs are obscured). This observation gives rise to Proposition 11.

Proposition 11. *Given a non-degenerate vertex of an arrangement as shown in Figure 4(i), (17)–(20) are valid and define facets of P.*

$$x_{r_1} \geq x_{r_3} \tag{17}$$

$$x_{r_2} \geq x_{r_4} \tag{18}$$

$$x_{r_3} + x_{r_4} \geq x_{r_1} \tag{19}$$

$$x_{r_3} + x_{r_4} \geq x_{r_2} \tag{20}$$

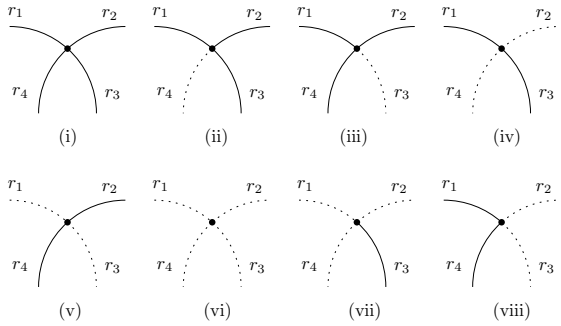

Fig. 4. A non-degenerate vertex (i), five feasible arc configurations: (ii)–(vi), and two infeasible ones: (vii) and (viii)

Proof. It is easy to see that the five feasible configurations shown in Figure 4(ii)–(vi) satisfy (17)–(20). In addition, because of symmetry, it suffices to show that (17) and (19) are facet defining. We will use the indirect method and define $\pi\mathbf{x} \leq \pi_0$ as usual (see the proof of Proposition 9).

The zero vector satisfies (17) as an equality, which yields $\pi_0 = 0$. Given $i \in S$ and $1 \leq p \leq m$, let $\mathbf{x}_{ip}$ be such that $y_{ip} = 1$ and all other variables are equal to zero. Clearly, $\mathbf{x}_{ip}$ belongs to P and satisfies (17) as an equality. Because $\pi\mathbf{x}_{ip} = \pi_{ip} = \pi_0$, we have that $\pi_{ip} = 0$ for all i and p. Given $1 \leq p \leq m$ and $r \in R \setminus \{r_1, r_3\}$, let $\mathbf{x}_{rp}$ satisfy $y_{d_r p} = x_r = 1$ and have zeroes everywhere else. Again, $\mathbf{x}_{rp}$ satisfies (17) as an equality and $\mathbf{x}_{rp} \in P$. Since $\pi\mathbf{x}_{rp} = \pi_{d_r p} + \pi_r = \pi_0$, we have $\pi_r = 0$. Finally, given $1 \leq p \leq m$, let $\mathbf{x}_{r_1 r_3}$ be such that $x_{r_1} = x_{r_3} = y_{d_{r_1} p} = 1$ (note that $d_{r_1} = d_{r_3}$). Then, $\pi\mathbf{x}_{r_1 r_3} = \pi_{r_1} + \pi_{r_3} + \pi_{d_{r_1} p} = \pi_0$. Because $\pi_{d_{r_1} p} = \pi_0 = 0$, we have that $\pi_{r_1} = -\pi_{r_3}$, as desired.

We now show that (19) is facet defining. By repeating the arguments of the previous paragraph, we can show that $\pi_0 = 0$, $\pi_{ip} = 0$ for all i and p, and $\pi_r = 0$ for all $r \in R \setminus \{r_1, r_3, r_4\}$. Let $\mathbf{x}_{r_1 r_3}$ be such that $x_{r_1} = x_{r_3} = y_{d_{r_1} 1} = 1$ and all other variables are equal to zero. Then, $\pi\mathbf{x}_{r_1 r_3} = \pi_{r_1} + \pi_{r_3} + \pi_{d_{r_1} 1} = \pi_0$, which implies $\pi_{r_1} = -\pi_{r_3}$. Finally, let $\mathbf{x}_{r_1 r_4}$ be such that $x_{r_1} = x_{r_4} = y_{d_{r_1} 1} = y_{d_{r_4} 2} = 1$, with all other variables equal to zero. Then, $\pi\mathbf{x}_{r_1 r_4} = \pi_{r_1} + \pi_{r_4} + \pi_{d_{r_1} 1} + \pi_{d_{r_4} 2} = \pi_0$, which also implies that $\pi_{r_1} = -\pi_{r_4}$. □

6 Decomposition Techniques

To reduce the size of the ILP model, we introduce decomposition techniques that allow us to consider smaller sets of disks at a time.

Without loss of generality, we assume that G_S is connected. Otherwise, each of its connected components can be treated separately. In addition, we can decompose a connected component around articulation points of G_S. Consider the example in Figure 5(i), in which $S = \{a, b, c, d, e, v\}$. The node corresponding to disk v, i.e. $V(v)$, is an articulation point of G_S because its removal disconnects

the graph into three connected components: $\{a, b\}$, $\{c, d\}$, and $\{e\}$. By adding v to each of these components, we get instances (ii), (iii), and (iv) of Figure 5, which are solved independently. Those three optimal solutions can be combined into an optimal solution for the entire set S by preserving the relative order of the disks in each solution. Proposition 12 formalizes this idea.

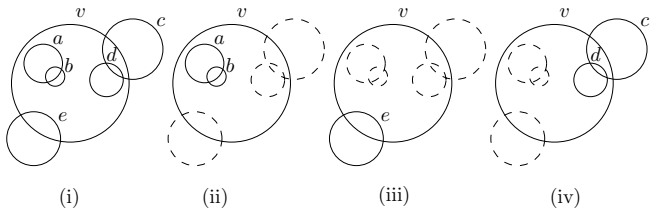

Fig. 5. An instance that allows for decomposition

Proposition 12. *Let S be a set of disks such that G_S is not 2-connected and let v be a disk corresponding to an articulation point of G_S. Let S_k contain v plus the disk set of the k-th connected component obtained after the removal of $V(v)$ from G_S. The optimal solutions for each S_k can be combined into an optimal solution for S in polynomial time.*

Proof. Let $V(v)$ be an articulation point of G_S and let v be its corresponding disk in S (note that articulation points can be found in $O(|E|)$ time [3]). Using the notation introduced in the proposition, consider the disk subsets S_i and S_j corresponding to any two distinct connected components of $G_S - V(v)$. By definition, the pieces of v's border contained in $S_i \setminus \{v\}$ and in $S_j \setminus \{v\}$ are disjoint. Hence, the optimal solutions of the problems defined over S_i and S_j do not influence each other. In other words, the relative order imposed by those solutions onto the disks of each such subset is optimal for the complete set of disks S. If we consider these orders as representing an orientation of the arcs of G_S, we have a directed acyclic graph G'_S. The optimal assignment of disks to levels can be obtained in polynomial time from a topological ordering of G'_S. □

If the graph of a connected component (G_{S_k}) is not 2-connected and has an articulation point, the above procedure can be applied recursively.

From Figure 5(ii), it is clear that there exists an optimal solution in which a and b are drawn above v. Hence, we can consider the pair a, b as a separate instance, and v as another. Proposition 13, whose proof can be seen in [8], formalizes this idea.

Proposition 13. *Let S be a set of disks and let H_S be a directed graph with one node for every disk in S and an arc from node i to node j whenever a portion of the border of i's disk is contained in the interior of j's disk. Let S_k be the disk set of the k-th strongly connected component of H_S. The optimal solutions for each S_k can be combined into an optimal solution for S in polynomial time.*

7 Computational Experiments

Our experiments are performed on the same set of instances used in the paper by Cabello et al. [1]. Instances *City 156* and *City 538* represent the 156 and 538 largest American cities, respectively, in which the area of each disk is proportional to the city's population. Instances *Deaths* and *Magnitudes* represent the death count and Richter scale magnitude of 602 earthquakes worldwide, respectively. Disks are placed at the epicenters of each earthquake, and disk areas are proportional to the corresponding quantities [11]. When disks in an instance coincide, we replace them by a single disk whose border is the total border length of the original disks. This is possible because we can assume that such disks would occupy adjacent levels in an optimal solution. This pre-processing step reduces the number of disks in Deaths and Magnitudes to 573 and 491, respectively.

In Table 1, column *Connected* shows the number of connected components in G_S for each instance, with the number of disks in the largest component in parentheses. Column *Strongly Connected* shows the resulting number of components (and largest component) after we apply the decomposition of Proposition 13. Proposition 12 yields further decomposition, as shown under column *2-Connected*. The reductions in problem size are remarkable. City 538 can now be solved by optimizing over sets of disks no larger than one tenth of its original size. Solving the original instances is now equivalent to solving 671 significantly smaller instances. Overall, the size of our largest instance dropped from 573 to 116 disks.

Table 1. Number of components and largest component before/after decomposition

Instance	# Disks	Connected	Strongly Connected	2-Connected
City 156	156	38 (57)	45 (56)	53 (29)
City 538	538	185 (98)	213 (94)	240 (53)
Deaths	573	134 (141)	317 (85)	333 (70)
Magnitudes	491	31 (155)	31 (155)	45 (116)

Our cut-and-branch algorithm uses the ILP model of Section 3, modeling (1) as SOS1, substituting (9) for (4), and adding (11), (17)–(20) at the root node. (Inequalities (14) did not help computationally.) Because $|\mathcal{K}|$ can be exponentially large, rather than including all of (3), we heuristically look for an edge covering of G_S by maximal cliques [9]. Alternatively, we also tried replacing (3) with $y_{ip} + y_{jp} \leq 1$ for each level p and all $(i, j) \in E$. Although theoretically weaker, the latter formulation performed better in our experiments. This might be explained by the sparser coefficient matrix of the weaker model, which typically yields easier-to-solve linear relaxations. Finally, instead of computing the exact value of m as in Proposition 1, which is NP-Hard [6], we use $m = n$ in every run because the exact m is equal to n in many of the large components.

Our model was implemented in C++, using CGAL [13] for data extraction. We use *XPRESS-Optimizer* [4] version 20.00 to solve each problem on a 2.4GHz

Intel® Core™2 Quad processor, with 4GB of RAM. We limit each run to five hours of CPU time.

7.1 Numerical Results

For comparison purposes, we use the $O(n^2 \log n)$ heuristic from [1,2] to find good feasible solutions. Despite being a Max-Min heuristic, its solutions also perform well in terms of the Max-Total objective.

Out of the 671 components obtained through decomposition, all but the five or six largest ones from each original instance are easily handled by our strengthened ILP model. We will focus on them first.

For components with $|S_k| \leq 2$, the solution is trivial. For the remaining easy-to-solve components, we summarize our results in Table 2. Column *Comp. w/* $|S_k| > 2$ indicates how many easy components from the corresponding original instance have more than two disks. The next nine columns indicate the minimum, average, and maximum values of component size, followed by the number of search nodes and CPU time required to find an optimal solution, respectively. When compared to the heuristic solutions, the optimal solutions to the 67 problems from Table 2 are 13.2% better on average (min = 0.0% and max = 158.4%).

Table 2. Average results over smallest non-trivial components of each instance

Original	Comp. w/	$\|S_k\|$			Nodes			Time (in sec.)		
Instance	$\|S_k\| > 2$	Min	Avg	Max	Min	Avg	Max	Min	Avg	Max
City 156	11	3	5.3	14	1	20.8	213	0	3.5	38
City 538	20	3	5.4	12	1	11.9	145	0	0.4	5
Deaths	22	3	4.7	10	1	5.8	93	0	0.1	1
Magnitudes	14	3	4.7	10	1	1.8	7	0	0.1	1

The results obtained with the five (or six) most challenging components of each original instance appear in Table 3. Component names are written as "α-β-γ (δ)", where α identifies the instance, β-γ indicates that this is the γ-th component generated by Proposition 12 when applied to the β-th component generated by Proposition 13, and δ is the number of disks. In Table 3, *Base Value* represents the total border length of arcs r that are visible in any feasible solution ($S_r^I = \emptyset$). This value is subtracted from the solution values in the remaining columns. *Best Feasible* and *Best UB* are the best lower and upper bounds on the optimal value found within the time limit, respectively (optimal solutions appear in bold). Column *% Gap* shows the relative difference between the lower and upper bounds, and *% Above Heur.* indicates how much better the best known lower bound is with respect to the heuristic solution discussed above.

Instance *City 156* presented no difficulties, having all of its five largest components solved in less than 8 minutes. In Figure 6, we can perceive subtle differences, highlighted in light gray, between the optimal solutions for Max-Min

Table 3. Results on largest components from each original problem instance

Component	Base Value	Best Feasible	Best UB	% Gap	% Above Heur.	Nodes	Time (s)
156-18-0 (7)	63.97	**12.91**	**12.91**	0	0	1	0
156-3-2 (8)	39.84	**40.99**	**40.99**	0	8.5	7	0
156-3-0 (14)	66.15	**71.17**	**71.17**	0	7.8	213	39
156-2-0 (26)	167.22	**138.05**	**138.05**	0	3.1	5949	381
156-2-1 (29)	219.36	**153.85**	**153.85**	0	1.4	117	10
538-47-2 (17)	26.75	**25.27**	**25.27**	0	2.0	2463	1259
538-3-0 (26)	34.27	**39.19**	**39.19**	0	15.0	23589	9562
538-29-1 (26)	46.48	**36.40**	**36.40**	0	4.3	1143	1260
538-1-6 (29)	21.98	43.51	47.05	8.0	9.6	2399	18000
538-1-0 (51)	77.37	82.13	107.35	30.7	0.0	22	18000
538-24-0 (53)	18.98	58.50	186.23	218.3	0.0	1	18000
death-6-0 (12)	953.08	**60.16**	**60.16**	0	0.0	51	1
death-8-0 (14)	68.05	**39.65**	**39.65**	0	3.1	87	0
death-0-0 (24)	175.78	**145.74**	**145.74**	0	5.7	4925	199
death-3-0 (24)	441.75	**323.18**	**323.18**	0	1.3	3919	210
death-2-0 (70)	725.28	964.66	1652.02	71.2	0.0	1	18000
mag-5-1 (25)	214.92	**593.74**	**593.74**	0	3.7	965	9609
mag-6-0 (26)	217.21	579.58	610.99	5.4	5.0	3385	18000
mag-1-1 (39)	417.32	919.28	1350.23	46.9	0.0	3	18000
mag-5-0 (81)	601.79	1741.24	2317.66	33.1	0.0	1	18000
mag-1-0 (113)	581.41	2743.68	-	-	0.0	1	18000
mag-7-0 (116)	700.37	2622.46	-	-	0.0	1	18000

and Max-Total problems for this instance. We found optimal or near optimal solutions to the first four of the largest components of *City 538*, with significant improvements in quality with respect to the heuristic solutions. The two largest

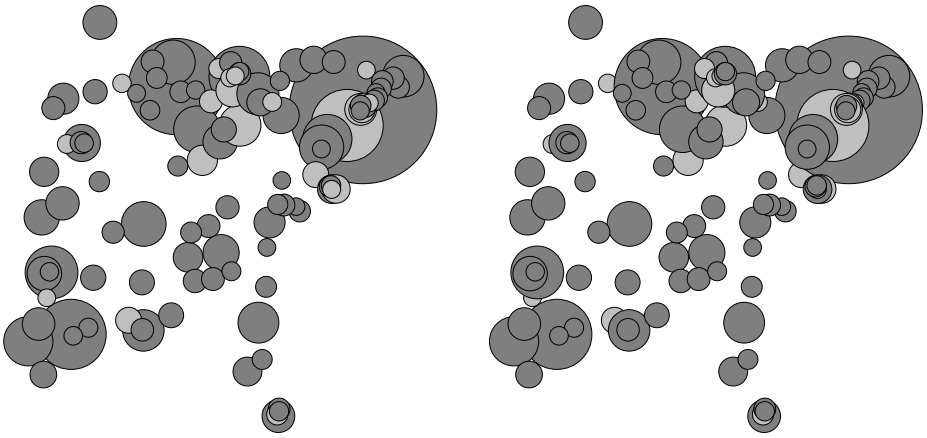

Fig. 6. Optimal solutions for *City 156* to Max-Min [2] and Max-Total problems, respectively

components of *City 538* turned out to be more challenging, with sizable gaps remaining after five hours of computation. All but one of the largest earthquake death components were solved to optimality.

As was the case with component 538-24-0, the time limit was exhausted during the solution of death-2-0 even before branching started. The largest components obtained from the decomposition of earthquake magnitudes turned out to be the most challenging ones. Note that we do not have valid upper bounds for instances mag-1-0 and mag-7-0 because the time limit was not even enough to solve their first linear relaxation. Overall, we were able to find optimal solutions to 662 out of the 671 components derived from our original four instances.

Cutting planes (11) and (17)–(20) were essential in achieving the results in tables 2 and 3. With those cuts, the number of search nodes was 54 times smaller on average, with some cases achieving reductions of almost three orders of magnitude. (Five of the 21 hardest components — six overall — would not have been solved to optimality without cuts.) As a consequence, computation times were also drastically reduced.

Because of its direct relationship to the amount of overlapping between disks, the number of arcs in an instance/component is a better measure of difficulty than the number of disks. Our strengthened ILP model appears to be capable of handling about 600 to 700 arcs in five hours of CPU which, for our benchmark set, roughly corresponds to instances having between 24 and 26 disks. Table 4 contains more details about the size of our five largest components and how big their ILP formulation is before and after the inclusion of cuts. Because the number of cuts is small, we opted not to implement a branch-and-cut algorithm.

Table 4. Number of arcs and size of ILP formulation for the 5 largest components

Component	# Disks	# Arcs	# Cols.	# Rows before cuts	# Rows after cuts
538-24-0	53	3753	6562	3026565	3035839
death-2-0	70	1366	6266	620970	624115
mag-5-0	81	2059	8620	914490	919623
mag-1-0	113	4318	17087	3733407	3744116
mag-7-0	116	3759	17215	2792468	2801845

8 Conclusion

We propose a novel ILP formulation to optimize stacking drawings of proportional symbol maps (PSMs) with the objective of maximizing the total visible border of its symbols (opaque disks, in our case). By studying structural and polyhedral aspects of PSMs, we devised effective decomposition techniques and new families of facet-defining inequalities that greatly reduce the computational effort required to solve the problem. These improvements enabled us to find the first provably optimal solutions to some of the real-world instances studied in [1,2]. Because solving PSM instances still pose great challenges when the

number of arcs exceeds 1000 or so, we continue to study the PSM polyhedron in search of new families of cutting planes and/or alternative formulations.

References

1. Cabello, S., Haverkort, H., van Kreveld, M., Speckmann, B.: Algorithmic aspects of proportional symbol maps. In: Azar, Y., Erlebach, T. (eds.) ESA 2006. LNCS, vol. 4168, pp. 720–731. Springer, Heidelberg (2006)
2. Cabello, S., Haverkort, H., van Kreveld, M., Speckmann, B.: Algorithmic aspects of proportional symbol maps. Algorithmica 58(3), 543–565 (2010)
3. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn. MIT Press, Cambridge (2001)
4. Fair Isaac Corporation. Xpress Optimizer Reference Manual (2009)
5. Dent, B.: Cartography – Thematic Map Design, 5th edn. McGraw-Hill, New York (1999)
6. Garey, M.R., Johnson, D.S.: Computers and Intractability. Freeman, San Francisco (1979)
7. Griffin, T.: The importance of visual contrast for graduated circles. Cartography 19(1), 21–30 (1990)
8. Kunigami, G., de Rezende, P.J., de Souza, C.C., Yunes, T.: Optimizing the layout of proportional symbol maps (2010), `http://www.optimization-online.org/DB_HTML/2010/11/2805.html`
9. Nemhauser, G.L., Sigismondi, G.: A strong cutting plane/branch-and-bound algorithm for node packing. Journal of the Operational Research Society 43(5), 443–457 (1992)
10. Nemhauser, G.L., Wolsey, L.A.: Integer and combinatorial optimization. Wiley-Interscience, New York (1988)
11. NOAA Satellite and Information Service. National geophysical data center (2005), `http://www.ngdc.noaa.gov`
12. Slocum, T.A., McMaster, R.B., Kessler, F.C., Howard, H.H.: Thematic Cartography and Geographic Visualization, 2nd edn. Prentice-Hall, Englewood Cliffs (2003)
13. Wein, R., Fogel, E., Zukerman, B., Halperin, D.: Advanced programming techniques applied to CGAL's arrangement package. Computational Geometry 38(1-2), 37–63 (2007), `http://www.cgal.org`