Visualizing Data as Objects by DC (Difference of Convex) Optimization

Carrizosa, Emilio; Guerrero, Vanesa; Romero Morales, Dolores

Document Version
Accepted author manuscript

Published in:
Mathematical Programming

DOI:
10.1007/s10107-017-1156-1

Publication date:
2018

License
Unspecified

Citation for published version (APA):
Carrizosa, E., Guerrero, V., & Romero Morales, D. (2018). Visualizing Data as Objects by DC (Difference of Convex) Optimization. *Mathematical Programming*, *169*(1), 119-140. https://doi.org/10.1007/s10107-017-1156-1

Visualizing Data as Objects by DC (Difference of Convex) Optimization

Emilio Carrizosa, Vanesa Guerrero, and Dolores Romero Morales

Noname manuscript No.
(will be inserted by the editor)

Visualizing data as objects by DC (difference of convex) optimization

Emilio Carrizosa · Vanesa Guerrero ·
Dolores Romero Morales

Received: date / Accepted: date

Abstract In this paper we address the problem of visualizing in a bounded region a set of individuals, which has attached a dissimilarity measure and a statistical value, as convex objects. This problem, which extends the standard Multidimensional Scaling Analysis, is written as a global optimization problem whose objective is the difference of two convex functions (DC). Suitable DC decompositions allow us to use the Difference of Convex Algorithm (DCA) in a very efficient way. Our algorithmic approach is used to visualize two real-world datasets.

Keywords Data Visualization · DC functions · DC algorithm · Multidimensional Scaling Analysis

1 Introduction

In the Big Data era, Data Visualization is an area of interest to specialists from a wide variety of disciplines, [18,19,30,31]. The information managed must be processed and, what is even more important, understood. Data Visualization techniques arise to respond to this requirement by developing specific frameworks to depict complex data structures as easy-to-interpret graphics, [46,57].

Mathematical Optimization has contributed significantly to the development of this area during recent years, see [17,34,48] and the references therein. Nowadays, complex datasets pose new challenges in order to visualize the data in such

This research is funded in part by Project MTM2015-65915-R (Spain), P11-FQM-7603 and FQM-329 (Andalucía), all with EU ERD Funds, and VPPI-US from the University of Seville.

Emilio Carrizosa
Instituto de Matemáticas Universidad de Sevilla, (Sevilla, Spain).
E-mail: ecarrizosa@us.es

Vanesa Guerrero
Instituto de Matemáticas Universidad de Sevilla, (Sevilla, Spain).
E-mail: vguerrero@us.es

Dolores Romero Morales
Copenhagen Business School (Frederiksberg, Denmark).
E-mail: drm.eco@cbs.dk

a way that patterns are captured and useful information is extracted. Special attention is paid to represent the underlying dissimilarity relationships that data may have. Although the most popular dissimilarity measures are those derived from metrics, such as the Euclidean or Mahalanobis distance, there exist also non-metric approaches, such as correlations, [14,37]. Classical dimensionality reduction techniques, such as Principal Component Analysis, [50], or Multidimensional Scaling (MDS), [33,39,59], have been customized to deal with more complex data structures, [1,5,20], and to make the interpretability of the results easier via, for instance, sparse models, [10,11,22].

Apart from adapting existing methods, specific problems may call also for new approaches. For instance, in addition to the dissimilarity measure, the data may have attached a statistical variable, to be related with the size of each object in the graphical representation of the dataset, [24]. This is the case for geographical data, to be visualized on a map in which countries are resized according to, for instance, population rates, but maintaining the neighboring relationships of countries. This type of representations, known as cartograms, [58], leads to plots in which countries are replaced by geometrical objects, frequently circles or rectangles, while the neighborhood relationships and the size of the objects are sought to be well represented. A key issue is how such problems are expressed as optimization programs, and which optimization tools are available to cope with them. For uses of optimization applied to cartograms construction and related visualization frameworks we refer the reader to [6,12,13,24,25,32,35,54,56] and references therein.

In this paper we present a new mathematical programming framework to build a visualization map, in which a set of N individuals are depicted as convex objects in a region $\Omega \subset \mathbb{R}^n$, usually $n \leq 3$. These objects must have a volume proportional to a given statistical value associated with the individuals, $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_N)$, and they should be placed accordingly to a dissimilarity measure also attached to the individuals, $\boldsymbol{\delta} = (\delta_{ij})_{i,j=1,\ldots,N}$. In order to locate the objects in Ω, a reference object $\mathcal{B}$ is used, to be translated and expanded. However, since our final goal is to obtain a visualization map which allows the analysts to understand the data they are working with, a criterion which somehow controls the appearance of the plot needs to be also considered. We will deal with this paradigm by focusing on how the objects are spread out over Ω.

Leaving aside the statistical values $\boldsymbol{\omega}$, the purpose of representing dissimilarities between individuals reminds to MDS, [5,20,22,33,39,40,44,59], which aims to represent the dissimilarity between individuals as empirical distances between points in an unbounded space of lower dimension. Although our visualization model may seem very close to MDS, it has the special feature of representing in the bounded region Ω not only dissimilarities as distances between objects, but also the statistical measure $\boldsymbol{\omega}$ through the volumes of the objects in Ω. Our visualization tool is able to rescale the dissimilarities between the individuals and the statistical values associated to them to fit into Ω. Observe that fitting the objects into Ω may yield representations in which the objects intersect if their sizes are not small enough, but, on the other hand, too small objects obstruct the visualization of the statistical measure. Ideally the objects should be spread out across the visualization region. This aim will be also taken into account when modeling the problem.

The methodology proposed in this paper has applications in fields other than Data Visualization, such as for instance, Location Analysis or Distance Geometry. In location problems, the facilities to be located are usually considered as points.

However, a natural extension is to consider facilities as dimensional structures, see [23], and difference of convex (DC) techniques have been specifically applied to this generalization, [3, 15]. Ours can also be seen as a problem in Distance Geometry optimization, as carefully reviewed in [44]. In Distance Geometry, a graph realization problem consists of finding a configuration of points such that their (Euclidean) distances fit a given dissimilarity matrix. Among them is the Sensor Network Location problem, [53, 55, 61, 65], in which one assumes that some individuals are anchors (their location is known) and the remaining ones are mobile sensors, whose location is to be obtained so that their Euclidean distances fit the dissimilarities. Thus, our method can also be applied to the Sensor Network Location problem, in which sensors and anchors have a nonnegligible area.

In this paper, the construction of a visualization map with the three characteristics mentioned above, namely the individuals are represented as geometrical objects whose volumes are proportionals to a statistical variable, which are located according to a dissimilarity measure and which are spread out in the visualization region Ω, is written as a global biobjective optimization problem with convex constraints. We show that the objective function of the aggregate problem can be expressed as a DC function, and thus DC optimization tools can be used to solve the optimization program, [43].

The rest of the paper is organized as follows. In Section 2 the biobjective optimization program to build the visualization map is formalized. In Section 3, structural properties of the optimization problem are analyzed. In Section 4, we present our algorithmic approach. Numerical results for two datasets of different size and nature are included in Section 5. Some conclusions and extensions are presented in Section 6. Finally, the Appendix closes the paper with some technical details.

2 The visualization model

Let us consider a set of N individuals, which have attached a statistical variable $\boldsymbol{\omega} \in \mathbb{R}_+^N$ and a dissimilarity matrix $\boldsymbol{\delta} = (\delta_{ij})_{\substack{i=1,\ldots,N \\ j=1,\ldots,N}}$. Let $\Omega \subset \mathbb{R}^n$ be a compact convex set and let $\mathcal{B} \subset \mathbb{R}^n$ be a compact convex set, with nonempty interior, and symmetric with respect to the origin (which belongs to $\mathcal{B}$), called reference object. Each individual i is associated with a set of the form $\boldsymbol{c}_i + \tau r_i \mathcal{B}$, where $r_i \geq 0$ is chosen so that the volume of $r_i \mathcal{B}$ is proportional to the statistical value $\omega_i \geq 0$, $\boldsymbol{c}_i \in \mathbb{R}^n$ is a translation vector and τ is a common positive rescaling for all objects. We seek the values of the variables $\boldsymbol{c}_i$, $i = 1, \ldots, N$, and τ so that objects $\boldsymbol{c}_i + \tau r_i \mathcal{B}$ are contained in Ω. Figure 1 illustrates the previously described representation.

Hereafter, we deal with a biobjective optimization problem: the distances between the objects representing the individuals i and j must resemble the dissimilarities δ_{ij} between such individuals, and the objects must be spread out in Ω to make the visualization easier. The two criteria are formalized in what follows.

2.1 First objective: distances resemble dissimilarities

Regarding the first objective, a function d, which gives us a strictly positive distance between two non-intersecting objects representing individuals i and j and

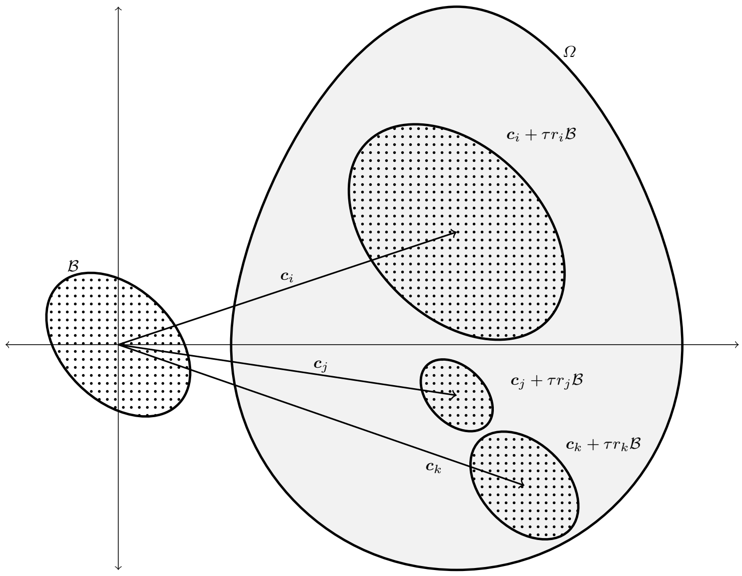


Fig. 1 Example in $\mathbb{R}^2$ of a visualization region Ω, a reference object $\mathcal{B}$ and three individuals i, j and k defined through the translation vectors $\boldsymbol{c}_i$, $\boldsymbol{c}_j$ and $\boldsymbol{c}_k$, which are scaled via τr_i, τr_j and τr_k.

zero otherwise, needs to be considered. Thus, we define the function g_{ij}, which assigns such distance to two individuals i and j, as follows

$$\begin{array}{rcl} g_{ij} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^+ & \longrightarrow & \mathbb{R}^+ \\ (\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) & \longmapsto & d(\boldsymbol{c}_i + \tau r_i \mathcal{B}, \boldsymbol{c}_j + \tau r_j \mathcal{B}). \end{array} \tag{1}$$

Then, to quantify the resemblance between the distances in the visualization map and the dissimilarities, the summation over all the individuals of the squared differences between the distances and the rescaled dissimilarities through a positive variable κ needs to be minimized. Thus, we consider as first objective the function F_1 defined as

$$\begin{array}{rcl} F_1 : \mathbb{R}^n \times \ldots \times \mathbb{R}^n \times \mathbb{R}^+ \times \mathbb{R}^+ & \longrightarrow & \mathbb{R}^+ \\ (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N, \tau, \kappa) & \longmapsto & \displaystyle\sum_{\substack{i,j=1,\ldots,N \\ i \neq j}} \left[g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) - \kappa \delta_{ij} \right]^2 . \end{array}$$

Observe that for simplicity all pairs (i, j) are considered in the summation in F_1, but our analysis remains valid if only some pairs (i, j) are taken into consideration, as done e.g. in [60].

2.2 Second objective: spread

To avoid that the objects collapse in a small subregion of Ω, we encourage objects to be spread out all over Ω. There are several ways to model spread. For instance, we could use the overall volume covered by the objects, the amount of intersections between them, or the distances between the objects. This last option is the one analyzed in detail in this paper, and therefore, our aim is to maximize the sum over all the individuals of the distances between the objects representing them. Let F_2 be a function which, given the translation vectors, $\boldsymbol{c}_i$, and the rescaling parameter, τ, computes the spread of the visualization map in such way. Then, written in minimization form, one has

$$\begin{aligned} F_2 : \mathbb{R}^n \times \ldots \times \mathbb{R}^n \times \mathbb{R}^+ &\longrightarrow \mathbb{R}^+ \\ (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N, \tau) &\longmapsto - \sum_{\substack{i,j=1,\ldots,N \\ i \neq j}} g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau). \end{aligned} \tag{2}$$

Note that F_2 does not distinguish between how much the objects intersect, since it penalizes in the same way two objects one on top of the other and two tangent objects. A possible way to quantify the amount of intersection between two objects is by measuring the minimum-norm translation of such objects which makes them not to intersect. This leads to the concept of penetration depth, [27, 49, 63].

Let $\|\cdot\|$ be a norm, A_1 and A_2 two convex compact sets with nonempty interior, and $\boldsymbol{p}$ a direction in $\mathbb{R}^n$. Let $int(\cdot)$ be the interior of a set and '+' the translator operator. The penetration depth of A_1 and A_2 is defined as the minimum norm vector $\boldsymbol{p} \in \mathbb{R}^n$ such that A_1 translated through the direction $\boldsymbol{p}$, namely $\boldsymbol{p} + A_1$, is disjoint with $int(A_2)$, i.e.

$$\pi(A_1, A_2) = \min_{\boldsymbol{p} \in \mathbb{R}^n} \left\{ \|\boldsymbol{p}\| : \, (\boldsymbol{p} + A_1) \cap int(A_2) = \emptyset \right\}.$$

Computing the penetration depth between two sets is costly, in general. Nevertheless, exact and heuristic algorithms exist for specific types of objects such as discs and convex polytopes in $\mathbb{R}^2$ and $\mathbb{R}^3$, [7, 45].

The amount of intersection between the objects in the visualization map can be quantified as the sum over all the individuals of the squared penetration depth between pairs of them, yielding the function F_2^{Π} defined as

$$\begin{aligned} F_2^{\Pi} : \mathbb{R}^n \times \ldots \times \mathbb{R}^n \times \mathbb{R}^+ &\longrightarrow \mathbb{R}^+ \\ (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N, \tau) &\longmapsto \sum_{\substack{i,j=1,\ldots,N \\ i \neq j}} \pi^2 \left(\boldsymbol{c}_i + \tau r_i \mathcal{B}, \boldsymbol{c}_j + \tau r_j \mathcal{B} \right). \end{aligned} \tag{3}$$

However, the penetration depth does not measure how separated the objects are. Then, an alternative to the two previous spread criteria, namely F_2 and F_2^{Π}, which does take into account both the amount of intersection and the separation of the objects, consists of measuring the distance between the centers of the objects. Maximizing the sum over all the individuals of the squared distances between the centers gives an alternative spread criterion, namely

$$\begin{aligned} F_2^{\boldsymbol{c}} : \mathbb{R}^n \times \ldots \times \mathbb{R}^n &\longrightarrow \mathbb{R}^+ \\ (\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N) &\longmapsto - \sum_{\substack{i,j=1,\ldots,N \\ i \neq j}} \|\boldsymbol{c}_i - \boldsymbol{c}_j\|^2. \end{aligned} \tag{4}$$

2.3 Problem statement

The problem of building a visualization map in which a set of convex objects in the form $\boldsymbol{c}_i + \tau r_i \mathcal{B}$ are represented in the region Ω, satisfying that the distances between the objects resemble the dissimilarities between the individuals and the map is spread enough, can be stated as a biobjective optimization problem. By proceeding in the usual way, we consider the convex combination of the objectives and solve the aggregate problem, see [26]. Thus, given $\lambda \in [0,1]$, the Visualization Map problem, $(VM)^*$, is stated as follows

$$\begin{array}{lll} \min\limits_{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_N,\tau,\kappa} & \lambda F_1(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_N,\tau,\kappa) + (1-\lambda)F_2^*(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_N,\tau) & \\ \text{s.t.} & \boldsymbol{c}_i + \tau r_i \mathcal{B} \subseteq \Omega, \ i = 1,\ldots,N & (VM)^* \\ & \tau \in T & \\ & \kappa \in K, & \end{array}$$

where $K, T \subset \mathbb{R}^+$ and F_2^* refers to either F_2, F_2^{Π} or $F_2^{\boldsymbol{c}}$ stated in (2), (3) and (4), yielding the models (VM), $(VM)^{\Pi}$ or $(VM)^{\boldsymbol{c}}$, respectively.

3 Properties

In this section we study the structure of problem $(VM)^*$ for the three different choices proposed as the spread criterion, namely the three possibilities for F_2^* outlined in Section 2.2 and given in (2), (3) and (4). For each choice, we will prove that their objective functions are DC, by considering distance functions d, defined in the space of compact convex sets of $\mathbb{R}^n$, which satisfy the following:

Assumption 1 *The function d, defined on pairs of compact convex sets of $\mathbb{R}^n$, satisfies for any A_1 and A_2*

(i) $d \geq 0$ *and* d *is symmetric*
(ii) $d(A_1, A_2) = d(A_1 + z, A_2 + z)$, $\forall z \in \mathbb{R}^n$
(iii) The function $d_z : z \in \mathbb{R}^n \longmapsto d(z + A_1, A_2)$ *is convex and satisfies for all* $\theta > 0$ *that* $d_z(\theta A_1, \theta A_2) = \theta d_{\frac{1}{\theta}z}(A_1, A_2)$.

Typical instances of d satisfying (i)-(iii) are

1. The infimum distance, defined as
$$d(A_1, A_2) = \inf\{\|a_1 - a_2\| : a_1 \in A_1, a_2 \in A_2\}. \quad (d1)$$
2. The supremum distance, defined as
$$d(A_1, A_2) = \sup\{\|a_1 - a_2\| : a_1 \in A_1, a_2 \in A_2\}. \quad (d2)$$
3. The average distance, defined as
$$d(A_1, A_2) = \int \|a_1 - a_2\| d\mu(a_1) d\nu(a_2), \quad (d3)$$
where μ, ν are probability distributions in $\mathbb{R}^n$ with support A_1 and A_2, [8,15,16,38].

4. The Hausdorff distance, defined as

$$d(A_1, A_2) = \max\left\{ \sup_{a_1 \in A_1} \inf_{a_2 \in A_2} \|a_1 - a_2\|, \sup_{a_2 \in A_2} \inf_{a_1 \in A_1} \|a_1 - a_2\| \right\}. \qquad (d4)$$

Checking that, indeed, $(d1)--(d4)$ above fulfill $(i)--(iii)$ is straightforward from the definition of the functions $(d1)--(d4)$ and the algebra of convex functions. Observe that, thanks to Assumption 1, the distance between two objects representing individuals i and j, given by the function g_{ij} in (1), can be expressed as

$$g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = \tau d_{\frac{1}{\tau}(\boldsymbol{c}_i - \boldsymbol{c}_j)}(r_i \mathcal{B}, r_j \mathcal{B}), \qquad (5)$$

and thus g_{ij} is the perspective of the convex function $f_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j) = d_{\boldsymbol{c}_i - \boldsymbol{c}_j}(r_i \mathcal{B}, r_j \mathcal{B})$. Hence, g_{ij} is convex as well, see e.g. [36] for the proof.

Elementary tools of DC optimization enable us to show that objective function in (VM), namely $\lambda F_1 + (1-\lambda) F_2$, is DC, and a DC decomposition can be given. The result is presented in Proposition 1 and the proof is included in the Appendix for the sake of completeness.

Proposition 1 *The function $\lambda F_1 + (1-\lambda)F_2$ is DC, $\lambda \in [0,1]$, and a decomposition is given by*

$$\lambda F_1 + (1-\lambda) F_2 = u - (u - \lambda F_1 - (1-\lambda) F_2),$$

where

$$u = \sum_{\substack{i,j=1,\ldots,N \\ i \neq j}} \left\{ \max\{3\lambda - 1, 0\} g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) + 2\lambda (\kappa \delta_{ij})^2 \right\}.$$

In the same vein, we can prove that the objective functions in $(VM)^{\Pi}$ and $(VM)^{\boldsymbol{c}}$ are DC as well, as stated in the following results.

Proposition 2 *Let h_{ij} be defined as the penetration depth between $\boldsymbol{c}_i + \tau r_i \mathcal{B}$ and $\boldsymbol{c}_j + \tau r_j \mathcal{B}$, namely*

$$\begin{aligned} h_{ij} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^+ &\longrightarrow \mathbb{R}^+ \\ (\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) &\longmapsto \pi\left(\boldsymbol{c}_i + \tau r_i \mathcal{B}, \boldsymbol{c}_j + \tau r_j \mathcal{B}\right). \end{aligned}$$

Denoting as $\sigma_{\mathcal{B}}$ the support function of $\mathcal{B}$, one has that h_{ij} is DC, and a decomposition is given by $h_{ij} = u_{ij} - (u_{ij} - h_{ij})$, where

$$u_{ij} = \max\left\{ \max_{\substack{\boldsymbol{\xi} \in \mathbb{R}^n \\ \|\boldsymbol{\xi}\| = 1}} \left\{ \boldsymbol{\xi}^\top (\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau (r_i + r_j) \sigma_{\mathcal{B}}(\boldsymbol{\xi}) \right\}, 0 \right\}.$$

Proof See Appendix. □

Corollary 1 *The function $\lambda F_1 + (1-\lambda) F_2^{\Pi}$ is DC, $\lambda \in [0,1]$.*

Proof The function F_1 is DC. Indeed, it is sufficient to take $\lambda = 1$ in Proposition 1. F_2^{Π} is also DC by using Proposition 2 and Proposition 3.7 in [62]. Then, since the summation of DC functions is also DC, the result holds. □

Corollary 2 *The function* $\lambda F_1 + (1-\lambda)F_2^{\mathbf{c}}$ *is DC,* $\lambda \in [0,1]$.

Proof Since the function F_1 is DC (take $\lambda = 1$ in Proposition 1) and $F_2^{\mathbf{c}}$ is concave, since it is minus the summation of squares of a nonnegative convex function, the result holds. □

Thus, DC decompositions for objective functions in $(VM)^{\Pi}$ and $(VM)^{\mathbf{c}}$ are readily available from the DC decomposition of F_1 in Proposition 1 ($\lambda = 1$), Proposition 2 and the concavity of $F_2^{\mathbf{c}}$.

Showing that a function is DC and giving explicitly a DC decomposition enables us to use DC optimization algorithms. It is well known that the performance of the procedures may strongly depend on the choice of the DC decomposition, [2,4,28]. Particularly, we seek a DC decomposition involving a quadratic convex separable function as those addressed in [42,52] for the special case where (*d1*) is used. We will show in Section 4 that such alternative decomposition yields a simple heuristic inspired by DC algorithm (DCA).

The following result, Proposition 3, states a different DC decomposition for the objective function in (VM) from the one given in Proposition 1, when the infimum distance given in (*d1*) is considered. In fact, this alternative decomposition involves a quadratic convex separable function.

Proposition 3 *The function* $\lambda F_1 + (1-\lambda)F_2$*, where* d *is the infimum distance (d1), can be expressed as a DC function,* $\lambda F_1 + (1-\lambda)F_2 = u - (u - \lambda F_1 - (1-\lambda)F_2)$*, where the quadratic separable convex function* u *is given by*

$$u = \max\{3\lambda-1, 0\} \cdot \left[\sum_{i=1,\dots,N} \left\{ 8(N-1)\|\boldsymbol{c}_i\|^2 \right\} + \tau^2 \sum_{\substack{i,j=1,\dots,N \\ i \neq j}} \beta_{ij} \right] + 2\lambda\kappa^2 \sum_{\substack{i,j=1,\dots,N \\ i \neq j}} \delta_{ij}^2,$$

where β_{ij} *satisfies* $\beta_{ij} \geq 2\|r_i b_i - r_j b_j\|^2$ *for all* $b_i, b_j \in \mathcal{B}$.

Proof See Appendix. □

Section 4 is mainly devoted to show how problem (VM) with the decomposition given in Proposition 3 can be efficiently solved by means of a heuristic inspired in DCA.

4 The algorithmic approach

Propositions 1-3 and Corollaries 1-2 show that $(VM)^*$ is an optimization problem with a DC objective function with a DC decomposition available and simple constraints. Then, DC optimization tools can be used, either of exact nature for very low dimensional problems, [2,3], or heuristics, as the DCA, [41,43,51]. The latter is the approach we are following in this paper, and we refer the reader to [21,40] for alternative mathematical optimization approaches to MDS.

Roughly speaking, DCA consists of an iterative process in which a sequence of convex programs are solved. Given a DC program of the form $\min\{f(x) = u(x) - v(x) : x \in X\}$, where X is a convex feasible region, at each iteration, the concave part $(-v(x))$ is replaced by its affine majorization at a certain $x_0 \in X$, and the resulting convex problem is then solved. Let $\langle \cdot, \cdot \rangle$ be an inner product in $\mathbb{R}^n$, and let $\partial v(x_0)$ the subdifferential of v at x_0. A general scheme of DCA is outlined in Algorithm 1.

Algorithm 1 DCA scheme, [43]

Input: $x_0 \in X$.
1: $t \leftarrow 0$
2: **repeat**
3: Compute some $y_t \in \partial v(x_t)$;
4: Compute $x_{t+1} \in \arg\min \{u(x) - (v(x_t) + \langle x - x_t, y_t \rangle) : x \in X\}$;
5: $t \leftarrow t + 1$;
6: **until** stop condition is met.
Output: x_t

However, running times would be dramatically reduced if a DC decomposition of the objective function were available so that the convex optimization problems to be solved in line 4 of Algorithm 1 were trivial, in the sense that an explicit expression for the optimal solution is available. This idea has been studied in [42, 52] and it will be customized to problem (VM), considering the infimum distance given in $(d1)$, in what follows.

When the DCA scheme is applied to problem (VM) with the DC decomposition given in Proposition 3, we see that the convex subproblems to be solved at line 4 of Algorithm 1 have the form

$$\begin{array}{ll} \min\limits_{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_N,\tau,\kappa} & \left\{ \sum\limits_{i=1,\ldots,N} \left\{ M^{\boldsymbol{c}_i} \|\boldsymbol{c}_i\|^2 \right\} + M^{\kappa}\kappa^2 + M^{\tau}\tau^2 + \sum\limits_{i=1,\ldots,N} \left\{ {\boldsymbol{c}_i}^{\top} \boldsymbol{q}^{\boldsymbol{c}_i} \right\} + p^{\kappa}\kappa + p^{\tau}\tau \right\} \\ \text{s.t.} & \boldsymbol{c}_i + \tau r_i \mathcal{B} \subseteq \Omega,\ i = 1, \ldots, N \\ & \tau \in T \\ & \kappa \in K, \end{array}$$

for scalars $M^{\boldsymbol{c}_i}$, M^{κ}, $M^{\tau} \in \mathbb{R}^+_{,}$ which come from the coefficients that multiply each term in the u part of the DC decomposition given in Proposition 3, and vectors $\boldsymbol{q}^{\boldsymbol{c}_i} \in \mathbb{R}^n$ and scalars p^{κ} and $p^{\tau} \in \mathbb{R}$, which come for the computation of the subgradients at a given point of the $u - \lambda F_1 - (1-\lambda)F_2$ part.

Such problem can be written as a summation of two separate problems,

$$\min_{\kappa \in K} \left\{ M^{\kappa}\kappa^2 + p^{\kappa}\kappa \right\} + \min_{\substack{\boldsymbol{c}_i + \tau r_i \mathcal{B} \subseteq \Omega \\ \tau \in T}} \left\{ \sum_{i=1,\ldots,N} \left\{ M^{\boldsymbol{c}_i} \|\boldsymbol{c}_i\|^2 + {\boldsymbol{c}_i}^{\top} \boldsymbol{q}^{\boldsymbol{c}_i} \right\} + M^{\tau}\tau^2 + p^{\tau}\tau \right\}. \quad (6)$$

The first problem in (6) is a quadratic problem in one variable, for which a closed form can be given for its optimal value. The second problem in (6) is separable in the variables $\boldsymbol{c}_i$ if the linking variable τ were fixed at τ_0. For this reason, an alternating strategy seems to be plausible, in which one alternates the optimization of τ for $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N$ fixed (and this is a one-dimensional quadratic problem and thus a closed formula for the optimal solution is readily obtained), and then for τ fixed, the centers $\boldsymbol{c}_i$ are to be optimized. This is done by solving separately N optimization problems of the form

$$\begin{aligned} &\min_{\boldsymbol{c}_i} \left\{ M^{\boldsymbol{c}_i} \|\boldsymbol{c}_i\|^2 + {\boldsymbol{c}_i}^\top \boldsymbol{q}^{\boldsymbol{c}_i} \right\} \\ &\text{s.t.} \;\; \boldsymbol{c}_i \in \Omega - \tau r_i \mathcal{B}. \end{aligned} \tag{7}$$

In order to solve problem (VM), we propose an alternating procedure which integrates a DCA strategy, to obtain $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N$ as stated in Algorithm 1, into an outer loop to get τ and κ. The alternating scheme to solve (VM) is stated in Algorithm 2: lines 3–8 contain the DCA as outlined in Algorithm 1 to find $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N$, which is embedded in a main loop to get κ and τ (lines 1–14). We point out that line 6 in Algorithm 2 contains the convex optimization problems to be solved in each iteration of DCA (line 4 of Algorithm 1), and explicit expressions for the optimal solution of the optimization problems in lines 10 and 12 are known. Therefore, if the optimal solution of the optimization problems in line 6 of Algorithm 2 could be optimally computed without calling any external numerical optimization routine, then each iteration of the inner DCA in lines 3–8 would be computationally cheap.

Algorithm 2 Alternating scheme for (VM)

Input: $\boldsymbol{c}_1^0, \ldots, \boldsymbol{c}_N^0 \in \Omega$, $\kappa^0 \in K$, $\tau^0 \in T$.
1: $s \leftarrow 0$;
2: **repeat**
3: $t \leftarrow 0$;
4: **repeat**
5: Compute $M^{\boldsymbol{c}_i{}^t}$ and $\boldsymbol{q}^{\boldsymbol{c}_i{}^t}$, $i = 1, \ldots, N$;
6: Compute $\boldsymbol{c}_1^{t+1}, \ldots, \boldsymbol{c}_N^{t+1}$ by solving (7) for τ fixed at τ^s;
7: $t \leftarrow t + 1$;
8: **until** stop condition is met.
9: Compute M^{κ^s} and p^{κ^s};
10: Compute κ^{s+1} by solving the first optimization problem in (6);
11: Compute M^{τ^s} and p^{τ^s};
12: Compute τ^{s+1} by solving the second optimization problem in (6) for $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N$ fixed at $\boldsymbol{c}_1^t, \ldots, \boldsymbol{c}_N^t$;
13: $s \leftarrow s + 1$;
14: **until** stop condition is met.
Output: $\boldsymbol{c}_1^t, \ldots, \boldsymbol{c}_N^t, \kappa^t, \tau^s$

Two particular cases of (7) have an amenable structure, yielding a closed formula for the optimal solution, and thus avoiding any call to external numerical optimization routines in line 7 of Algorithm 2. Indeed, suppose Ω is a rectangle, for simplicity taken as $[0, 1]^n$, τ is fixed to a real positive value τ_0, and $\mathcal{B}$ is the disc centered at the origin with radius r_0. Then, the constraint in (7) can be rewritten as

$$\tau_0 r_0 r_i \leq c_{ij} \leq 1 - \tau_0 r_0 r_i, \;\; j = 1, \ldots, n,$$

and thus (7) is expressed as

$$\sum_{j=1,\ldots,n} \min_{c_{ij}} \left\{ M^{\boldsymbol{c}_i} c_{ij}^2 + \boldsymbol{q}_j^{\boldsymbol{c}_i} c_{ij} : \ \tau_0 r_0 r_i \leq c_{ij} \leq 1 - \tau_0 r_0 r_i \right\} \tag{8}$$

In other words, (7) is decomposed into n one dimensional quadratic problems on an interval, and thus a closed formula is readily obtained for the optimal solution of each problem of the form (8), and thus also for (7).

Similarly, suppose Ω and $\mathcal{B}$ are discs centered at the origin of radius 1 and r_0 respectively. Then, (7) is rewritten as

$$\begin{aligned} &\min_{\boldsymbol{c}_i} \left\{ M^{\boldsymbol{c}_i} \|\boldsymbol{c}_i\|^2 + \boldsymbol{c}_i{}^\top \boldsymbol{q}^{\boldsymbol{c}_i} \right\} \\ &\text{s.t.} \ \ \|\boldsymbol{c}_i\| \leq 1 - \tau_0 r_0 r_i. \end{aligned} \tag{9}$$

Karush-Kuhn-Tucker conditions immediately yield an expression for the optimal solution of (9).

Summarizing, while the alternating strategy stated in Algorithm 2, which contains a DCA scheme, could be applied to solve $(VM)^*$ for an arbitrary DC decomposition of the objective function, we see that the DC decomposition given in Proposition 3 for (VM) is particularly attractive. We have shown that some convenient choices of Ω (a rectangle or a disc) and $\mathcal{B}$ (a disc) yield a closed formula for the optimal solution of the subproblems to be addressed at each stage of the inner DCA, thus avoiding the need of using numerical optimization routines. See [42] for other problems in which this strategy has been successful.

5 Numerical illustrations

The methodology proposed in Section 4 is illustrated using two real-world datasets of diverse nature, for which both the statistical values and the dissimilarities are readily available in the literature. In particular, the dissimilarity measures come from a correlation matrix and a shortest paths matrix in a directed graph, whereas the statistical variables represent a proportion (continuous variable) and the outdegree of a set of nodes (discrete variable), respectively.

The first dataset consists of $N = 11$ financial markets across Europe and Asia. The statistical value ω_i relates to the importance of market i relative to the world market portfolio, [29], and the dissimilarity δ_{ij} is based on the correlation between markets i and j, [5]. The second dataset is a social network of $N = 200$ musicians, modeled as a graph, where there is an arc connecting two nodes if one musician was influential on the other, [24]. The statistical value ω_i represents the outdegree of node i and the dissimilarity between musicians i and j is based on the shortest distance from node i to j, [24].

Algorithm 2 has been coded in C and the experiments have been carried out in a Windows 8.1 PC Intel® Core™ i7-4500U, 16GB of RAM. We set $\lambda = 0.9$ and $\mathcal{B}$ equal to the circle centered at $(0, 0)$ with radius equal to one. Since (VM) is a multimodal problem and the our algorithm may get stuck at a local optimum, 100 runs of a multistart are executed. Initial values for $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N$ are uniformly generated in Ω, whereas the initial values for κ and τ are chosen as the midpoint of intervals K and T, respectively. At each run of the multistart procedure, index s in Algorithm 2 takes a maximum value of 3 and index t a maximum value of 50.

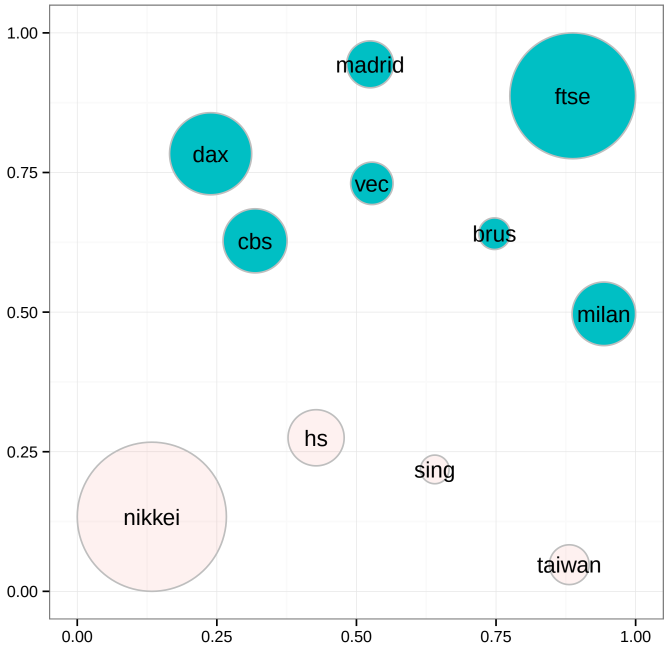


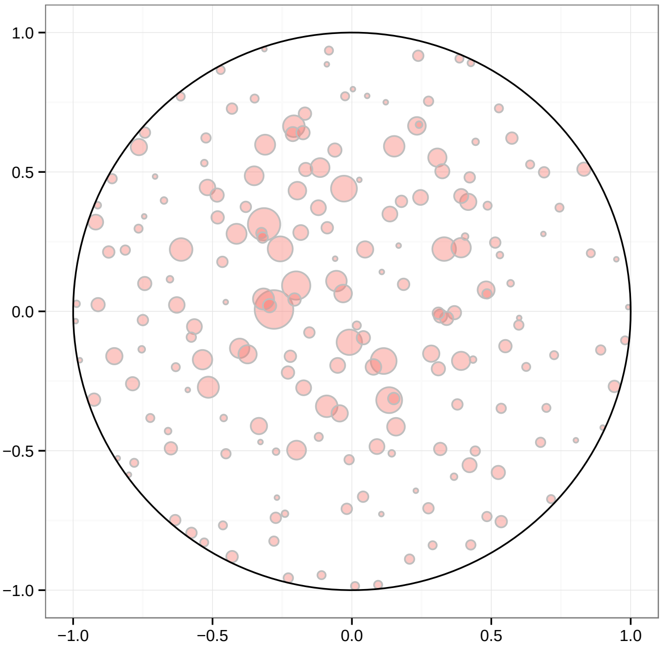
Fig. 2 Visualizing financial markets

Figure 2 plots the financial markets dataset on the visualization region $\Omega = [0,1]\times[0,1]$, with the scaling parameters ranging in the intervals $K = T = [0.4, 0.6]$. Observe that, the European markets are clustered above the Asian ones, covering the upper half rectangle. These two clusters are represented with different colours. Figure 3 plots the musicians' social network taking a circular visualization region, namely $\Omega = \mathcal{B}$, with the scaling parameters ranging in the intervals $K = [0.075, 0.100]$ and $T = [0.015, 0.030]$, respectively. In the plot at the top, we find all musicians. In the plot at the bottom, we have highlighted one of the most influential nodes, the Rolling Stones, and the connected nodes: musicians influencing the Rolling Stones (respectively, those influenced by them) can be found in a lighter (respectively darker) colour.

6 Concluding remarks and extensions

In this paper we have addressed the problem of representing, in a so-called visualization region Ω, a set of individuals by means of convex objects so that the distance between the objects fits as close as possible a given dissimilarity matrix, the volume of the objects represents a statistical variable, and, at the same time, the spread of the objects within Ω is maximized.

The problem has been formulated as a DC optimization problem, and a heuristic inspired in the DCA has been proposed as solution approach. For particular choices of the visualization region Ω (a rectangle and a disc), the reference ob-



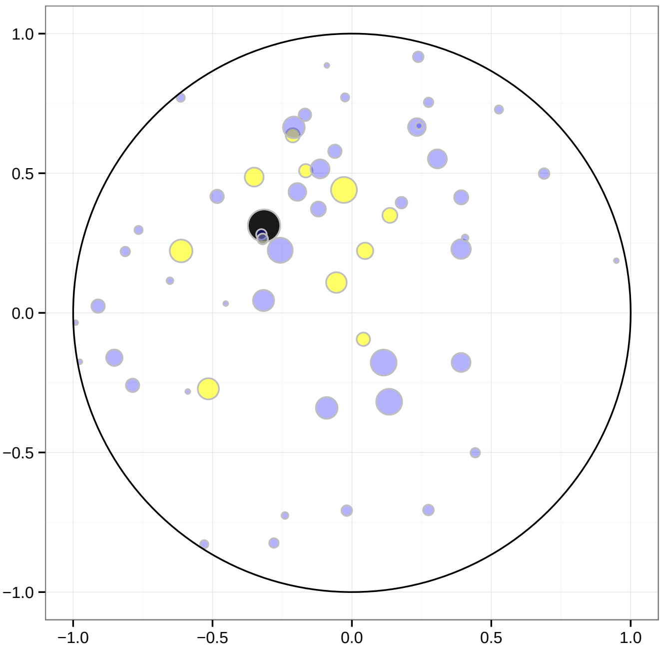


Fig. 3 Visualizing the musicians' social network

ject (a disc) and the function d (the infimum distance), closed formulas for the optimal solutions of the DCA subproblems are obtained, thus avoiding the need to use numerical optimization routines. The examples presented demonstrate the usefulness of our approach.

Several extensions deserve further analysis.

In the algorithmic section, we have considered the infimum distance (*d1*). Instead, one can consider other classical distances in Cluster Analysis such as the supremum distance (*d2*) or the average one (*d3*), [34]. It should be observed that the average distance between two convex sets may not have an easy expression, and thus approximations may be needed, [38,64].

We have assumed the reference object $\mathcal{B}$ to be convex, to guarantee the convexity of the function giving the infimum distance and thus allowing us to express $(VM)^*$ as a DC optimization problem. For arbitrary sets $\mathcal{B}$ the infimum distance (*d1*) and the Hausdorff distance (*d4*) functions may not be DC, see [2]. However, as discussed e.g. in [3], important classes of nonconvex sets (e.g. finite union of convex sets) make the infimum distance a DC function, and thus the analysis in this paper extends gracefully to such cases. It should be observed that if the supremum distance or the average distance are used instead, then the distance function is convex for arbitrary reference objects. This follows from the fact that the supremum in (*d2*) and the integral in (*d3*) preserve the convexity of the norm function used in their definitions. Thus, the objective function of $(VM)^*$ would be DC regardless of the shape of $\mathcal{B}$.

Another promising extension to be modeled is the case in which objects have associated not a dissimilarity $\boldsymbol{\delta}$, but a time series of dissimilarities $\{\boldsymbol{\delta}^l : l = 1, \ldots, L\}$. In this case, we seek each individual to be represented at each time instant $l = 1, \ldots, L$ by an object so that distances between objects are as close as possible to those in $\boldsymbol{\delta}^l$, but, at the same time, smooth transitions take place between the representation at time l and $l + 1$, $l = 1, \ldots, L - 1$. The approach developed in this paper can be adapted to include such smoothness criterion too.

Regarding the optimization, we have proposed DCA as a plausible approach, which can quickly handle problems of non-negligible size since, for convenient choices of Ω and $\mathcal{B}$, (costly) numerical routines are not needed to solve the subproblems at each stage of the DCA. Convergence of Algorithm 2 to the global optimum is not guaranteed. A better performance can be obtained if instead of a uniform multistart, a more guided strategy is used, or if DCA is plugged, as a local search routine, within a strategy which avoids local optima, such as (continuous) Variable Neighborhood Search, [9,47]. This extension, together with a detailed study of convergence of Algorithm 2 using the convergence results of DCA [41, 43,51], calls for further analysis and testing.

Acknowledgements We thank the reviewers for their helpful suggestions and comments, which have been very valuable to strengthen the paper and to improve its quality.

References

1. H. Abdi, L. J. Williams, D. Valentin, and M. Bennani-Dosse. STATIS and DISTATIS: optimum multitable principal component analysis and three way metric multidimensional scaling. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(2):124–167, 2012.

2. R. Blanquero and E. Carrizosa. Continuous location problems and big triangle small triangle: constructing better bounds. *Journal of Global Optimization*, 45(3):389–402, 2009.
3. R. Blanquero, E. Carrizosa, and P. Hansen. Locating objects in the plane using global optimization techniques. *Mathematics of Operations Research*, 34(4):837–858, 2009.
4. I. M. Bomze, M. Locatelli, and F. Tardella. New and old bounds for standard quadratic optimization: dominance, equivalence and incomparability. *Mathematical Programming*, 115(1):31–64, 2008.
5. I. Borg and P.J.F. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2005.
6. K. Buchin, B. Speckmann, and S. Verdonschot. Evolution strategies for optimizing rectangular cartograms. In N. Xiao, M.-P. Kwan, M.F. Goodchild, and S. Shekhar, editors, *Geographic Information Science*, volume 7478 of *Lecture Notes in Computer Science*, pages 29–42. Springer, 2012.
7. S. Cameron and R. Culley. Determining the minimum translational distance between two convex polyhedra. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 591–596, 1986.
8. E. Carrizosa, E. Conde, M. Muñnoz-Márquez, and J Puerto. The generalized weber problem with expected distances. *Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle*, 29(1):35–57, 1995.
9. E. Carrizosa, M. Dražić, Z. Dražić, and N. Mladenović. Gaussian variable neighborhood search for continuous optimization. *Computers & Operations Research*, 39(9):2206–2213, 2012.
10. E. Carrizosa and V. Guerrero. Biobjective sparse principal component analysis. *Journal of Multivariate Analysis*, 132:151–159, 2014.
11. E. Carrizosa and V. Guerrero. rs-Sparse principal component analysis: A mixed integer nonlinear programming approach with VNS. *Computers & Operations Research*, 52:349–354, 2014.
12. E. Carrizosa, V. Guerrero, and D. Romero Morales. A multi-objective approach to visualize adjacencies in weighted graphs by rectangular maps. Technical report, Optimization Online, 2015. http://www.optimization-online.org/DB_HTML/2015/12/5226.html.
13. E. Carrizosa, V. Guerrero, and D. Romero Morales. Visualizing proportions and dissimilarities by space-filling maps: a large neighborhood search approach. *Computers & Operations Research*, 78:369–380, 2017.
14. E. Carrizosa, B. Martín-Barragán, F. Plastria, and D. Romero Morales. On the selection of the globally optimal prototype subset for nearest-neighbor classification. *INFORMS Journal on Computing*, 19(3):470–479, 2007.
15. E. Carrizosa, M. Muñoz-Márquez, and J. Puerto. Location and shape of a rectangular facility in $\mathbb{R}^n$. Convexity properties. *Mathematical Programming*, 83(1-3):277–290, 1998.
16. E. Carrizosa, M. Muñoz-Márquez, and J. Puerto. The weber problem with regional demand. *European Journal of Operational Research*, 104(2):358–365, 1998.
17. E. Carrizosa and D. Romero Morales. Supervised classification and mathematical optimization. *Computers & Operations Research*, 40(1):150–165, 2013.
18. C. P. Chen and C.-Y. Zhang. Data-intensive applications, challenges, techniques and technologies: A survey on big data. *Information Sciences*, 275:314–347, 2014.
19. J. Choo and H. Park. Customizing computational methods for visual analytics with big data. *IEEE Computer Graphics and Applications*, 33(4):22–28, 2013.
20. T. F. Cox and M. A. A. Cox. *Multidimensional scaling*. CRC Press, 2000.
21. J. De Leeuw and W.J. Heiser. Convergence of correction matrix algorithms for multidimensional scaling. In J.C. Lingoes, E.E. Roskam, and I. Borg, editors, *Geometric Representations of Relational Data*, pages 735–752. Mathesis Press, Ann Arbor, MI, 1977.
22. V. De Silva and J. B. Tenenbaum. Sparse multidimensional scaling using landmark points. Technical report, Stanford University, 2004.
23. J.M. Díaz-Báñez, J. A. Mesa, and A. Schöbel. Continuous location of dimensional structures. *European Journal of Operational Research*, 152(1):22–44, 2004.
24. M. Dörk, S. Carpendale, and C. Williamson. Visualizing explicit and implicit relations of complex information spaces. *Information Visualization*, 11(1):5–21, 2012.
25. D. Dorling. Area cartograms: their use and creation. In *Concepts and Techniques in Modern Geography series no. 59*. University of East Anglia: Environmental Publications, 1996.
26. M. Ehrgott. A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research*, 147(1):343–360, 2006.

27. A. Elkeran. A new approach for sheet nesting problem using guided cuckoo search and pairwise clustering. *European Journal of Operational Research*, 231(3):757–769, 2013.
28. A. Ferrer and J. E. Martínez-Legaz. Improving the efficiency of DC global optimization methods by improving the DC representation of the objective function. *Journal of Global Optimization*, 43(4):513–531, 2009.
29. T. Flavin, M. Hurley, and F. Rousseau. Explaining stock market correlation: A gravity model approach. *The Manchester School*, 70:87–106, 2002.
30. K. Fountoulakis and J. Gondzio. Performance of first- and second-order methods for ℓ_1-regularized least squares problems. *Computational Optimization and Applications*, 65(3):605–635, 2016.
31. K. Fountoulakis and J. Gondzio. A second-order method for strongly convex ℓ_1-regularization problems. *Mathematical Programming*, 156(1):189–219, 2016.
32. E. Gomez-Nieto, F. San Roman, P. Pagliosa, W. Casaca, E. S. Helou, M. C. F. de Oliveira, and L. G. Nonato. Similarity preserving snippet-based visualization of web search results. *IEEE Transactions on Visualization and Computer Graphics*, 20(3):457–470, 2014.
33. J. C. Gower. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53(3-4):325–338, 1966.
34. P. Hansen and B. Jaumard. Cluster analysis and mathematical programming. *Mathematical Programming*, 79(1-3):191–215, 1997.
35. R. Heilmann, D. A. Keim, C. Panse, and M. Sips. Recmap: Rectangular map approximations. In *Proceedings of the IEEE Symposium on Information Visualization*, pages 33–40. IEEE Computer Society, 2004.
36. J.B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms*. Springer, 1993.
37. L. Kaufman and P. J. Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Wiley, New York.
38. T. Koshizuka and O. Kurita. Approximate formulas of average distances associated with regions and their applications to location problems. *Mathematical Programming*, 52(1-3):99–123, 1991.
39. J. B. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.
40. H. A. Le Thi and T. Pham Dinh. D.C. Programming Approach to the Multidimensional Scaling Problem. In A. Migdalas, P.M. Pardalos, and P. Värbrand, editors, *From Local to Global Optimization*, volume 53 of *Nonconvex Optimizations and Its Applications*, pages 231–276. Springer, 2001.
41. H. A. Le Thi and T. Pham Dinh. DC programming approaches for distance geometry problems. In A. Mucherino, C. Lavor, L. Liberti, and N. Maculan, editors, *Distance Geometry*, pages 225–290. Springer, 2013.
42. H.A. Le Thi. An efficient algorithm for globally minimizing a quadratic function under convex quadratic constraints. *Mathematical programming*, 87:401–426, 2000.
43. H.A. Le Thi and T. Pham Dinh. The DC (difference of convex functions) programming and DCA revisited with DC models of real world nonconvex optimization problems. *Annals of Operations Research*, 133(1-4):23–46, 2005.
44. L. Liberti, C. Lavor, N. Maculan, and A. Mucherino. Euclidean distance geometry and applications. *SIAM Review*, 56(1):3–69, 2014.
45. M. C. Lin and D. Manocha. Collision and proximity queries. In J. ORourke and E. Goodman, editors, *Handbook of Discrete and Computational Geometry*. CRC Press, 2004.
46. S. Liu, W. Cui, Y. Wu, and M. Liu. A survey on information visualization: recent advances and challenges. *The Visual Computer*, 30(12):1373–1393, 2014.
47. N. Mladenović, M. Dražić, V. Kovačevic-Vujčić, and M. Čangalović. General variable neighborhood search for the continuous optimization. *European Journal of Operational Research*, 191(3):753–770, 2008.
48. S. Olafsson, X. Li, and S. Wu. Operations research and data mining. *European Journal of Operational Research*, 187(3):1429–1448, 2008.
49. C. J. Ong and E. G. Gilbert. Growth distances: New measures for object separation and penetration. *IEEE Transactions on Robotics and Automation*, 12(6):888–903, 1996.
50. K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine*, 2:559–572, 1901.
51. T. Pham Dinh and H.A. Le Thi. Convex analysis approach to d.c. programming: Theory, algorithms and applications. *Acta Mathematica Vietnamica*, 22(1):289–355, 1997.

52. T. Pham Dinh and H.A. Le Thi. A branch-and-bound method via DC optimization algorithm and ellipsoidal technique for box constrained nonconvex quadratic programming problems. *Journal of Global Optimization*, 13:171–206, 1998.
53. T. K. Pong and P. Tseng. (Robust) edge-based semidefinite programming relaxation of sensor network localization. *Mathematical programming*, 130(2):321–358, 2011.
54. R. L. Rabello, G. R. Mauri, G. M. Ribeiro, and L. A. N. Lorena. A clustering search metaheuristic for the point-feature cartographic label placement problem. *European Journal of Operational Research*, 234(3):802–808, 2014.
55. A. M.-C. So and Y. Ye. Theory of semidefinite programming for sensor network localization. *Mathematical Programming*, 109(2-3):367–384, 2007.
56. B. Speckmann, M. van Kreveld, and S. Florisson. A linear programming approach to rectangular cartograms. In *Proceedings of the 12th International Symposium on Spatial Data Handling*, pages 527–546. Springer, 2006.
57. J. Thomas and P.C. Wong. Visual analytics. *IEEE Computer Graphics and Applications*, 24(5):20–21, 2004.
58. W. Tobler. Thirty five years of computer cartograms. *Annals of the Association of American Geographers*, 94(1):58–73, 2004.
59. W.S. Torgerson. *Theory and Methods of Scaling*. Wiley, 1958.
60. M.W. Trosset. Extensions of classical multidimensional scaling via variable reduction. *Computational Statistics*, 17:147–163, 2002.
61. P. Tseng. Second-order cone programming relaxation of sensor network localization. *SIAM Journal on Optimization*, 18(1):156–185, 2007.
62. H. Tuy. *Convex Analysis and Global Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.
63. S. Umetani, M. Yagiura, S. Imahori, T. Imamichi, K. Nonobe, and T. Ibaraki. Solving the irregular strip packing problem via guided local search for overlap minimization. *International Transactions in Operational Research*, 16(6):661–683, 2009.
64. R. Vaughan. Approximate formulas for average distances associated with zones. *Transportation Science*, 18(3):231–244, 1984.
65. Z. Wang, S. Zheng, Y. Ye, and S. Boyd. Further relaxations of the semidefinite programming approach to sensor network localization. *SIAM Journal on Optimization*, 19(2):655–673, 2008.

Appendix

Proof of Proposition 1

$$
\begin{aligned}
&\lambda F_1 + (1-\lambda)F_2 = \\
&= \sum_{\substack{i,j=1,\ldots,N \\ i\neq j}} \left\{ \lambda \left[g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) - \kappa\delta_{ij} \right]^2 - (1-\lambda) g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) \right\} \\
&= \sum_{\substack{i,j=1,\ldots,N \\ i\neq j}} \left\{ (3\lambda - 1) g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) + 2\lambda\kappa^2\delta_{ij}^2 - \lambda(g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) + \kappa\delta_{ij})^2 \right\}
\end{aligned}
$$

In Section 3, the convexity of the function g_{ij} was stated. Moreover, since g_{ij}, λ, $\delta_{ij} \geq 0$, then $g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau)$, $2\lambda\kappa^2\delta_{ij}^2$ and $(g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) + \kappa\delta_{ij})^2$ are convex. Finally, $(3\lambda - 1)g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau)$ is convex for $3\lambda - 1 \geq 0$ and concave otherwise. □

Proof of Proposition 2

For convex sets A_1 and A_2 with nonempty interior, the condition in the definition of penetration depth stated in Section 2.2 is equivalent to the existence of a separating hyperplane between the sets $\boldsymbol{p} + A_1$ and A_2, i.e., of some $\boldsymbol{\xi} \neq 0$, such that

$$\boldsymbol{\xi}^\top(\boldsymbol{p} + \boldsymbol{a}_1) \leq \boldsymbol{\xi}^\top \boldsymbol{a}_2 \qquad \forall \boldsymbol{a}_1 \in A_1, \boldsymbol{a}_2 \in A_2.$$

Without loss of generality, we can consider $\|\boldsymbol{\xi}\| = 1$ and thus we have

$$\begin{aligned} \pi(A_1, A_2) = \min_{\boldsymbol{p},\boldsymbol{\xi} \in \mathbb{R}^n} \ & \|\boldsymbol{p}\| \\ \text{s.t.} \ & \boldsymbol{\xi}^\top(\boldsymbol{p} + \boldsymbol{a}_1) \leq \boldsymbol{\xi}^\top \boldsymbol{a}_2 \qquad \forall \boldsymbol{a}_1 \in A_1, \boldsymbol{a}_2 \in A_2 \\ & \|\boldsymbol{\xi}\| = 1. \end{aligned}$$

Thus, h_{ij} can be written as follows

$$\begin{aligned} h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = \min_{\boldsymbol{p},\boldsymbol{\xi} \in \mathbb{R}^n} \ & \|\boldsymbol{p}\| \\ \text{s.t.} \ & \boldsymbol{\xi}^\top(\boldsymbol{p} + \boldsymbol{c}_i + \tau r_i \boldsymbol{x}_i) \leq \boldsymbol{\xi}^\top(\boldsymbol{c}_j + \tau r_j \boldsymbol{x}_j) \qquad \forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{B} \\ & \|\boldsymbol{\xi}\| = 1. \end{aligned}$$

Equivalently, the first constraint, i.e.,

$$\boldsymbol{\xi}^\top(\boldsymbol{p} + \boldsymbol{c}_i + \tau r_i \boldsymbol{x}_i) \leq \boldsymbol{\xi}^\top(\boldsymbol{c}_j + \tau r_j \boldsymbol{x}_j) \quad \forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{B},$$

can be written as follows,

$$\boldsymbol{\xi}^\top(\boldsymbol{p} + \boldsymbol{c}_i) + \tau r_i \max_{\boldsymbol{x} \in \mathcal{B}} \boldsymbol{\xi}^\top \boldsymbol{x} \leq \boldsymbol{\xi}^\top \boldsymbol{c}_j + \tau r_j \min_{\boldsymbol{x} \in \mathcal{B}} \boldsymbol{\xi}^\top \boldsymbol{x}.$$

Let $\sigma_{\mathcal{B}}$ be the support function of $\mathcal{B}$, i.e.,

$$\sigma_{\mathcal{B}}(z) = \max_y \{y^\top z : y \in \mathcal{B}\}$$

Since $\mathcal{B}$ is assumed to be symmetric with respect to the origin, we have

$$\begin{aligned} \max_{x \in \mathcal{B}} \boldsymbol{\xi}^\top x &= \sigma_{\mathcal{B}}(\boldsymbol{\xi}) \\ \min_{x \in \mathcal{B}} \boldsymbol{\xi}^\top x &= -\sigma_{\mathcal{B}}(\boldsymbol{\xi}). \end{aligned}$$

Hence, by replacing the expression of the support function in the constraint above, one has

$$\begin{aligned} h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = \min_{\boldsymbol{p},\boldsymbol{\xi} \in \mathbb{R}^n} \ & \|\boldsymbol{p}\| \\ \text{s.t.} \ & \boldsymbol{\xi}^\top \boldsymbol{p} \leq \boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi}) \\ & \|\boldsymbol{\xi}\| = 1. \end{aligned}$$

For $\boldsymbol{\xi}$ fixed with $\|\boldsymbol{\xi}\| = 1$, let $\eta(\boldsymbol{\xi}) = \boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})$. It follows that the inner minimum in $h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau)$, is the distance from the origin to the halfspace $\boldsymbol{\xi}^\top \boldsymbol{p} \leq \eta(\boldsymbol{\xi})$, and such distance equals 0, if 0 belongs to the halfspace, i.e., if $0 \leq \boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})$, and $-\eta(\boldsymbol{\xi})$ else. Hence

$$\begin{aligned} h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) &= \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \max\left\{0, -\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\} \\ &= \max\left\{0, \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{-\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\}\right\} \end{aligned}$$

But, for $\boldsymbol{\xi}$ fixed, the function $(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) \longmapsto -\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})$ is affine, and thus the function $(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) \longmapsto \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{-\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\}$ is the minimum of affine functions, and is thus concave. Hence, h_{ij} is the maximum between 0 and a concave function, which is DC, whose decomposition is

$$\begin{aligned} &h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = \\ &= \max\left\{0, \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{-\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\}\right\} \\ &= \max\left\{- \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{-\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\}, 0\right\} + \min_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{-\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) + \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\} \\ &= \max\left\{\max_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\}, 0\right\} - \max_{\substack{\boldsymbol{\xi}\in\mathbb{R}^n \\ \|\boldsymbol{\xi}\|=1}} \left\{\boldsymbol{\xi}^\top(\boldsymbol{c}_j - \boldsymbol{c}_i) - \tau(r_i + r_j)\sigma_{\mathcal{B}}(\boldsymbol{\xi})\right\} \\ &= u_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) - (u_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) - h_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau)). \end{aligned}$$

□

Proof of Proposition 3

Before giving the proof of Proposition 3, the following technical result is needed.

Lemma 1 *Let $\beta_{ij} \in \mathbb{R}$ be such that $\beta_{ij} \geq 2\|r_i\boldsymbol{b}_i - r_j\boldsymbol{b}_j\|^2$, $\forall \boldsymbol{b}_i, \boldsymbol{b}_j \in \mathcal{B}$. Then, g_{ij}^2 can be expressed as a DC function, $g_{ij}^2 = u_{ij} - (u_{ij} - g_{ij}^2)$, where*

$$u_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = 2\|\boldsymbol{c}_i - \boldsymbol{c}_j\|^2 + \beta_{ij}\tau^2.$$

Proof

$$\begin{aligned} &g_{ij}^2(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) = \\ &= \min_{\boldsymbol{b}_i, \boldsymbol{b}_j \in \mathcal{B}} \|\boldsymbol{c}_i - \boldsymbol{c}_j + \tau(r_i\boldsymbol{b}_i - r_j\boldsymbol{b}_j)\|^2 \\ &= \min_{\boldsymbol{b}_i, \boldsymbol{b}_j \in \mathcal{B}} \left\{\|\boldsymbol{c}_i - \boldsymbol{c}_j\|^2 + \tau^2\|r_i\boldsymbol{b}_i - r_j\boldsymbol{b}_j\|^2 + 2\tau(\boldsymbol{c}_i - \boldsymbol{c}_j)^\top(r_i\boldsymbol{b}_i - r_j\boldsymbol{b}_j)\right\} \end{aligned}$$

$$= \min_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B}} \Big\{\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\tau^2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2$$
$$+\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\tau^2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j)\|^2\Big\}$$
$$= 2\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\beta_{ij}\tau^2+\min_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B}}\Big\{-\beta_{ij}\tau^2+2\tau^2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau\left(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\right)\|^2\Big\}$$
$$= 2\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\beta_{ij}\tau^2+\min_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B}}\Big\{\tau^2\left(2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\beta_{ij}\right)-\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau\left(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\right)\|^2\Big\}$$
$$= 2\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\beta_{ij}\tau^2-\max_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B}}\Big\{\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau\left(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\right)\|^2-\tau^2\left(2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\beta_{ij}\right)\Big\}$$

Observe that taking $\beta_{ij}\in\mathbb{R}$ such that

$$2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\beta_{ij}\le 0 \qquad \forall \boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B},$$

the function

$$(\boldsymbol{c}_i,\boldsymbol{c}_j,\tau)\longmapsto\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau\left(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\right)\|^2-\tau^2\left(2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\beta_{ij}\right)$$

is convex. Since the maximum of convex functions is convex, hence taking $u_{ij}=2\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+\beta_{ij}\tau^2$, we have obtained a DC decomposition for g_{ij}^2 as in the statement. □

We prove now Proposition 3:

If $\lambda<\frac{1}{3}$, considering Proposition 1, one has

$$\lambda F_1+(1-\lambda)F_2=\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}\left\{2\lambda\kappa^2\delta_{ij}^2-\left[\lambda(g_{ij}+\kappa\delta_{ij})^2-(3\lambda-1)g_{ij}^2(\boldsymbol{c}_i,\boldsymbol{c}_j,\tau)\right]\right\},$$

and thus $u=\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}2\lambda\kappa^2\delta_{ij}^2$ holds.

If $\lambda\geq\frac{1}{3}$, by using the DC decomposition for g_{ij}^2 obtained in Lemma 1 and Proposition 1, one has

$$\lambda F_1+(1-\lambda)F_2=$$
$$=\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}\left\{(3\lambda-1)g_{ij}^2(\boldsymbol{c}_i,\boldsymbol{c}_j,\tau)+2\lambda\kappa^2\delta_{ij}^2-\lambda(g_{ij}(\boldsymbol{c}_i,\boldsymbol{c}_j,\tau)+\kappa\delta_{ij})^2\right\}$$
$$=\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}\Big\{2(3\lambda-1)\|\boldsymbol{c}_i-\boldsymbol{c}_j\|^2+(3\lambda-1)\beta_{ij}\tau^2+2\lambda\kappa^2\delta_{ij}^2-\Big[\lambda(g_{ij}(\boldsymbol{c}_i,\boldsymbol{c}_j,\tau)+\kappa\delta_{ij})^2$$
$$+\ (3\lambda-1)\max_{\boldsymbol{b}_i,\boldsymbol{b}_j\in\mathcal{B}}\Big\{\|\boldsymbol{c}_i-\boldsymbol{c}_j-\tau\left(r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\right)\|^2-\tau^2\left(2\|r_i\boldsymbol{b}_i-r_j\boldsymbol{b}_j\|^2-\beta_{ij}\right)\Big\}\Big]\Big\}$$
$$=\sum_{i=1,\ldots,N}\left\{8(3\lambda-1)(N-1)\|\boldsymbol{c}_i\|^2\right\}+(3\lambda-1)\tau^2\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}\beta_{ij}+2\lambda\kappa^2\sum_{\substack{i,j=1,\ldots,N\\ i\neq j}}\delta_{ij}^2$$

$$
- \sum_{\substack{i,j=1,\dots,N \\ i \neq j}} \Big[2(3\lambda - 1) \| \boldsymbol{c}_i + \boldsymbol{c}_j \|^2 + \lambda (g_{ij}(\boldsymbol{c}_i, \boldsymbol{c}_j, \tau) + \kappa \delta_{ij})^2
$$

$$
+ (3\lambda - 1) \max_{\boldsymbol{b}_i, \boldsymbol{b}_j \in \mathcal{B}} \Big\{ \| \boldsymbol{c}_i - \boldsymbol{c}_j - \tau \left(r_i \boldsymbol{b}_i - r_j \boldsymbol{b}_j \right) \|^2 - \tau^2 \Big(2 \| r_i \boldsymbol{b}_i - r_j \boldsymbol{b}_j \|^2 - \beta_{ij} \Big) \Big\} \Big]
$$

□