Is a Finite Intersection of Balls Covered by a Finite Union of Balls in Euclidean Spaces?

Vincent Runge[1] (ORCID)

Abstract

Considering a finite intersection of balls and a finite union of other balls in an Euclidean space, we propose an exact method to test whether the intersection is covered by the union. We reformulate this problem into quadratic programming problems. For each problem, we study the intersection between a sphere and a Voronoi-like polyhedron. That way, we get information about a possible overlap between the frontier of the union and the intersection of balls. If the polyhedra are non-degenerate, the initial nonconvex geometric problem, which is NP-hard in general, is tractable in polynomial time by convex optimization tools and vertex enumeration. Under some mild conditions, the vertex enumeration can be skipped. Simulations highlight the accuracy and efficiency of our approach compared with competing algorithms in Python for nonconvex quadratically constrained quadratic programming. This work is motivated by an application in statistics to the problem of multidimensional changepoint detection using pruned dynamic programming algorithms.

1 Introduction

Geometric problems for balls often separately address the intersection and the union problems. Without optimization tools, the detection of a nonempty intersection

---

✉ Vincent Runge
vincent.runge@univ-evry.fr

[1] Université Paris-Saclay, CNRS, Univ Evry, Laboratoire de Mathématiques et Modélisation d'Evry, 91037 Evry-Courcouronnes, France

between balls is difficult to solve. Helly-type theorems can be adapted to balls [1,2], but no efficient algorithm arises from this approach. The union of balls is a problem linked in the literature to molecular structures, where the volume and the surface area of molecules in 3D are important properties. Powerful algorithms based on Voronoi diagrams have been recently developed [3,4]. Even if the number of balls is small, that is more than two, the exact computation of simple geometric properties, as volume, is a challenging question [5].

One of the first problems to associate union and intersection is the historical disk covering problem, which consists in finding the minimum number of identical disks (with a given radius) needed to cover the unit disk [6]. This problem is still open and remains mainly unsolved, although research on this subject is active [7,8] as it has practical applications, for example in optical interferometry [9].

Our problem also involves covers but is different in several important ways. We study the covering of an intersection of balls by other balls, but they are not necessary of the same size. Furthermore, we do not consider the question of the optimal covering, but only the covering test. This problem is part of computational geometry problems and, as far as we know, is original in the literature.

Quadratically constrained quadratic programming (QCQP) problems are a major tool for many practical applications: problems of transmit beamforming in wireless communication [10] or signal processing [11] have stimulated the development of this research area. Asking whether a finite intersection of balls is covered by a finite union of other balls can be reformulated as a collection of nonconvex QCQP problems. However, this approach often fails to return the exact solution and is based on time-consuming iterative algorithms.

In this paper, we transform our problem into quadratic programming problems by introducing hyperplanes "separating" two balls. For each problem, we study the intersection between a sphere and a polyhedron to get information about a possible overlap between the frontier of the union and the intersection of balls. That way, we are able to build an efficient and exact algorithm with polynomial running time in most cases.

This work is motivated by an application to a changepoint detection method in statistics as explained in "Appendix A."

The outline of this paper is the following. In Sect. 2, we present the problem and introduce the notations. We also give its naive standard resolution by QCQP problems. We end the section with a detailed presentation of our solution. Section 3 focuses on the resolution of a unique QP problem, giving the link between this problem and the initial geometric structure. In "Appendix B," we propose a similar method adapted to covering tests for a large number of exclusion balls and for sequential tests. With some conditions on the centers and radii of the balls, we are able to ensure that the polyhedron is unbounded and can then skip the maximization problem to only consider the convex quadratic one. We present in Sect. 4 several cases where the only problem to solve is convex. If we have only one exclusion ball, we introduce the so-called concave QCQP problem and highlight simplified results. Finally, in Sect. 5, we compare our approach with recent methods of the Python library 'qcqp' gathering together the best approaches for solving nonconvex QCQP problem. These simulations highlight the benefit of our method specifically developed for the problem at hand.

2 Preliminaries

2.1 Problem Description

We consider two finite sets of balls in an Euclidean space, $\Lambda = \{B_1, \ldots, B_p\}$ and $V = \{\mathbf{B}_1, \ldots, \mathbf{B}_q\}$, with $p, q \in \mathbb{N}^*$ and arbitrary centers and radii. We introduce the intersection set $\mathcal{I} = \cap_{i=1}^{p} B_i$ and union set $\mathcal{U} = \cup_{j=1}^{q} \mathbf{B}_j$. Our problem consists in finding an exact and efficient method to decide whether the inclusion $\mathcal{I} \subset \mathcal{U}$ is true or false. Denoting by $\mathcal{U}^{\mathrm{c}}$ the complement of $\mathcal{U}$, this problem is equivalent to the study of the emptiness of $\mathcal{I} \cap \mathcal{U}^{\mathrm{c}}$, which is a challenging question, both theoretically and computationally, due to the non-convexity of the $\mathbf{B}_j^{\mathrm{c}}$ sets ($j = 1, \ldots, q$).

With $\mathbb{R}^n$ the n-dimensional Euclidean space, $n \geq 2$, we denote by $\langle \cdot, \cdot \rangle$ the canonic inner product in $\mathbb{R}^n$ and by $\|\cdot\|$ its associated norm. The open balls B_i and closed balls $\mathbf{B}_j$ are defined by their centers $c_i, \mathbf{c}_j \in \mathbb{R}^n$ and radii $R_i, \mathbf{R}_j \in \mathbb{R}_+^*$, respectively. Thus,

$$B_i = \{x \in \mathbb{R}^n \ : \ \|x - c_i\|^2 < R_i^2\} \quad \text{and} \quad \mathbf{B}_j = \{x \in \mathbb{R}^n \ : \ \|x - \mathbf{c}_j\|^2 \leq \mathbf{R}_j^2\},$$

where, for example, $\|x - c_i\|^2 = \sum_{k=1}^{n}(x_k - c_{ik})^2$, with $x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$. We assume that the centers of balls in $\Lambda \cup V$ are all different (non-concentric) and that for all $(B_a, B_b) \in \Lambda^2$ and $(\mathbf{B}_c, \mathbf{B}_d) \in V^2$ (with $a \neq b$ and $c \neq d$) we have:

$$\begin{aligned}
B_a \cap B_b \neq \emptyset, \quad & B_a \cap B_b^{\mathrm{c}} \neq \emptyset, \\
B_a \cap \mathbf{B}_c \neq \emptyset, \quad & B_a \cap \mathbf{B}_c^{\mathrm{c}} \neq \emptyset, \\
\mathbf{B}_c \cap \mathbf{B}_d^{\mathrm{c}} \neq \emptyset. &
\end{aligned} \tag{1}$$

These conditions can be verified in polynomial time in a preprocessing step in order to avoid unnecessary computations (for example, we remove B_2 in Λ, if $B_1 \cap B_2^{\mathrm{c}} = \emptyset$) or trivial solutions (for example, if $B_1 \cap B_2 = \emptyset$, then $\mathcal{I} \cap \mathcal{U}^{\mathrm{c}} = \emptyset$).

Remark 2.1 We consider open balls B_i. This is useful in proofs because we get the set $\mathcal{I} \cap \mathcal{U}^{\mathrm{c}}$ open. With closed balls B_i, the decision problem is the same.

2.2 A Standard Approach

We reformulate our geometric problem into a collection of q quadratically constrained quadratic programming (QCQP) : for $j = 1, \ldots, q$,

$$P_0(j) : \begin{cases}
\max_{x \in \mathbb{R}^n} (\|x - \mathbf{c}_j\|^2 - \mathbf{R}_j^2), \\
\text{such that } \|x - c_i\|^2 - R_i^2 \leq 0, \quad i = 1, \ldots, p, \\
\qquad\qquad \|x - \mathbf{c}_k\|^2 - \mathbf{R}_k^2 \geq 0, \quad k = 1, \ldots, j - 1.
\end{cases} \tag{2}$$

We solve these problems sequentially for increasing integers j. Notwithstanding the difficult resolution of (2), the naive Algorithm 1 gives an exact response concerning the geometric inclusion $\mathcal{I} \subset \mathcal{U}$.


Algorithm 1 QCQP-based naive algorithm

1: **procedure** ISINCLUDED(Λ, V)
2: $response \leftarrow false, j \leftarrow 1$
3: **while** $j < q + 1$ **do**
4:     $x^*(j) = \text{Argmax}\{P_0(j)\}$ ▷ We solve the nonconvex QCQP problem $P_0(j)$
5:     **if** $\|x^*(j) - \mathbf{c}_j\|^2 \leq \mathbf{R}_j^2$ **then**
6:         $response \leftarrow true, j \leftarrow q$
7:     **end if**
8:     $j \leftarrow j + 1$
9: **end while**
10: **return** $response$

Fig. 1 The dark blue region shows us that $\mathcal{I} \not\subset \mathcal{U}$. Solving $P_0(1)$ stucks the solution in the green region. If only continuous and feasible moves are allowed, as with an interior point method, we would fail to solve $P_0(2)$ as its feasible region is not connected
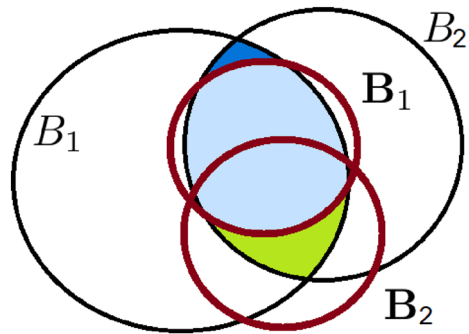
The feasible region $\mathcal{X}_j$ is the subset of points in $\mathbb{R}^n$ satisfying the constraints in problem $P_0(j)$. If the (nonempty) feasible region $\mathcal{X}_j$ for problem $P_0(j)$ is strictly included into $\mathbf{B}_j$, that is, $\|x - \mathbf{c}_j\|^2 - \mathbf{R}_j^2 < 0$ for all $x \in \mathcal{X}_j$, then $\mathcal{X}_{j+1} = \emptyset$ and $\cup_{k=1}^{j} \mathbf{B}_k$ covers $\mathcal{I}$. Thus, Algorithm 1 tries to cover $\mathcal{I}$ with the balls in V, adding them one by one. The algorithm stops as soon as the feasible region becomes empty (justifying the while loop).

Other reformulations are possible. We choose one of them ensuring the feasibility of the considered problem at each step j. If feasibility is not required, solving $P_0(q)$ is enough work. In both cases, the non-convexity of the initial geometric problem is transferred to problems $P_0(j)$ for which the feasible regions (if $q > 1$ and $j > 1$) and the objective functions (with the standard formulation with a minimization) are nonconvex. For this reason, line search strategies for interior point methods can fail to converge toward the global optimum. On a simple example in Fig. 1, we highlight the failing of any point following methods.

For decades, many approximate methods for the generic nonconvex QCQP problems (2) have been developed (see the review [12]) to avoid using primal methods. They are relaxation methods that usually convexify the nonconvex part of the problem or solve successive convex optimization approximate problems. Approaches as semi-definite relaxation (SDR) [13], reformulation linearization technique (RLT) [14] or successive convex approximation (SCA) [15,16] are among the most popular. However, they are often computationally greedy (using n^2 unknowns) and only converge toward KKT stationary points.

We propose in this paper, to the best of our knowledge, the first exact and simple problem-solving method for the subclass of nonconvex QCQP problems involving only balls and complement of balls.

2.3 Proposed Solution

Our QCQP problem is specific as it only involves balls. We can take advantage of the fact that the intersection of two spheres, when they meet, belongs to an hyperplane. Considering a sphere $\mathbf{S}_j = \partial \mathbf{B}_j$, where $\partial(\cdot)$ denotes the frontier operator, we build hyperplanes as soon as a sphere $S_i = \partial B_i$ ($i = 1, \ldots, p$) or $\mathbf{S}_k = \partial \mathbf{B}_k$ ($k \neq j$, $k = 1, \ldots, q$) intersects $\mathbf{S}_j$. Each hyperplane defines a favored half-space. The intersection of the obtained half-spaces yields an open convex polyhedron $\mathtt{P}_j$ which can be seen as a Voronoi-like structure (more details in Sect. 2).

The solution proposed in this paper is as follows. We introduce the notation $\mathfrak{U}_j = \cup_{k=1 : k \neq j}^{q} \mathbf{B}_k$ and prove that we are able to detect an intersection between $\mathcal{I}$ and $\mathbf{S}_j \setminus \mathfrak{U}_j$ only by using the convex polyhedron $\mathtt{P}_j$. This method is based on the set equality $\mathbf{S}_j \cap (\mathcal{I} \setminus \mathfrak{U}_j) = \mathbf{S}_j \cap \mathtt{P}_j$. The closure of $\mathtt{P}_j$ is denoted Π_j ($\Pi_j = \overline{\mathtt{P}}_j$). Detection of a nonempty intersection $\mathbf{S}_j \cap \mathtt{P}_j$ can be handled solving the following $2q$ (q for the minimum and q for the maximum) quadratic programs (QP): for $j = 1, \ldots, q$,

$$
P_1(j) : \begin{cases} \operatorname{extr}_{x \in \mathbb{R}^n} (\|x - \mathbf{c}_j\|^2 - \mathbf{R}_j^2), \\ \text{such that } x \in \Pi_j. \end{cases} \tag{3}
$$

The minimum QP problem is tractable in polynomial time [17,18]. Solving $P_1(j)$ for the maximum is a nonconvex (concave) problem, which can be solved by vertex enumeration in polynomial time if the polyhedron Π_j is non-degenerate [19]. There exist particular cases, in practice rarely encountered, for which the vertex enumeration problem remains NP-hard [20].

In fact, as soon as we find a feasible point strictly inside the ball $\mathbf{B}_j$ and another one strictly outside, we have $\mathbf{S}_j \cap \mathtt{P}_j \neq \emptyset$ and we will prove that we get $\mathcal{I} \setminus \mathcal{U} \neq \emptyset$.

If for all j, this intersection $\mathbf{S}_j \cap \mathtt{P}_j$ is empty, we have $\mathbf{S}_j \cap (\mathcal{I} \setminus \mathfrak{U}_j) = \mathcal{I} \cap (\mathbf{S}_j \setminus \mathfrak{U}_j) = \emptyset$ for all j, that is $\mathcal{I} \cap \partial \mathcal{U} = \emptyset$. Thus, with our method, we shift from the emptiness of the set $\mathcal{I} \setminus \mathcal{U}$ to the emptiness of the set $\mathcal{I} \cap \partial \mathcal{U}$. To solve the initial covering problem ($\mathcal{I} \setminus \mathcal{U} = \emptyset$?), we notice that we only need to know a point inside $\mathcal{I}$ and test whether this point is also inside $\mathcal{U}$ to decide our question.

3 Equivalent Quadratic Programming

We now focus on the building of a unique polyhedron and show its close link with the initial nonconvex sets $\mathcal{I} \setminus \mathfrak{U}_q$. Finally, we show that the q polyhedra and their associated q quadratic programs solve the covering problem.
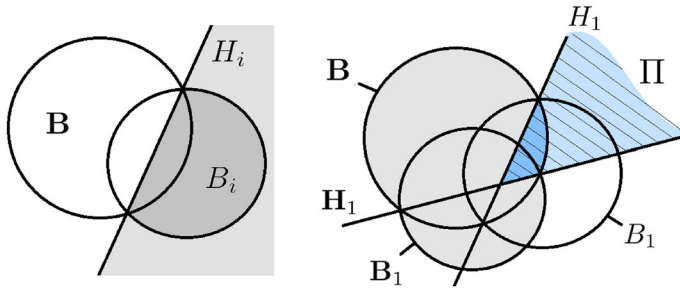
Fig. 2 Left: half-space H_i defined by the intersection of the spheres $\mathbf{S} = \partial\mathbf{B}$ and $S_i = \partial B_i$ and containing $B_i \setminus \mathbf{B}$. Right: example of feasible region Π (the hatched area) with $p = 1$ and $q = 2$. Here, $\mathrm{P}^+ = H_1$ and $\mathrm{P}^- = \mathbf{H}_1$. The darker shade of blue is the subset of Π lying inside the ball $\mathbf{B}$

3.1 Linear Constraints

From now one, we consider a unique problem centered on the closed ball $\mathbf{B}_q$ and write $\mathbf{B} = \mathbf{B}_q$, $\mathbf{c} = \mathbf{c}_q$, $\mathbf{R} = \mathbf{R}_q$, $\mathrm{P} = \mathrm{P}_q$, $\Pi = \Pi_q$, $\mathfrak{U} = \mathfrak{U}_q$. For all $i \in \{1, \dots, p\}$ such that $\mathbf{S} = \partial\mathbf{B}$ and $S_i = \partial B_i$ intersect, we define the functions $h_i : \mathbb{R}^n \to \mathbb{R}$ as

$$h_i(x) = 2\langle x, \mathbf{c} - c_i \rangle - \|\mathbf{c}\|^2 + \|c_i\|^2 + \mathbf{R}^2 - R_i^2 .$$

The associated hyperplane equation is $h_i(x) = 0$ and we have the open half-space containing the set $B_i \setminus \mathbf{B}$: $H_i = \{x \in \mathbb{R}^n : h_i(x) < 0\}$. The geometric configuration of the balls $\mathbf{B}$ and B_i with the half-space H_i is given in Fig. 2 (left). All the balls in Λ intersect $\mathbf{B}$, and the inclusion $\mathbf{B} \subset B_i$ is not excluded (see conditions (1)): in this case, we do not build any hyperplane.

Similar hyperplanes and half-spaces are built between spheres $\mathbf{S} = \partial\mathbf{B}$ and $\mathbf{S}_j = \partial\mathbf{B}_j$ for $j \in \{1, \dots, q-1\}$ when they intersect, but here, we consider the half-space containing $\mathbf{B} \setminus \mathbf{B}_j$. Thus, we define the functions $\mathbf{h}_j : \mathbb{R}^n \to \mathbb{R}$ such that

$$\mathbf{h}_j(x) = 2\langle x, \mathbf{c} - \mathbf{c}_j \rangle - \|\mathbf{c}\|^2 + \|\mathbf{c}_j\|^2 + \mathbf{R}^2 - \mathbf{R}_j^2 ,$$

and the associated hyperplane equation is $\mathbf{h}_j(x) = 0$ with the open half-space $\mathbf{H}_j = \{x \in \mathbb{R}^n : -\mathbf{h}_j(x) < 0\}$. If $\mathbf{B} \cap \mathbf{B}_j = \emptyset$, a case not excluded in (1), we also do not build hyperplane.

With these half-spaces, we define the open convex polyhedron

$$\mathrm{P} = \{x \in \mathbb{R}^n : h_i(x) < 0, \ -\mathbf{h}_j(x) < 0, \ i = 1, \dots, p, \ j = 1, \dots, q-1\}$$

and the polyhedra $\mathrm{P}^+ = \cap_{i=1}^{p} H_i$ and $\mathrm{P}^- = \cap_{j=1}^{q-1} \mathbf{H}_j$ such that $\mathrm{P} = \mathrm{P}^+ \cap \mathrm{P}^-$. The open polyhedron P will be used to prove geometrical properties, whereas its closure, $\Pi = \overline{\mathrm{P}} = \overline{\mathrm{P}^+ \cap \mathrm{P}^-} = \Pi^+ \cap \Pi^-$, is the feasible region of the QP problem $P_1(q)$ with $\Pi^+ = \overline{\mathrm{P}^+}$ and $\Pi^- = \overline{\mathrm{P}^-}$.

An efficient resolution of the nonconvex problem of type P_1 (cf (3)) for maximum is made possible, solving for example a vertex enumeration problem [19]. An example of feasible region is drawn in Fig. 2 (right).

Before proving the equivalence of Problems (2) and (3), we present some simple equalities and inclusions used throughout this paper between sets involving balls and half-spaces.

Lemma 3.1 *For $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q-1\}$, we have the relations:*

$$(a) \quad \mathbf{S} \cap B_i = \mathbf{S} \cap H_i \,, \qquad\qquad (b) \quad \mathbf{S} \cap \mathbf{B}_j^c = \mathbf{S} \cap \mathbf{H}_j \,,$$

$$(c) \quad B_i \setminus \mathbf{B} \subset H_i \,, \qquad\qquad\qquad (d) \quad \mathbf{B} \setminus \mathbf{B}_j \subset \mathbf{H}_j \,.$$

The proof is straightforward.

3.2 Links Between the Polyhedron and the Initial Geometric Problem

We give a set equality linking the convex polyhedron P and the open set $\mathcal{I} \setminus \mathfrak{U}$ involved in the initial geometric problem.

Corollary 3.1 $\mathbf{S} \cap (\mathcal{I} \setminus \mathfrak{U}) = \mathbf{S} \cap \mathrm{P}$.

Proof $\mathbf{S} \cap \mathcal{I} = \mathbf{S} \cap (\cap_{i=1}^p B_i) = \cap_{i=1}^p (\mathbf{S} \cap B_i) = \cap_{i=1}^p (\mathbf{S} \cap H_i) = \mathbf{S} \cap (\cap_{i=1}^p H_i) = \mathbf{S} \cap \mathrm{P}^+$ by relation (a) of Lemma 3.1. In the same way, we have $\mathbf{S} \cap \mathfrak{U}^c = \mathbf{S} \cap (\cup_{j=1}^{q-1} \mathbf{B}_j)^c = \mathbf{S} \cap (\cap_{j=1}^{q-1} \mathbf{B}_j^c) = \cap_{j=1}^{q-1} (\mathbf{S} \cap \mathbf{B}_j^c) = \cap_{j=1}^{q-1} (\mathbf{S} \cap \mathbf{H}_j) = \mathbf{S} \cap \mathrm{P}^-$ by De Morgan's laws and relation (b) of Lemma 3.1. Therefore, $\mathbf{S} \cap \mathcal{I} \setminus \mathfrak{U} = (\mathbf{S} \cap \mathcal{I}) \cap (\mathbf{S} \cap \mathfrak{U}^c) = (\mathbf{S} \cap \mathrm{P}^+) \cap (\mathbf{S} \cap \mathrm{P}^-) = \mathbf{S} \cap \mathrm{P}$ and the relation is proven. $\qquad\square$

For $\epsilon \in \mathbb{R}$, we introduce the ball $\mathbf{B}^{R+\epsilon}$ with center $\mathbf{c}$ and radius $\mathbf{R} + \epsilon$. We also define $\mathbf{S}^{R+\epsilon} = \partial \mathbf{B}^{R+\epsilon}$. V_n is the volume (Lebesgue measure) in dimension n. In the following proposition, we connect the position of points in the closed feasible region Π to the spatial position of $\mathcal{I} \setminus \mathfrak{U}$.

Proposition 3.1 *If the volume of the feasible region is nonzero ($V_n(\Pi) > 0$), the following assertions are equivalent:*

(i) $\mathbf{S} \cap P \neq \emptyset$,
(ii) *there exists $(x^-, x^+) \in \Pi \times \Pi$ such that $\|x^- - \mathbf{c}\|^2 < \mathbf{R}^2 < \|x^+ - \mathbf{c}\|^2$,*
(iii) $V_{n-1}(\mathbf{S} \cap \Pi) > 0$ *(the Lebesgue measure of the surface area is nonzero),*
(iv) *there exists $r > 0$ such that, for all $\epsilon \in (-r, r)$, $\mathbf{S}^{R+\epsilon} \cap (\mathcal{I} \setminus \mathfrak{U}) \neq \emptyset$.*

To state this result, we use the following lemma.

Lemma 3.2 *With $\mathcal{O}$ an open set of $\mathbb{R}^n$, we have equivalence between propositions:*

(a) $\mathbf{S} \cap \mathcal{O} \neq \emptyset$;
(b) $V_{n-1}(\mathbf{S} \cap \mathcal{O}) > 0$;
(c) *there exists $r > 0$ such that, for all $\epsilon \in (-r, r)$, $\mathbf{S}^{R+\epsilon} \cap \mathcal{O} \neq \emptyset$.*

Proof With $\tilde{x} \in \mathbf{S} \cap \mathcal{O}$, there exists an open ball $B(\tilde{x}, \rho)$ centered on $\tilde{x}$ and of radius ρ, such that $B(\tilde{x}, \rho) \subset \mathcal{O}$. Thus, $V_{n-1}(\mathbf{S} \cap \mathcal{O}) \geq V_{n-1}(\mathbf{S} \cap B(\tilde{x}, \rho)) > 0$ and for all $\epsilon \in (-\rho, \rho)$, $\mathbf{S}^{R+\epsilon} \cap \mathcal{O} \subset \mathbf{S}^{R+\epsilon} \cap B(\tilde{x}, \rho) \neq \emptyset$. We have shown that $a) \implies b) \implies c)$. The implication $c) \implies a)$ is obvious, and Lemma is proven. $\qquad \square$

Proof of Proposition 3.1 The implication $i) \implies ii)$ is due to the openness of P. We prove the converse. As $V_n(\Pi) > 0$ and Π is convex, we have $\mathrm{P} \neq \emptyset$ and $\overline{\mathrm{P}} = \Pi$. Thus, there exist sequences (x_n^-) and (x_n^+) such that $\lim_{n \to +\infty} x_n^- = x^-$ and $\lim_{n \to +\infty} x_n^+ = x^+$ with $x_n^-, x_n^+ \in \mathrm{P}$ for all $n \in \mathbb{N}$. Therefore, there exists $N \in \mathbb{N}$ such that, for all $n \geq N$, $\|x_n^- - \mathbf{c}\|^2 < \mathbf{R}^2 < \|x_n^+ - \mathbf{c}\|^2$. By convexity of P, the segment $[x_n^-, x_n^+]$ is in P and $\mathbf{S} \cap [x_n^-, x_n^+] \neq \emptyset$ and we get $\mathbf{S} \cap \mathrm{P} \neq \emptyset$. We use the Lemma 3.2 $(a \iff b)$ with the open set P to prove the equivalence between $i)$ and $iii)$, knowing that $V_{n-1}(\mathbf{S} \cap \Pi) = V_{n-1}(\mathbf{S} \cap \mathrm{P})$. As $\mathbf{S} \cap (\mathcal{I} \setminus \mathfrak{U}) = \mathbf{S} \cap \mathrm{P}$ (see Corollary 3.1), we use again Lemma 3.2 $(b \iff c)$ with the open set $\mathcal{I} \setminus \mathfrak{U}$ to get the equivalence between propositions $iii)$ and $iv)$. $\qquad \square$

With this last result, we have shown that some propositions involving the closed polyhedron Π are related to results using P and hence $\mathcal{I} \setminus \mathfrak{U}$. We now combine these results with the q quadratic programming problems to solve the problem.

3.3 Quadratic Programming Decision

We still focus on a unique quadratic program:

$$P_1(q) = P_1 : \begin{cases} \mathrm{extr}_{x \in \mathbb{R}^n} \left(\|x - \mathbf{c}\|^2 - \mathbf{R}^2 \right), \\ \text{such that } x \in \Pi, \end{cases} \tag{4}$$

and we denote by x_{min}^* the value of the argument x for which the objective function attains its minimum over the set Π. If Π is bounded, x_{max}^* is one value of the argument x for which the objective function attains its maximum. Notice that x_{max}^* is not necessarily unique, hence the use of "one value" in previous sentence. If Π is unbounded, we consider thereafter that $\|x_{max}^* - \mathbf{c}\|^2 = +\infty$, even if x_{max}^* is not well-defined.

The non-existence of a point x^- or x^+ satisfying the strict inequalities in Proposition 3.1 case (ii) is related to the resolution of problem P_1. For example, there is no point x^- if $\mathbf{R}^2 \leq \|x_{min}^* - \mathbf{c}\|^2$. Before solving this extremum problem, one should verify the feasibility of the constraints. Studying the feasibility of these constraints, we get some inclusion criteria.

Proposition 3.2 If $V_n(\Pi^+) = 0$, then $\mathcal{I} \subset \mathbf{B}$ (or $\mathcal{I} = \emptyset$),
if $V_n(\Pi^-) = 0$, then $\mathbf{B} \subset \mathfrak{U}$,
if $V_n(\Pi) = 0$, then $\mathcal{I} \cap (\mathbf{S} \setminus \mathfrak{U}) = \emptyset$.

Proof $\mathcal{I} \setminus \mathbf{B} = \cap_{i=1}^p (B_i \setminus \mathbf{B}) \subset \cap_{i=1}^p H_i = \mathrm{P}^+$ using relation (c) of Lemma 3.1. If $V_n(\Pi^+) = 0$, then $\mathrm{P}^+ = \emptyset$ and we have $\mathcal{I} \subset \mathbf{B}$. With relation (d) of Lemma 3.1, $\mathbf{B} \setminus \mathfrak{U} = \cap_{j=1}^{q-1}(\mathbf{B} \setminus \mathbf{B}_j) \subset \cap_{j=1}^{q-1} \mathbf{H}_j = \mathrm{P}^-$. If $V_n(\Pi^-) = 0$, then $\mathrm{P}^- = \emptyset$ and we have $\mathbf{B} \subset \mathfrak{U}$. The last result is a direct application of Corollary 3.1. $\qquad \square$

We can now present our main result, showing the equivalent between the resolution of the QP problems $P_1(j)$, $j = 1, \ldots, q$, and the decision about $\mathcal{I} \setminus \mathcal{U}$.

Theorem 3.1 *If $V_n(\Pi) > 0$, we have the following results.*

(A) *If there exists $(x^-, x^+) \in \Pi \times \Pi$ such that $\|x^- - \mathbf{c}\|^2 < \mathbf{R}^2 < \|x^+ - \mathbf{c}\|^2$, then $\mathcal{I} \setminus \mathcal{U} \neq \emptyset$, $V_n(\mathcal{I} \setminus \mathcal{U}) > 0$ and $V_n(\mathcal{I} \cap \mathcal{U}) > 0$;*

(B) *if $\mathbf{R}^2 \leq \|x_{min}^* - \mathbf{c}\|^2$ or $\|x_{max}^* - \mathbf{c}\|^2 \leq \mathbf{R}^2$, then $\mathcal{I} \cap (\mathbf{S} \setminus \mathfrak{U}) = \emptyset$.*

Proof In case (A), Proposition 3.1 gives the existence of $r > 0$ and $x \in \mathbb{R}^n$ such that $x \in \mathbf{S}^{R+\frac{r}{2}} \cap (\mathcal{I} \setminus \mathfrak{U})$. In particular, $x \in \mathbf{S}^{R+\frac{r}{2}}$, then $x \in \mathbf{B}^c$. Consequently, $x \in (\mathcal{I} \setminus \mathfrak{U}) \cap \mathbf{B}^c = \mathcal{I} \setminus \mathcal{U}$ (an open set) and $V_n(\mathcal{I} \setminus \mathcal{U}) > 0$. With a point $x' \in \mathbf{S}^{R-\frac{r}{2}} \cap (\mathcal{I} \setminus \mathfrak{U})$, we get $\mathbf{B} \cap \mathcal{I} \neq \emptyset$ and $V_n(\mathcal{I} \cap \mathcal{U}) > 0$. Case (B) is the negation of case (A). That is, there exists $r > 0$ such that for all $\epsilon \in (0, r)$ or for all $\epsilon \in (-r, 0)$, we have $\mathbf{S}^{R+\epsilon} \cap (\mathcal{I} \setminus \mathfrak{U}) = \emptyset$ using Proposition 3.1. Therefore, $\mathcal{I} \cap (\mathbf{S} \setminus \mathfrak{U}) = \emptyset$. Otherwise, with $\mathcal{O} = \mathcal{I} \setminus \mathfrak{U}$ in Lemma 3.2 we get $\mathbf{S}^{R+\epsilon} \cap (\mathcal{I} \setminus \mathfrak{U}) \neq \emptyset$ for all $\epsilon \in (-r^*, r^*)$ and fixed $r^* > 0$, which is impossible. □

If case (B) is satisfied, the convex set $\mathcal{I}$ does not intersect the frontier $\mathbf{S} \setminus \mathfrak{U}$ and we consider another reference ball in V to solve a new P_1-type problem (3). If for all elements of V we get case (B), we have $\mathcal{I} \cap \partial \mathcal{U} = \emptyset$. Using a point $\hat{x}$ in $\mathcal{I}$, we test whether $\hat{x}$ is included in $\mathcal{U}$ to conclude.

4 Simplifications

4.1 Convex Reduction

We show that, if some mild conditions are satisfied, the feasible region Π of Problem P_2 (see (4)) can be unbounded and the maximum problem in Theorem 3.1 does not have to be solved anymore. The problem is reduced to a convex quadratic programming problem, which can be solved in polynomial time [17,18].

Theorem 4.1 *For all $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q - 1\}$, if $R_i^2 < \mathbf{R}^2 + \|\mathbf{c} - c_i\|^2$ and $\mathbf{R}_j^2 > \mathbf{R}^2 + \|\mathbf{c} - \mathbf{c}_j\|^2$, then $\Pi \setminus \mathbf{B} \neq \emptyset$ or $\Pi = \emptyset$.*

Proof First, we show that $\mathbf{c} \notin \Pi$. Indeed, we have for $i = 1, \ldots, p$,

$$h_i(\mathbf{c}) = \langle 2\mathbf{c} - (\mathbf{c} + c_i), \mathbf{c} - c_i \rangle + (\mathbf{R}^2 - R_i^2) = \mathbf{R}^2 + \|\mathbf{c} - c_i\|^2 - R_i^2 > 0,$$

and for $j = 1, \ldots, q - 1$,

$$-\mathbf{h}_j(\mathbf{c}) = -\langle 2\mathbf{c} - (\mathbf{c} + \mathbf{c}_j), \mathbf{c} - \mathbf{c}_j \rangle - (\mathbf{R}^2 - \mathbf{R}_j^2) = \mathbf{R}_j^2 - \mathbf{R}^2 - \|\mathbf{c} - \mathbf{c}_j\|^2 > 0,$$

using the hypothesis. Second, we prove that, if $\Pi \neq \emptyset$, then Π is unbounded. Suppose that there exists $\hat{x} \in \Pi$; we have $\mathbf{c} - \hat{x} \neq 0$. We consider the affine functions $g_i : \lambda \mapsto h_i(\hat{x} + \lambda(\mathbf{c} - \hat{x}))$ and $\mathbf{g}_i : \lambda \mapsto -\mathbf{h}_i(\hat{x} + \lambda(\mathbf{c} - \hat{x}))$ with $\lambda \in \mathbb{R}$. We have $g_i(0) \leq 0$, $g_i(1) > 0$ and $\mathbf{g}_i(0) \leq 0$, $\mathbf{g}_i(1) > 0$, so that these functions are strictly

increasing. Thus, for all negative lambda, we have $\hat{x} + \lambda(\mathbf{c} - \hat{x}) \in \Pi$. Therefore, as $\lim_{\lambda \to -\infty} \|\hat{x} + \lambda(\mathbf{c} - \hat{x})\| = +\infty$, the set $\Pi \setminus \mathbf{B}$ cannot be empty. $\qquad \square$

With only an upper bound on the number of involved balls in $\Lambda \cup V$, we have the same result.

Proposition 4.1 *If the number of constraints is less or equal to the dimension, that is $p + q - 1 \leq n$, then Π is unbounded (or empty) and the conclusion of Theorem 4.1 still holds.*

Proof We explicit the constraints as a system of linear inequalities, $Ax \leq b$, with $A \in \mathbb{R}^{(p+q-1)\times n}$ and $b \in \mathbb{R}^{p+q-1}$. If the rank of A is equal to n, the system $Ax = b - t\mathbb{I}$, with $\mathbb{I}$ the vector filled by ones, has a (unique) solution $x(t)$ and $\lim_{t \to +\infty} \|x(t)\| = \lim_{t \to +\infty} \|A^{-1}(b - t\mathbb{I})\| = +\infty$ with $Ax(t) = b - t\mathbb{I} \leq b$ for positive t. If $rank(A) < n$, the rank-nullity theorem shows that the linear subspace $Ker(A)$ is of dimension $n - rank(A) > 0$ and then $Ker(A) \neq \{0\}$. If a point $x_0 \in \mathbb{R}^n$ satisfies the inequalities, thus, $A(x_0 + ty) = Ax_0 \leq b$, with $y \in Ker(A) \setminus \{0\}$ and $t \in \mathbb{R}$. We have $\lim_{t \to +\infty} \|x_0 + ty\| = +\infty$, and the result is proven. $\qquad \square$

Remark 4.1 The conditions of Theorem 4.1 are sharp. If one of the relations is false, then there exist sets of balls Λ and V with $\#(\Lambda \cup V) = p + q = n + 2$, such that we have $\Pi \subset \mathbf{B}$. An example of such a set Π is the n-dimensional pyramid with summit near point $\mathbf{c}$ and a basis obtained with the only one hyperplane that do not satisfies the conditions of Theorem 4.1. The necessary condition of Proposition 4.1 is also not verified as $p + q = n + 2 > n + 1$.

Theorem 4.2 *If there exists $d \in \mathbb{R}^n \setminus \{0\}^n$ such that $\langle d, \mathbf{c} - c_i \rangle < 0$ and $\langle d, \mathbf{c} - \mathbf{c}_j \rangle > 0$ for all $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q-1\}$ (linear separability), then Π is unbounded or $\Pi = \emptyset$.*

Proof Suppose that there exist $\hat{x} \in \Pi$ and d satisfying the condition of the theorem. Then for all $\alpha < 0$, we have for all the constraints

$$h_i(\hat{x} + \alpha d) = 2\langle \hat{x}, \mathbf{c} - c_i \rangle + 2\alpha \langle d, \mathbf{c} - c_i \rangle - \|\mathbf{c}\|^2 + \|c_i\|^2 + (\mathbf{R}^2 - R_i^2) \leq h_i(\hat{x}) \leq 0,$$
$$\mathbf{h}_j(\hat{x} + \alpha d) = 2\langle \hat{x}, \mathbf{c} - \mathbf{c}_j \rangle + 2\alpha \langle d, \mathbf{c} - \mathbf{c}_j \rangle - \|\mathbf{c}\|^2 + \|\mathbf{c}_j\|^2 + (\mathbf{R}^2 - R_j^2) \geq \mathbf{h}_j(\hat{x}) \geq 0,$$

and $\hat{x} + \alpha d$ defines an infinite direction inside the polyhedron Π. $\qquad \square$

All these results are important for our changepoint detection algorithm (see "Appendix A") as they can considerably reduce the computation time.

4.2 The Concave QCQP Problem

If there is only one ball in the set V and then no more concave constraint in (2), we consider only one QCQP problem of type (2) and only one QP problem of type (4) ($q = 1$). We say that we solve a concave QCQP problem, because we minimize the opposite of a convex function over a set of convex constraints.

A first direct consequence is that $\Pi = \Pi^+, \Pi^- = \mathbb{R}^n$ and $\mathfrak{U} = \emptyset$. In this particular configuration, some of our results in Sects. 3 and 4.1 can be simplified.

`Feasibility:`
If $V_n(\Pi) = 0$, then $\mathcal{I} \subset \mathbf{B}$ (or $\mathcal{I} = \emptyset$).

`Quadratic programming reduction:`
(A) If $(x^-, x^+) \in \Pi \times \Pi$ is such that $\|x^- - \mathbf{c}\|^2 < \mathbf{R}^2 < \|x^+ - \mathbf{c}\|^2$, then $\mathcal{I} \setminus \mathbf{B} \neq \emptyset$;
(B) if $\mathbf{R}^2 \leq \|x_{min}^* - \mathbf{c}\|^2$, then $\mathcal{I} \cap \mathbf{S} = \emptyset$;
(C) if $\|x_{max}^* - \mathbf{c}\|^2 \leq \mathbf{R}^2$, then $\mathcal{I} \subset \mathbf{B}$.

Proof Only the case (C) has to be proven. Using Lemma 3.1, we have $\mathcal{I} \setminus \mathbf{B} = \cap_{i=1}^p (B_i \setminus \mathbf{B}) \subset \cap_{i=1}^p (H_i \setminus \mathbf{B}) = \Pi \setminus \mathbf{B}$. However, $\Pi \setminus \mathbf{B} = \emptyset$ and then $\mathcal{I} \subset \mathbf{B}$. □

`Convex optimization reduction:`
if $R_i^2 < \mathbf{R}^2 + \|\mathbf{c} - c_i\|^2$, for all $i \in \{1, \ldots, p\}$, then $\Pi \setminus \mathbf{B} \neq \emptyset$ or $\Pi = \emptyset$.

5 Simulations

Algorithm 2 describes our new procedure based on equivalent QP problems. It is made of two steps: the detection of an intersection between $\mathcal{I}$ and $\partial \mathcal{U}$ using QP problems (see Sect. 3) and the test $\mathcal{I} \subset \mathcal{U}$ with a unique point (if $\mathcal{I} \cap \partial \mathcal{U} = \emptyset$). We chose to determine x_{min}^* and x_{max}^* for each QP problem but, in fact, we only need points x^- and x^+, as shown by Theorem 3.1.


Algorithm 2 QP-based decision algorithm

1: **procedure** ISINCLUDED(Λ, V)
2: $response \leftarrow true, j \leftarrow 1$
3: **while** $j < q + 1$ **do**
4:     $x_{min}^*(j) = \text{Argmin} \{P_1(j)\}$ ▷ We solve the QP problem $P_1(j)$ for min
5:     **if** $\|x_{min}^*(j) - \mathbf{c}_j\|^2 < \mathbf{R}_j^2$ **then**
6:         $x_{max}^*(j) = \text{Argmax} \{P_1(j)\}$ ▷ We solve the QP problem $P_1(j)$ for max
7:         **if** $\mathbf{R}_j^2 < \|x_{max}^*(j) - \mathbf{c}_j\|^2$ **then**
8:             $response \leftarrow false, j \leftarrow q$
9:         **end if**
10:    **end if**
11:    $j \leftarrow j + 1$
12: **end while**
13: **if** $response = true$ **then**
14:    Find $\tilde{x} \in \mathcal{I}$
15:    **if** $\tilde{x} \notin \mathcal{U}$ **then**
16:        $response \leftarrow false$
17:    **end if**
18: **end if**
19: **return** $response$

Our goal is to compare the performances of this algorithm against Algorithm 1 on two aspects: the exactness of the result and their computational efficiency. We will also illustrate the polynomial time complexity of our new algorithm by increasing the dimension n at fixed p and q.
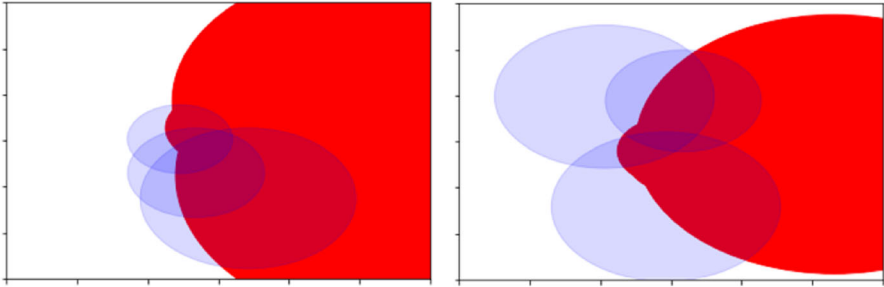
Fig. 3 Two examples of simulated data with $p = q = 3$ in dimension 2. On the left, $\mathcal{I} \not\subset \mathcal{U}$, on the right $\mathcal{I} \subset \mathcal{U}$. The opaque red disks are the balls of the union set V. The transparent blue disks are the balls of the intersection set Λ

We have implemented[1] Algorithm 1 in Python using the recent (2017) *suggest and improve* method for nonconvex QCQP [12] based on packages 'cvxpy' [21] and its extension 'qcqp'. This allows us to use state-of-the-art methods on Python, more elaborated than interior point one, which is not adapted to our problem (see Fig. 1). The *suggest* method is set to *random* and the *improve* step to *ADMM* (alternating directions method of multipliers [22]) followed by a coordinate descent step. This is the most efficient combination of methods (empirically speaking) and is used in some examples given in the package.

The center points for balls are randomly generated with a normal distribution centered on zero with standard deviation $\sigma = 10$, and their radius is the distance to zero plus a quantity $\epsilon = 5$ for intersection ball and 2ϵ for union balls. We also verify that the obtained balls intersect each other as in (1), if not, we simulate new balls. Examples of simulated data are shown in Fig. 3.

In Table 1, we generate 100 examples for each proposed configuration. We first consider examples with $\mathcal{I} \not\subset \mathcal{U}$. In case ($n = 2$, $p = 3$), we compare the results of the two algorithms with the 2d graphical representation of the balls problem to confirm the exactness of our method.

Algorithm 2 always finds the exact result in dimension 2, and we get the value 100. For higher dimensions, we know the result of Algorithm 2 to be exact (as shown by theoretical results of Sect. 3) and count the number of identical results between the two algorithms to get the number of true results for Algorithm 1. Algorithm 1 often fails, in particular when the non-inclusion is difficult to visually detect (obtained on a small volume). This behavior is a consequence of our simulations: we generated data with a unique simulation procedure, so that, for increasing q it becomes harder to generate examples without inclusion. An increasing dimension n gives more space to get a non-inclusion and facilitate the detection of $\mathcal{I} \not\subset \mathcal{U}$ for Algorithm 1.

In the case of the inclusion $\mathcal{I} \subset \mathcal{U}$, we notice that Algorithm 1 always finds the right answer in our simulations. Interestingly, the resolution of the unique problem $P_0(q)$ often fails, unlike the non-inclusion case.

---

[1] The Python code is available on my github 'vrunge' in repository called 'QP_vs_QCQP' https://github.com/vrunge/QP_vs_QCQP/blob/master/simulations.ipynb.

Table 1 Number of exact results over 100 simulations

$\mathcal{I} \not\subset \mathcal{U}$	$n = 2, p = 3$			$p = 3, q = 3$			$p = 5, q = 5$
	$q = 1$	$q = 2$	$q = 3$	$n = 3$	$n = 5$	$n = 10$	$n = 10$
Algorithm 1	83	65	48	57	63	79	65
$P_0(q)$	83	80	87	91	91	90	92
Algorithm 2	100	100	100	.	.	.	.
$\mathcal{I} \subset \mathcal{U}$							
Algorithm 1	100	100	100	100	100	100	100
$P_0(q)$	100	92	81	75	75	83	77

Table 2 Comparison of execution time with $p = q = 3$ and varying dimension n

i	$n = 5$		$n = 10$		$n = 20$	
	time QP	time QCQP	time QP	time QCQP	time QP	time QCQP
1	0.00411	2.013	0.00448	2.667	0.00464	2.923
2	0.00280	1.213	0.00261	2.553	0.00298	1.732
3	0.00260	1.034	0.00262	2.358	0.00305	2.570
4	0.00274	1.044	0.00262	2.133	0.00290	15.10
5	0.00279	1.967	0.00272	2.098	0.00306	4.843
6	0.00304	1.039	0.00278	1.402	0.00294	5.856
7	0.00267	1.639	0.00267	1.766	0.00288	48.80
8	0.00261	1.070	0.00277	8.500	0.00296	2.780
9	0.00547	0.832	0.00275	1.803	0.00375	2.913
10	0.00263	0.564	0.00286	1.663	0.00666	2.964
	RATIO mean time QCQP by mean time QP					
	394.6		933.0		2526	

In Table 2, we highlight the efficiency of our new algorithm, compared with the Python package 'qcqp' based on nonconvex QCQP methods. Notice that improvements in the code for Algorithm 2 are still possible with a direct implementation of the QP problem stopping as soon as we get x^- or x^+ and in the vertex enumeration solver (we simply used package cdd[2] [23]). We have made 10 simulations ($i = 1, \ldots, 10$) for 3 configurations ($n = 5, 10, 20$) with $p = q = 3$ in case $\mathcal{I} \not\subset \mathcal{U}$ (simpler to obtain in our ball generation algorithm with increasing dimension n). Our results show for Algorithm 1 a computational time complexity of order $O(10^3)$ slower than for Algorithm 2 and a higher time variability between iterates.

An empirical polynomial time complexity is confirmed by our simulations as shown in Fig. 4. Simulations have been performed with $\mathcal{I} \not\subset \mathcal{U}$ for an increasing n, so that, in these simulations, we only solve a unique QP problem for maximum and minimum

---

[2] website of the Python package http://pycddlib.readthedocs.io/en/latest/.
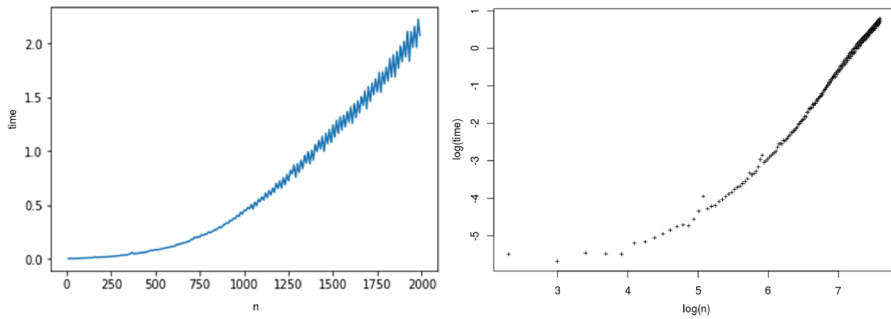
Fig. 4 Left: $p = q = 3$ and $n = 10, 20, 30, \ldots, 2000$. Each datapoint is the mean over 10 identical simulations in case $\mathcal{I} \not\subset \mathcal{U}$. A unique QP problem is solved for each simulation. Right: in the log-log scale, the curve stabilizes, corroborating a polynomial time complexity of type an^b. A linear model returns a coefficient of determination of 0.935 with $b \approx 1.76$

to detect an intersection. With a linear regression based on the model $t = an^b$, we get a power of about $b = 1.76$ with $R^2 = 0.935$ in the logarithmic scale.

Conclusions

The geometric question of the cover of an intersection of balls by an union of other balls has been addressed using optimization tools. The collection of nonconvex quadratically constrained quadratic programming problems has been transformed into a collection of quadratic optimization problems: the minimum and the maximum distances between a point and a Voronoi-like polyhedron has to be found for each problem. The maximum problem can be handled efficiently by vertex enumeration. If simple conditions are satisfied, the polyhedron is unbounded, the maximization problem has not to be considered anymore, and the complexity is known to be in polynomial time. Simulations show that state-of-the-art nonconvex QCQP algorithms, developed in Python, often fail the decision test and are computationally greedy. Our new method never fails (is exact) and efficiently finds the solution with a time complexity of order $O(10^3)$ smaller in our simulation study. In a further work, we will apply this method to the efficient implementation of a multidimensional changepoint detection algorithm based on pruned dynamic programming.

Appendices

A Motivation

Our covering problem has a direct application for the implementation of the pruned penalized[3] dynamic programming algorithm for changepoint detection in a multidimensional setting [25–27]. This problem consists in finding the optimal changepoint within the set $\mathbb{S}_m = \{\tau = (\tau_1, \ldots, \tau_k) \in \mathbb{N}^k \mid 1 = \tau_0 < \tau_1 < \cdots < \tau_k < m + 1 = \tau_{k+1}\}$ such that we minimize a quadratic (for Gaussian modelization) penalized cost (by $\beta > 0$):

$$Q_m = \min_{\tau \in \mathbb{S}_m} \left[\sum_{i=0}^{k} \{ \mathcal{C}(y_{\tau_i:\tau_{i+1}-1}) + \beta \} \right],$$

$$\text{with} \quad \mathcal{C}(y_{\tau_i:\tau_{i+1}-1}) = \min_{x \in \mathbb{R}^n} \sum_{j=\tau_i}^{\tau_{i+1}-1} \sum_{k=1}^{n} (x_k - y_{jk})^2,$$

and $(y_1, \ldots, y_m)^T \in (\mathbb{R}^n)^m$ the data to segment. Using a dynamic programming procedure, we build the recursion $Q_{t+1}(x) = \min\{Q_t(x), m_t + \beta\} + \sum_{k=1}^{n}(x_k - y_{(t+1)k})^2$ with $m_t = \min_x Q_t(x)$ and the initialization $Q_0(x) = 0$. We solve this recursion iteratively from $t = 0$ to $t = m - 1$.

At each t, the recursive function $Q_t(\cdot)$ is a piecewise quadratic function defined on t non-overlapping regions $R_t^1, \ldots, R_t^t \subset \mathbb{R}^n$, for which each quadratics is active on a set R_t^i of type "$\mathcal{I} \setminus \mathcal{U}$". Precisely,

$$R_t^1 = \cap_{i=1}^{t} B_i^1,$$
$$R_t^2 = \cap_{i=2}^{t} B_i^2 \setminus B_1^1,$$
$$R_t^3 = \cap_{i=3}^{t} B_i^3 \setminus (B_2^1 \cup B_2^2),$$
$$\vdots$$
$$R_t^t = R_t^t \setminus (\cup_{j=1}^{t-1} B_{t-1}^j).$$

where all the "B_l^k" sets designate balls determined by the data $(y_k, \ldots, y_l)$. At the next iteration $t + 1$, each R_t^i is intersected by a new ball (that is, we add a ball in each $\mathcal{I}$) and the set R_{t+1}^{t+1} is created.

In order to get the global minimum m_t, we compare the minima of all present quadratics. To speed-up the procedure, it is worthwhile to search for vanishing sets, that is to detect efficiently the emptiness of sets $R_t^1, \ldots, R_t^t$. In fact, once a set is proved to be empty, we do not need to consider its minimum anymore at any further iteration. This method is called *pruning* in the changepoint detection literature.

---

[3] It would also work for (non-penalized) segment neighborhood method [24].

In a paper in preparation, we will compare this exact method to heuristic approaches in order to build a fast algorithm for an application to genomic data. Moreover, our methods can be extended to other distributions of the natural exponential family.

B An Alternative Method

In some applications, as for example in "Appendix A," we need to test sequentially the covering. This means that we add at each iteration a new ball in the intersection $\mathcal{I}$. For large q, it could be expensive to solve q QP problems at each iteration for minimum and maximum. Another approach consists in detecting an intersection between $\mathcal{I} \setminus \mathcal{U}$ and $\mathbb{S}$ (as soon as $\#\mathcal{I} > 0$), where $\mathbb{S}$ is the frontier of the new ball $\mathbb{B}$ (with center c and radius R) to add in $\mathcal{I}$. The problem is now centered on the "newest" ball of set Λ rather than the successive balls of V. Thus, a unique QP problem centered on the ball $\mathbb{B}$ is built and we get a result similar to Theorem 3.1:

Theorem B.1 *If $V_n(\Pi) > 0$, we have the following results.*

(a) *If there exists $(x^-, x^+) \in \Pi \times \Pi$ such that $\|x^- - c\|^2 < R^2 < \|x^+ - c\|^2$, then $(\mathbb{B} \cap \mathcal{I}) \setminus \mathcal{U} \neq \emptyset$ and $V_n((\mathbb{B} \cap \mathcal{I}) \setminus \mathcal{U}) > 0$;*
(b) *if $R^2 \leq \|x_{min}^* - c\|^2$ or $\|x_{max}^* - c\|^2 \leq R^2$, then $\mathbb{S} \cap (\mathcal{I} \setminus \mathcal{U}) = \emptyset$.*

In case (b), we can conclude knowing a point $\tilde{x}_k$ for each connected component C_k $(k = 1, \ldots, K)$ of $\mathcal{I} \setminus \mathcal{U}$ and testing $\tilde{x}_k \in \mathbb{B}$. If always false, we get $(\mathbb{B} \cap \mathcal{I}) \setminus \mathcal{U} = \emptyset$. The drawback of this approach is the necessity to know the number of connected components in $\mathcal{I} \setminus \mathcal{U}$. However, with the conditions of Theorem 4.1 the set $\mathcal{I} \setminus \mathcal{U}$ is connected (see the proof) and a unique point in $\mathcal{I} \setminus \mathcal{U}$ is required ($K = 1$).
Proposition 3.2 remains the same with $\mathbb{B}$ instead of **B**. The detection of the inclusion $\mathcal{I} \subset \mathbb{B}$ is now made possible and is important for sequential problems, since in this case, the ball $\mathbb{B}$ is not added to the set Λ.

References

1. Baker, M.J.C.: A spherical helly-type theorem. Pac. J. Math. **23**(1), 1–3 (1967)
2. Maehara, H.: Helly-type theorems for spheres. Discrete Comput. Geom. **4**(3), 279–285 (1989). https://doi.org/10.1007/BF02187730
3. Avis, D., Bhattacharya, B.K., Imai, H.: Computing the volume of the union of spheres. Vis. Comput. **3**(6), 323–328 (1988). https://doi.org/10.1007/BF01901190
4. Cazals, F., Kanhere, H., Loriot, S.: Computing the volume of a union of balls: a certified algorithm. ACM Trans. Math. Softw. **38**(1), 3:1–3:20 (2011). https://doi.org/10.1145/2049662.2049665
5. Chkhartishvili, L.S.: Volume of the intersection of three spheres. Math. Notes **69**(3), 421–428 (2001). https://doi.org/10.1023/A:1010295711303
6. Böröczky, K.: Finite Packing and Covering, vol. 154. Cambridge University Press, Cambridge (2004)
7. Glazyrin, A.: Covering a ball by smaller balls. Discrete Comput. Geom. (2018). https://doi.org/10.1007/s00454-018-0010-4
8. Verger-Gaugry, J.L.: Covering a ball with smaller equal balls in n. Discrete Comput. Geom. **33**(1), 143–155 (2005). https://doi.org/10.1007/s00454-004-2916-2
9. Nguyen, T., Boissonnat, J., Falzon, F., Knauer, C.: A disk-covering problem with application in optical interferometry. CoRR arXiv:abs/cs/0612026 (2006). URL http://arxiv.org/abs/cs/0612026

10. Gershman, A.B., Sidiropoulos, N.D., Shahbazpanahi, S., Bengtsson, M., Ottersten, B.: Convex optimization-based beamforming. IEEE Signal Process. Mag. **27**(3), 62–75 (2010)
11. Huang, Y., Palomar, D.P.: Randomized algorithms for optimal solutions of double-sided qcqp with applications in signal processing. IEEE Trans. Signal Process. **62**(5), 1093–1108 (2014). https://doi.org/10.1109/TSP.2013.2297683
12. Park, J., Boyd, S.: General heuristics for nonconvex quadratically constrained quadratic programming. ArXiv e-prints (2017)
13. Luo, Q.Z., Ma, K.W., So, C.A.M., Ye, Y., Zhang, S.: Semidefinite relaxation of quadratic optimization problems. IEEE Signal Process. Mag. **27**(3), 20–34 (2010). https://doi.org/10.1109/MSP.2010.936019
14. Anstreicher, M.: On convex relaxations for quadratically constrained quadratic programming. Math. Program. **136**(2), 233–251 (2012). https://doi.org/10.1007/s10107-012-0602-3
15. Mehanna, O., Huang, K., Gopalakrishnan, B., Konar, A., Sidiropoulos, N.D.: Feasible point pursuit and successive approximation of non-convex qcqps. IEEE Signal Process. Lett. **22**(7), 804–808 (2015). https://doi.org/10.1109/LSP.2014.2370033
16. Scutari, G., Facchinei, F., Lampariello, L.: Parallel and distributed methods for constrained nonconvex optimization. part i: theory. IEEE Trans. Signal Process. **65**(8), 1929–1944 (2017). https://doi.org/10.1109/TSP.2016.2637317
17. Kozlov, M., Tarasov, S., Khachiyan, L.: The polynomial solvability of convex quadratic programming. USSR Comput. Math. Math. Phys. **20**(5), 223–228 (1980). https://doi.org/10.1007/s10107-012-0602-3
18. Ye, Y., Tse, E.: An extension of karmarkar's projective algorithm for convex quadratic programming. Math. Programm. **44**(1), 157–179 (1989). https://doi.org/10.1007/BF01587086
19. Avis, D., Fukuda, K.: A pivoting algorithm for convex hulls and vertex enumeration of arrangements and polyhedra. Discrete Comput. Geom. **8**(3), 295–313 (1992). https://doi.org/10.1007/BF02293050
20. Freund, R.M., Orlin, J.B.: On the complexity of four polyhedral set containment problems. Math. Program. **33**(2), 139–145 (1985). https://doi.org/10.1007/BF01582241
21. Diamond, S., Boyd, S.: CVXPY: a Python-embedded modeling language for convex optimization. J. Mach. Learn. Res. **17**(83), 1–5 (2016)
22. Boyd, S., Parikh, N., Chu, E., Peleato, B., Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. **3**(1), 1–122 (2011). https://doi.org/10.1561/2200000016
23. Fukuda, K., Prodon, A.: Double description method revisited. In: Deza, M., Euler, R., Manoussakis, I. (eds.) Combinatorics and Computer Science, pp. 91–111. Springer, Berlin (1996)
24. Auger, I.E., Lawrence, C.E.: Algorithms for the optimal identification of segment neighborhoods. Bull. Math. Biol. **51**(1), 39–54 (1989). https://doi.org/10.1007/BF02458835
25. Fearnhead, P., Rigaill, G.: Changepoint detection in the presence of outliers. J. Am. Stat. Assoc. (2017). https://doi.org/10.1080/01621459.2017.1385466
26. Maidstone, R., Hocking, T., Rigaill, G., Fearnhead, P.: On optimal multiple changepoint algorithms for large data. Stat. Comput. **27**(2), 519–533 (2017). https://doi.org/10.1007/s11222-016-9636-3
27. Rigaill, G.: A pruned dynamic programming algorithm to recover the best segmentations with 1 to kmax change-points. J. Soc. Franc. Stat. **156**(4), 180–205 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.