INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Minimal enclosing parallelepiped in 3D

Frédéric Vivien, Nicolas Wicker

No 4685

Décembre 2002

THÈME 1

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--4685--FR+ENG

Minimal enclosing parallelepiped in 3D

Frédéric Vivien, Nicolas Wicker

Thème 1 — Réseaux et systèmes
Projet ReMaP

Rapport de recherche n° 4685 — Décembre 2002 — 23 pages

Abstract: We investigate the problem of finding a minimal volume parallelepiped enclosing a given set of n three-dimensional points. We give two mathematical properties of these parallelepipeds, from which we derive two algorithms of theoretical complexity $O(n^6)$. Experiments show that in practice our quickest algorithm runs in $O(n^2)$ (at least for $n \leq 10^5$). We also present our application in structural biology.

Key-words: Algorithmic geometry, parallelepiped, bioinformatic.

(Résumé : tsvp)

This text is also available as a research report of the Laboratoire de l'Informatique du Parallélisme `http://www.ens-lyon.fr/LIP`.

Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN (France)
Téléphone : 04 76 61 52 00 - International : +33 4 76 61 52 00
Télécopie : 04 76 61 52 52 - International : +33 4 76 61 52 52

Minimal enclosing parallelepiped in 3D

Résumé : Nous étudions le problème de la recherche d'un parallélépipède de volume minimal englobant un ensemble donné de n points d'un espace de dimension trois. Nous démontrons deux propriétés mathématiques de ces parallélépipèdes à partir desquelles nous élaborons deux algorithmes de complexité théorique en $O(n^6)$. Nos expériences montrent que la complexité en pratique de notre algorithme le plus rapide est en $O(n^2)$ (au moins quand n est inférieur à 10^5). Nous présentons également notre application en biologie structurale.

Mots-clé : Géométrie algorithmique, parallélépipèdes, bioinformatique.

1 Introduction

It is sometimes useful to circumscribe a complex three-dimensional shape with a simpler shape, of minimum volume. Solutions for this problem are known if one is looking for the minimal volume enclosing ball or ellipsoid [10], cylinder [5], tetrahedron [12], or rectangular box [2]. Our original motivation was to approximate the surface of a protein with a set of regular shapes in the hope of finding some "outstanding faces" of the protein, e.g. responsible of interactions with other proteins. From biological considerations, parallelepipeds seemed more suitable for our problem. So, in this paper we show how to compute a parallelepiped of minimal volume enclosing a three-dimensional shape or set of points. Our algorithms rely on mathematical properties inspired by the properties satisfied in the plane by the minimal enclosing parallelogram [7–9].

In Section 2 we prove two mathematical properties of minimal enclosing parallelepipeds. From these properties, we derive two algorithms in Section 3. In Section 4, we report the experiments we performed on these algorithms. Finally, in Section 5, we give an insight of our biological motivation: we apply our technique to a protein and discuss the result.

2 Mathematical properties

First, we remark that the minimal volume parallelepiped enclosing a set S of points is the minimal volume parallelepiped enclosing the convex hull of S as the convex hull of S is the smallest convex enclosing S. Then, the first theorem states that each pair of opposite faces of the minimal enclosing parallelepiped must flush a face or two edges of the convex hull (and not just a face as in 2D). In this paper we never consider degenerated sets of points, i.e. included in a plane.

Theorem 1 *For any set of points of convex hull $\mathcal{C}$ there exists a minimal enclosing parallelepiped $\mathcal{P}$ such that, for any pair of opposite faces of $\mathcal{P}$, either one of the faces contains a face of $\mathcal{C}$ or both faces contain an edge of $\mathcal{C}$ and the two edges are not parallel.*

Proof We consider a set of points of convex hull $\mathcal{C}$ and one of its minimal enclosing parallelepiped $\mathcal{P}$. Any face $\mathcal{F}$ of $\mathcal{P}$ contains at least one vertex of $\mathcal{C}$: otherwise it would be possible to move $\mathcal{F}$ closer to its opposite face to obtain an enclosing parallelogram of smaller volume.

To prove the theorem, we suppose that $\mathcal{P}$ does not satisfy the property stated by the theorem and we show that we can build an enclosing parallelogram satisfying the property and at least as small as $\mathcal{P}$. As, by hypothesis, $\mathcal{P}$ does not satisfy the property stated by the theorem, there exist two opposite faces $\mathcal{F}_1$ and $\mathcal{F}_2$ of $\mathcal{P}$ such that none of them contain a face of $\mathcal{C}$ and if both contain an edge of $\mathcal{C}$, both edges are parallel. We denote by $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) the plane containing $\mathcal{F}_1$ (resp. $\mathcal{F}_2$).

A parallelepiped is defined by its eight vertices. It is also defined by the three pairs of parallel planes that contain its faces. We will call these planes the *supporting planes*. Let us consider a pair of supporting planes p_1 and p_2, i.e. two supporting planes corresponding to opposite faces of $\mathcal{P}$. We take two parallel lines, d_1 and d_2, the first included in p_1 and the second in p_2. We rotate p_1 around d_1 and p_2 around d_2 with a same angle. This way, we obtain a new pair of parallel planes which defines, with the four remaining supporting planes of $\mathcal{P}$, a new parallelepiped. This new parallelepiped may or may not be an enclosing parallelepiped for $\mathcal{C}$. We say that we have rotated the pair of supporting planes $\{p_1, p_2\}$.

We first study the freedom we have to rotate the pair of supporting planes $\{\mathcal{P}_1, \mathcal{P}_2\}$ while the obtained parallelepiped remains enclosing for $\mathcal{C}$.

The possibility to rotate some supporting planes

We consider the number n_v of vertices of $\mathcal{C}$ belonging to either of the two faces $\mathcal{F}_1$ and $\mathcal{F}_2$:

$\boldsymbol{n_v \geq 5}$**:** one of the two faces contains at least three vertices and thus a face of $\mathcal{C}$. This is impossible by definition of $\mathcal{F}_1$ and $\mathcal{F}_2$.

$\boldsymbol{n_v = 4}$**:** by definition of $\mathcal{F}_1$ and $\mathcal{F}_2$, both faces contain an edge of $\mathcal{C}$ and these two edges are parallel. We denote by d_1 (respectively d_2) the line of $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) containing the edge of $\mathcal{C} \cap \mathcal{F}_1$ (resp. $\mathcal{C} \cap \mathcal{F}_2$). Then one can (slightly) rotate, in any direction, $\mathcal{P}_1$ and $\mathcal{P}_2$ of a same angle around d_1 and d_2 while transforming $\mathcal{P}$ into another parallelepiped enclosing $\mathcal{C}$, as long as the angle of the rotation remains small. Indeed, we can rotate the pair of supporting planes $\{\mathcal{P}_1, \mathcal{P}_2\}$ until one of the rotated planes touch a new vertex of $\mathcal{C}$.

$\boldsymbol{n_v = 3}$**:** because of our hypothesis, one face contains a single vertex of $\mathcal{C}$ and the other one an edge of $\mathcal{C}$. Without any loss of generality, we denote by $\mathcal{F}_1$ the face containing the edge. We define d_1 as previously and d_2 as the line of $\mathcal{P}_2$ containing $\mathcal{C} \cap \mathcal{F}_2$ and parallel to d_1. Then one can (slightly) rotate, in any direction, $\mathcal{P}_1$ and $\mathcal{P}_2$ of a same angle around d_1 and d_2, under the same conditions than previously.

$\boldsymbol{n_v = 2}$**:** each face contains exactly one vertex of $\mathcal{C}$. We randomly peak any vector $\boldsymbol{v}$ in $\mathcal{P}_1$ to define the direction of d_1 and d_2: d_1 (resp. d_2) is then the line of $\mathcal{P}_1$ (resp. $\mathcal{P}_2$) parallel to $\boldsymbol{v}$ and containing $\mathcal{C} \cap \mathcal{F}_1$ (resp. $\mathcal{C} \cap \mathcal{F}_2$). Then one can (slightly) rotate, in any direction, $\mathcal{P}_1$ and $\mathcal{P}_2$ of a same angle around d_1 and d_2, under the same conditions than previously.

Building an enclosing parallelogram smaller than $\mathcal{P}$

From what precedes, whatever the case, one can (slightly) rotate, in any direction, $\mathcal{P}_1$ and $\mathcal{P}_2$ of a same angle around d_1 and d_2 while transforming $\mathcal{P}$ into another parallelepiped, $\mathcal{P}'$, enclosing $\mathcal{C}$. We now compute the volume of the new parallelepiped $\mathcal{P}'$. In the following, given two points I and J, $\boldsymbol{IJ}$ denotes the vector from point I to point J and IJ the algebraic measure.

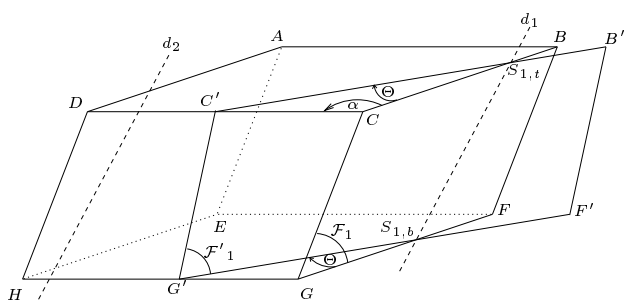

Figure 1: Original parallelepiped and the rotation.

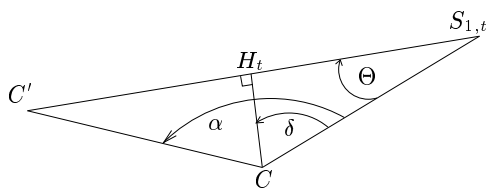

Figure 2: Detail of Figure 1.

Figure 1 shows the original parallelepiped and the new face $\mathcal{F}_1^{'} = (B'F'G'C')$ obtained from the rotation of $\mathcal{P}_1$ by an angle of Θ around d_1. We use the notations defined on Figure 1. $S_{1,t}$ (resp. $S_{1,b}$) is the intersection of d_1 with the line (BC) (resp. (FG)). In order to ease the computations, we measure the rotation around d_1 and d_2 not by an angle measured in a plane orthogonal to d_1 but in the plane $(ABCD)$. Θ is the angle defined by the vectors $\boldsymbol{S_{1,t}C}$ and $\boldsymbol{S_{1,t}C'}$.

The volume of the parallelepiped $\mathcal{P}$ is equal to: $\mathrm{vol}(\mathcal{P}) = |(\boldsymbol{CB} \wedge \boldsymbol{CD}).\boldsymbol{CG}|$. The volume of $\mathcal{P}'$ is equal to: $\mathrm{vol}(\mathcal{P}') = |(\boldsymbol{C'B'} \wedge \boldsymbol{C'D'}).\boldsymbol{C'G'}|$. To explicit the value of $\mathrm{vol}(\mathcal{P}')$, we need to explicit the values of $\boldsymbol{C'B'}$, $\boldsymbol{C'D'}$, and $\boldsymbol{C'G'}$. We start with $\boldsymbol{C'D'}$.

The value of $\boldsymbol{C'D'}$. $\boldsymbol{C'D'} = \boldsymbol{C'C} + \boldsymbol{CD} + \boldsymbol{DD'}$. To compute the value of $\boldsymbol{C'D'}$ we focus on Figure 2, which is a magnification of Figure 1. We denote by α the angle defined by the vectors $\boldsymbol{CB}$ and $\boldsymbol{CD}$. Then $\cos(\alpha - \delta) = \frac{CH_t}{CC'}$ and $\cos(-\delta) = \frac{CH_t}{CS_{1,t}}$. As the sum of the angles in a triangle is equal to π, $\delta = \frac{\pi}{2} + \Theta$, and:

$$CC' = \frac{\cos(\delta)}{\cos(\alpha - \delta)} CS_{1,t} = \frac{\cos(\frac{\pi}{2} + \Theta)}{\cos((\alpha - \Theta) - \frac{\pi}{2})} CS_{1,t} = -\frac{\sin(\Theta)}{\sin(\alpha - \Theta)} CS_{1,t},$$

and, $\boldsymbol{u_{CD}}$ denoting the unitary vector of same direction and orientation than $\boldsymbol{CD}$,

$$\boldsymbol{CC'} = -\frac{\sin(\Theta)}{\sin(\alpha - \Theta)} CS_{1,t}.\boldsymbol{u_{CD}}.$$

Symmetrically, we have for $\boldsymbol{DD'}$ ($S_{2,t}$ (resp. $S_{2,b}$) being the intersection of d_2 with the line (DA) (resp. (HE))): $\boldsymbol{DD'} = -\frac{\sin(\Theta)}{\sin(\alpha - \Theta)} DS_{2,t}.\boldsymbol{u_{CD}}$. Gathering these two results, we obtain:

$$\boldsymbol{C'D'} = \boldsymbol{C'C} + \boldsymbol{CD} + \boldsymbol{DD'} = \boldsymbol{CD} + \frac{\sin(\Theta)}{\sin(\alpha - \Theta)} (CS_{1,t} - DS_{2,t}).\boldsymbol{u_{CD}}.$$

The values of $\boldsymbol{C'B'}$ and $\boldsymbol{C'G'}$. $\boldsymbol{C'B'} = \boldsymbol{C'C} + \boldsymbol{CB} + \boldsymbol{BB'}$. Thus, as $\boldsymbol{C'C}$ and $\boldsymbol{BB'}$ are parallel to $\boldsymbol{u_{CD}}$, there exists a value x such that: $\boldsymbol{C'B'} = \boldsymbol{CB} + x.\boldsymbol{u_{CD}}$. Symmetrically, there exists a value y such that $\boldsymbol{C'G'} = \boldsymbol{CG} + y.\boldsymbol{u_{CD}}$.

The volume of $\mathcal{P}'$. Collecting the previous results, we have:

$$\begin{aligned}(\boldsymbol{C'B'} \wedge \boldsymbol{C'D'}).\boldsymbol{C'G'} &= ((\boldsymbol{CB} + x.\boldsymbol{u_{CD}}) \\ &\wedge(\boldsymbol{CD} + \tfrac{\sin(\Theta)}{\sin(\alpha-\Theta)}(CS_{1,t} - DS_{2,t}).\boldsymbol{u_{CD}}).(\boldsymbol{CG} + y.\boldsymbol{u_{CD}}) \\ &= \boldsymbol{CB} \wedge \boldsymbol{CD}.\boldsymbol{CG} \\ &\quad + \tfrac{\sin(\Theta)}{\sin(\alpha-\Theta)}(CS_{1,t} - DS_{2,t})(\boldsymbol{CB} \wedge \boldsymbol{u_{CD}}.\boldsymbol{CG}) \\ &= \left(1 + \tfrac{\sin(\Theta)}{\sin(\alpha-\Theta)}\tfrac{(CS_{1,t}-DS_{2,t})}{||\boldsymbol{CD}||}\right) \boldsymbol{CB} \wedge \boldsymbol{CD}.\boldsymbol{CG}.\end{aligned}$$

Therefore:

$$\mathrm{vol}(\mathcal{P}') = \left|1 + \frac{\sin(\Theta)}{\sin(\alpha - \Theta)}\frac{(CS_{1,t} - DS_{2,t})}{||\boldsymbol{CD}||}\right| \mathrm{vol}(\mathcal{P}). \tag{1}$$

We have two cases to consider, depending whether $(CS_{1,t} - DS_{2,t})$ is null:

1. $CS_{1,t} - DS_{2,t} \neq 0$. $\sin(\alpha)$ is obviously non null, knowing the definition of α. For very small values of Θ, $\sin(\alpha - \Theta)$ has the same sign than $\sin(\alpha)$. As we can chose Θ to be either strictly negative or strictly positive (see the discussion above), we chose for Θ a very small value such that $\Theta \sin(\alpha)(CS_{1,t} - DS_{2,t}) < 0$. Then $\mathrm{vol}(\mathcal{P}') < \mathrm{vol}(\mathcal{P})$ and we have built an enclosing parallelogram of (strictly) smaller volume.

2. $CS_{1,t} - DS_{2,t} = 0$. Then $\mathcal{P}$ and $\mathcal{P}'$ are two enclosing parallelepipeds of same volume (whatever the value of Θ). We take for Θ the largest value possible. The two new faces $\mathcal{F}_1'$ and $\mathcal{F}_2'$ contain by definition of d_1 and d_2 all the points of $\mathcal{F}_1$ and $\mathcal{F}_2$ belonging to $\mathcal{C}$. Because of the maximality of Θ, $\mathcal{F}_1' \cup \mathcal{F}_2'$ contains at least one more point of $\mathcal{C}$ (and thus one more vertex of $\mathcal{C}$) than $\mathcal{F}_1 \cup \mathcal{F}_2$. If $\mathcal{P}'$ satisfies the property stated by the theorem, we are happy. Otherwise, we apply to $\mathcal{P}'$ the process we have applied to $\mathcal{P}$ to obtain $\mathcal{P}'$. This way we obtain a new enclosing parallelogram $\mathcal{P}''$. As the number of vertices n_v of $(\mathcal{F}_1 \cup \mathcal{F}_2) \cap \mathcal{C}$ is strictly increasing with this process, we shortly end up with a parallelepiped of volume at most equal to $\mathrm{vol}(\mathcal{P})$ and which satisfies the property stated by the theorem. Indeed, any parallelepiped with $n_v \geq 5$ satisfies this property (as we have shown above).

In both cases we obtain, may be after a few iterations, a parallelepiped enclosing $\mathcal{S}$, satisfying the desired property, and whose volume is less than or equal to the volume of $\mathcal{P}$. ■

Theorem 2 *Let $\mathcal{S}$ be a set of points and $\mathcal{C}$ its convex hull. Let $\mathcal{P}$ be a minimal volume parallelepiped enclosing $\mathcal{S}$ and which satisfies the property stated by Theorem 1. Let $\mathcal{F}_1$ and $\mathcal{F}_2$ be two opposite faces of $\mathcal{P}$. Then, the projection of $\mathcal{F}_1 \cap \mathcal{C}$ on $\mathcal{F}_2$ along the other faces of $\mathcal{P}$ has a non-null intersection with $\mathcal{F}_2 \cap \mathcal{C}$.*

Proof We prove this result by contradiction. Thus we suppose that, $\mathcal{P}$, a minimal volume enclosing parallelepiped which satisfies the property stated by Theorem 1, does not satisfy the property stated by Theorem 2. Then we show that we can build an enclosing parallelepiped of strictly smaller volume. The proof rely on a careful study of Equation 1. First, we remark that, because of its definition, the angle α has a value strictly between 0 and π. Therefore $\sin(\alpha)$ is always (strictly) positive. Θ will be chosen small. Thus Θ and $\sin(\Theta)$ will have the same sign. Also $\sin(\alpha - \Theta)$ and $\sin(\alpha)$ will have the same sign. If $CS_{1,t} - DS_{2,t}$ is not null and if $CS_{1,t} - DS_{2,t}$ and $\sin(\Theta)$ have opposite signs, i.e. if $\sin(\Theta)(CS_{1,t} - DS_{2,t}) < 0$, the volume of $\mathcal{P}'$ is strictly smaller than that of $\mathcal{P}$. We now show that, because of our hypotheses, there always exist a rotation satisfying this property.

Let P' be projection of $\mathcal{F}_2 \cap \mathcal{C}$ on $\mathcal{F}_1$ along the other faces of $\mathcal{P}$. By hypothesis, the intersection of P' and $\mathcal{F}_1 \cap \mathcal{C}$ is empty. P' and $\mathcal{F}_1 \cap \mathcal{C}$ are polyhedra as $\mathcal{F}_1 \cap \mathcal{C}$ (resp. $\mathcal{F}_2 \cap \mathcal{C}$) is either a single vertex, an edge, or a face of $\mathcal{C}$. P' and $\mathcal{F}_1 \cap \mathcal{C}$ are two bounded (convex) polyhedra and, as their intersection is empty, there exists a line d of $\mathcal{F}_1$ which separates them strictly: in $\mathcal{F}_1$, P' and $\mathcal{F}_1 \cap \mathcal{C}$ lie on either sides of d, none of them having some points in common with d (the dubious reader will find in the Appendix the lemma 2 which proves the existence of d). We take for d_1 the line of $\mathcal{F}_1$ parallel to d and containing a vertex of $\mathcal{F}_1 \cap \mathcal{C}$ which is the closest to P'. We chose for d_2 the line of $\mathcal{F}_2$ parallel to d and containing a vertex of $\mathcal{F}_2 \cap \mathcal{C}$ whose projection on $\mathcal{F}_1$ is a vertex of P' which is the closest to $\mathcal{F}_1 \cap \mathcal{C}$. We define $S_{1,t}$ and $S_{2,t}$ from d_1 and d_2 as previously. Then, $CS_{1,t}$ and $DS_{2,t}$ cannot be equal. Otherwise, the projection of d_2 on $\mathcal{F}_1$ would be equal to d_1 which is impossible by definition of d_1, d_2, and d (d would have points common to P' and $\mathcal{F}_1 \cap \mathcal{C}$).

Figure 3 (resp. 4) shows the case where $CS_{1,t} - DS_{2,t} > 0$ (resp. $CS_{1,t} - DS_{2,t} < 0$). In this case, one can rotate the pair of planes $\{\mathcal{P}_1, \mathcal{P}_2\}$ of a same angle $\Theta < 0$ (resp. $\Theta > 0$) around d_1 and d_2 respectively while the obtained parallelepiped remains enclosing, and while $(CS_{1,t} - DS_{2,t})\sin(\Theta) < 0$. Hence, the obtained parallelepiped has a volume strictly smaller than $\mathcal{P}$. ■

The following lemma is a corollary of Theorem 2. This lemma states whether two pairs of planes satisfying the condition of Theorem 1 can satisfy the condition

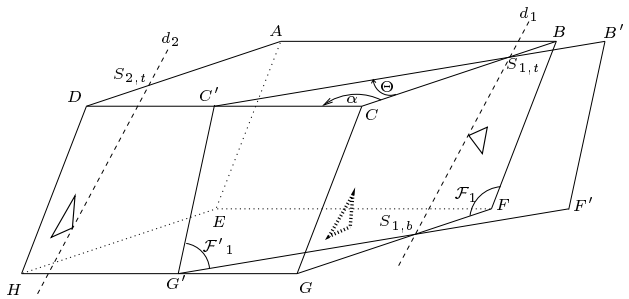

Figure 3: Case $CS_{1,t} - DS_{2,t} > 0$. The intersections of $\mathcal{C}$ and $\mathcal{F}_1$ and $\mathcal{F}_2$ are drawn in bold. The projection P' of $\mathcal{F}_2 \cap \mathcal{C}$ is drawn in dotted lines.

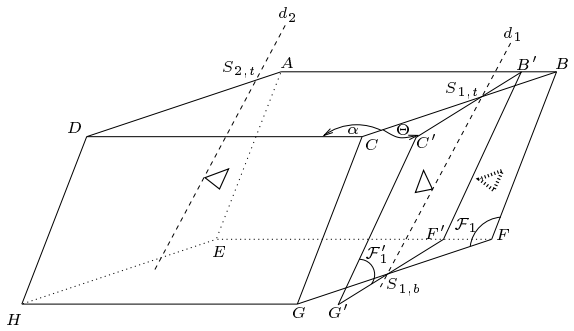

Figure 4: Case $CS_{1,t} - DS_{2,t} < 0$. The intersections of $\mathcal{C}$ and $\mathcal{F}_1$ and $\mathcal{F}_2$ are drawn in bold. The projection P' of $\mathcal{F}_2 \cap \mathcal{C}$ is drawn in dotted lines.

of Theorem 2, in which case we speak of *compatible* pairs of planes. This lemma is thus a weak version of Theorem 2.

Lemma 1 *Let $\mathcal{S}$ be a set of points and $\mathcal{C}$ its convex hull. Let $\{\mathcal{P}_1, \mathcal{P}_2\}$ and $\{\mathcal{P}_3, \mathcal{P}_4\}$ be two pairs of planes satisfying the property stated by Theorem 1 for $\mathcal{S}$. Let $\mathcal{V}_i = \{v_1^i, ..., v_{|\mathcal{V}_i|}^i\}$ be the vertices of $\mathcal{P}_i \cap \mathcal{C}$, for any $i \in [1;4]$. Let $n_{1,2}$ (resp. $n_{3,4}$) be a vector normal to $\mathcal{P}_1$ and $\mathcal{P}_2$ (resp. $\mathcal{P}_3$ and $\mathcal{P}_4$).*

$\{\mathcal{P}_1, \mathcal{P}_2\}$ and $\{\mathcal{P}_3, \mathcal{P}_4\}$ can satisfy the property stated by Theorem 2 if and only if

$$\begin{cases} \exists (a,b) \in \mathcal{V}_1 \times \mathcal{V}_2, & (b-a).n_{3,4} \geq 0, \quad \exists (c,d) \in \mathcal{V}_1 \times \mathcal{V}_2, \quad (d-c).n_{3,4} \leq 0 \\ \exists (e,f) \in \mathcal{V}_3 \times \mathcal{V}_4, & (f-e).n_{1,2} \geq 0, \quad \exists (g,h) \in \mathcal{V}_3 \times \mathcal{V}_4, \quad (h-g).n_{1,2} \leq 0. \end{cases}$$

This lemma, proved in the appendix, just mathematically states that the pair of planes $\{\mathcal{P}_1, \mathcal{P}_2\}$ contains a direction which maps a point of $\mathcal{P}_3 \cap \mathcal{C}$ on a point of $\mathcal{P}_4 \cap \mathcal{C}$, and reciprocally.

3 Algorithms

Using Theorem 1 we derive a rather simple algorithm. Then we refine it using Lemma 1.

3.1 A first algorithm

Theorem 1 tells us that there is at least one minimal volume enclosing parallelepiped such that each of its faces is either parallel to a face of the convex hull or to two non-parallel edges of this convex hull. Then, Algorithm 1 simply enumerates all the possible triplets of orientation of the supporting planes, and search which one gives an enclosing parallelepiped of minimal volume. The algorithm is rather straightforward: after the computation of the convex hull, we build the pair of candidate supporting planes defined by faces of the convex hull, then the pair of candidate supporting planes defined by a pair of edges of the convex hull, and we test all the triplets of pairs of candidate supporting planes. The volumes of the parallelepipeds are computed using a formula proved in appendix (Lemma 3).

Theoretical complexity

Let n be the number of points in $\mathcal{S}$. Its convex hull $\mathcal{C}$ contains v vertices with $v \leq n$. If $\mathcal{C}$ was enforced to be simplicial, it contains exactly $2v-4$ faces and $3v-6$ edges [1].

Algorithm 1 Compute a minimal volume parallelepiped enclosing the set of points $\mathcal{S}$.

1: Compute the convex hull $\mathcal{C}$ of the set of points $\mathcal{S}$
2: $\mathcal{N} = \emptyset$ {The set of candidate supporting planes}
3: Let $\mathcal{F}$ be the set of all the faces of $\mathcal{C}$
4: **for** each face f of $\mathcal{F}$ **do**
5: Find the vertex v of $\mathcal{C}$ which is the furthest from f
6: Associate to f the vector n_f normal to f and linking f and v ($v + n_f$ is a point of f)
7: $\mathcal{N} = \mathcal{N} \cup \{(f, f - n_f, n_f)\}$
8: Let $\mathcal{E}$ be the set of all the edges of $\mathcal{C}$
9: **for** each pair $\{e_1, e_2\}$ of elements of $\mathcal{E}$ **do**
10: **if** e_1 and e_2 are not parallel **then**
11: Build the planes f_1 and f_2 parallel to e_1 and e_2, f_1 containing e_1 and f_2 including e_2
12: Compute the vector n_{f_1} normal to f_1 (and thus to f_2) such that $f_1 + n_{f_1} = f_2$
13: **if** $\mathcal{C}$ is enclosed in the space between the planes f_1 and f_2 **then**
14: $\mathcal{N} = \mathcal{N} \cup \{(f_1, f_2, n_{f_1})\}$
15: vol_min $= +\infty$
16: planes $= \emptyset$
17: **for** each element (f_1, f_1', n_1) of $\mathcal{N}$ **do**
18: **for** each element (f_2, f_2', n_2) of $\mathcal{N}$ **do**
19: **for** each element (f_3, f_3', n_3) of $\mathcal{N}$ **do**
20: **if** $n_1 \wedge n_2.n_3 \neq 0$ **then**
21: vol $= \left| \frac{||n_1||^2 ||n_2||^2 ||n_3||^2}{n_1 \wedge n_2.n_3} \right|$
22: **if** vol $<$ vol_min **then**
23: vol_min $=$ vol
24: planes $= \{f_1, f_1', f_2, f_2', f_3, f_3'\}$
25: **return** planes

Then, the set $\mathcal{N}$ contains at most $\Theta(9v^2) = O(n^2)$ faces. Except for the loops, all the operations in this algorithm are performed in constant time except for the steps 1, 5 and 13:

- Step 1: the computation of the convex hull costs $O(n \log n)$ [1];
- Step 5: to find the vertex which is the furthest from a face of the convex hull, we need to scan all the vertices which costs at worst $O(n)$;

- Step 13: for this test we simply check that the direction of edge e_1 (resp. e_2) has two scalar products of opposite signs with the normals to the two faces of the convex hull containing e_2 (resp. e_1) (to see it, write that $e_1 \wedge e_2$, the normal to the new plane, is a convex combination of the normals to the two faces, and take the scalar product with e_1 or e_2); hence a cost of $O(1)$.

The overall theoretical complexity of this algorithm is thus at worst $O(n^6)$, where n is the number of vertices of $\mathcal{S}$, because of the search on all the triplets of elements of $\mathcal{N}$. More precisely, the complexity of this algorithm is in $O(n \log n + v^6)$, where n is the number of vertices of S and v the number of vertices of its convex hull. We will see in Section 4 that the complexity is far better in practice. Nevertheless, we now use Lemma 1 to speed-up our algorithm.

3.2 A second algorithm

We use Theorem 2 to refine Algorithm 1. Theorem 2 gives us a condition for a triplet of pairs of parallel planes to be an actual candidate for a minimal volume enclosing parallelepiped. Of course, we do not want to enumerate anymore any triplets of pairs of candidate parallel planes. Thus we use Lemma 1 to check whether two pairs of candidate planes can be used together in a minimal enclosing parallelepiped. This way we obtain Algorithm 2.

Theoretical complexity

The worst case complexity of Algorithms 1 and 2 is obviously the same. If we study more carefully the algorithm and denote by v the number of vertices of the convex hull, by e the number of faces built at steps 9 to 14, and by c the size of the largest of the sets "compatible(f_1, f_1', n_1)". Then steps 4 to 7 have a complexity of $O(v^2)$, steps 9 to 14 have a complexity of $O(v^2)$, steps 19 to 23 have a complexity of $O((v+e)^2)$ (at least if $\mathcal{C}$ is simplicial), and steps 24 to 31 have a complexity of $O((v+e) \times c^2)$. Hence the overall complexity of

$$O(n \log n + (v+e)^2 + v \times c^2) \tag{2}$$

4 Experiments

We first compare the two algorithms on our application: we run the two algorithms on all the 45 proteins we had. The results presented on Figure 5 show that Algorithm

Algorithm 2 Compute a minimal volume parallelepiped enclosing the set of points $\mathcal{S}$ (optimized).

Compute the convex hull $\mathcal{C}$ of the set of points $\mathcal{S}$
$\mathcal{N} = \emptyset$ {The set of candidate supporting planes}
Let $\mathcal{F}$ be the set of all the faces of $\mathcal{C}$
for each face f of $\mathcal{F}$ **do**
  Find the vertex v of $\mathcal{C}$ which is the furthest from f
  Associate to f the vector n_f normal to f and linking f and v ($v + n_f$ is a point of f)
  $\mathcal{N} = \mathcal{N} \cup \{(f, f - n_f, n_f)\}$
Let $\mathcal{E}$ be the set of all the edges of $\mathcal{C}$
for each pair $\{e_1, e_2\}$ of elements of $\mathcal{E}$ **do**
  if e_1 and e_2 are not parallel **then**
    Build the planes f_1 and f_2 parallel to e_1 and e_2, f_1 containing e_1 and f_2 including e_2
    Compute the vector n_{f_1} normal to f_1 (and thus to f_2) such that $f_1 + n_{f_1} = f_2$
    if $\mathcal{C}$ is enclosed in the space between the planes f_1 and f_2 **then**
      $\mathcal{N} = \mathcal{N} \cup \{(f_1, f_2, n_{f_1})\}$
vol_min $= +\infty$
planes $= \emptyset$
for each element (f_1, f_1', n_1) of $\mathcal{N}$ **do**
  compatible$(f_1, f_1', n_1) = \emptyset$
for each element (f_1, f_1', n_1) of $\mathcal{N}$ **do**
  for each element (f_2, f_2', n_2) of $\mathcal{N}$ **do**
    if (f_1, f_1', n_1) and (f_2, f_2', n_2) satisfy Lemma 1 **then**
      compatible$(f_1, f_1', n_1) =$ compatible$(f_1, f_1', n_1) \cup \{(f_2, f_2', n_2)\}$
      compatible$(f_2, f_2', n_2) =$ compatible$(f_2, f_2', n_2) \cup \{(f_1, f_1', n_1)\}$
for each element (f_1, f_1', n_1) of $\mathcal{N}$ **do**
  for each element (f_2, f_2', n_2) of compatible(f_1, f_1', n_1) **do**
    for each element (f_3, f_3', n_3) of (compatible$(f_1, f_1', n_1) \cap$ compatible(f_2, f_2', n_2)) **do**
      if $n_1 \wedge n_2.n_3 \neq 0$ **then**
        vol $= \left| \frac{||n_1||^2 ||n_2||^2 ||n_3||^2}{n_1 \wedge n_2.n_3} \right|$
        if vol $<$ vol_min **then**
          vol_min $=$ vol
          planes $= \{f_1, f_1', f_2, f_2', f_3, f_3'\}$
return planes

2 is significantly more efficient than Algorithm 1 even for small inputs. These results are confirmed by Figure 6 which presents a comparison of the two algorithms on larger and synthetic input sets (points randomly picked on a sphere).

Algorithm 2 being far more efficient, we focused on it. We wanted to determine what was its complexity in practice. Thus we needed to run it on convex hulls with a large number of vertices. As the proteins we had did not give us such examples —the convex hull of our worst-case protein only had 94 vertices— we used synthetic data. We randomly picked points on the surface of a sphere as for such sets of points the convex hull is almost equal to the number of points in the set. Figure 7 shows the result of the experiment for convex hulls containing up to 10 000 vertices. The graph of the execution time *Time(n)* in function of the number n of vertices of the convex hull "looks" quadratic. Indeed the graph of $Time(n)/n^2$ is almost an horizontal line (this graph is also displayed on Figure 7 but scaled up to be readable). To confirm this result we approximate the execution with a cubic function (using the nonlinear least-squares Marquardt-Levenberg algorithm implemented in gnuplot). We exactly found:

$$Time(n) \approx 2.15263 \times 10^{-11} \times n^3 + 2.09904 \times 10^{-06} \times n^2 - 0.00101368 \times n + 0.770604$$

with an asymptotic error of 21.18% on the cubic term, and of 3.188% on the quadratic term. The corresponding graph is also drawn on Figure 7 but is hardly seen as it is almost equal to the *Time(n)* graph. Even if this function is cubic, its cubic term has almost no influence for convex hulls of up to 10^5 vertices as, until then, the quadratic term is dominant. We tried to extend this result by running Algorithm 2 on larger sets. The result is presented on Figure 8. There, the computed cubic approximation as an even less important cubic term $((8.66169 \times 10^{-14} \pm 8.503 \times 10^{-13}) \times n^3$ for $(2.44779 \times 10^{-06} \pm 5.343 \times 10^{-08}) \times n^2)$. This is not really surprising as the experimental uncertainties are rather important compared to this cubic term. Furthermore, we only ran experiments up to 40 000 vertices as for such large convex hulls, the algorithm already takes around one hour to run on our experimental platform (Intel Xeon CPU running at 1.80 GHz and 512 MB of memory, C++ program compiled with GNU g++ 3.0, the convex hulls being computed using the Qhull library [3]).

One can wonder whether these results are influenced by the type of synthetic data we used. Therefore, we studied the execution time of Algorithm 2 on purely random sets of points containing up to 150 000 points. Figure 9 presents the graph *Time(n)* in function of the number n of vertices of the convex hull and the graph of $Time(n)/n^2$ (scaled up). In this figure, the execution time does not take into account the time needed to compute the convex hull (when it is included in all other figures). The reason of this removal is quite simple: even with large sets of points, the size of the convex hull is rather small (less than 250 vertices) but most of the time is spent in its computation because of the size of the input sets. The graphs have the desired shape. But the convex hulls are too small for the graphs to be conclusive.

From our experiments we can conclude that Algorithm 2 as an apparent complexity of

$$O(n \log n + v^2)$$

where n is the number of points in the sets S, and v is the number of vertices of the convex hull. This seems at least true for input sets whose convex hull as up to 10^5 vertices, which seems to be the only input sets that may be processed in a reasonable time (we may even wonder whether so large convex hulls exist in practice). This result is quite coherent with Equation 2 when we remark that in all our examples we have found that $e \leq v$ (with the notations of Section 3.2).

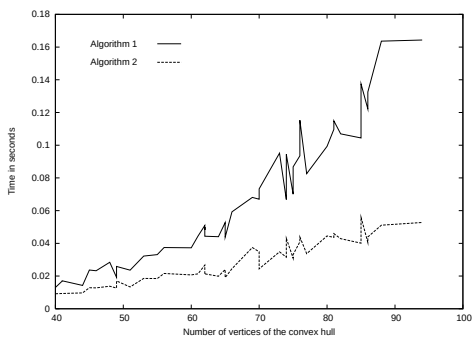

Figure 5: Comparison of the execution time of the two algorithms on 45 proteins.

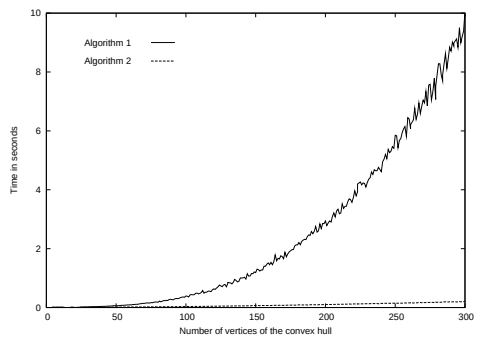

Figure 6: Comparison of the execution time of the two algorithms on synthetic data.

5 Application to proteins

Our initial motivation is to approximate the "surface" of a protein with a set of regular shapes. We hope to be able to discover, by this method, the "faces" of the protein responsible of its interactions with other biological objects, when such faces actually exist. Once we have approximated a protein by its minimal volume enclosing parallelepiped, we consider the "composition" of each of the six faces of the minimal volume enclosing parallelepiped.

A protein is a sequence of amino-acids. The two main characteristics of amino-acids are whether they are electrically charged [1] and whether they are attracted

[1] The electrically charged amino-acids are: aspartic acid, glutamic acid, lysine, arginine, and histidine.

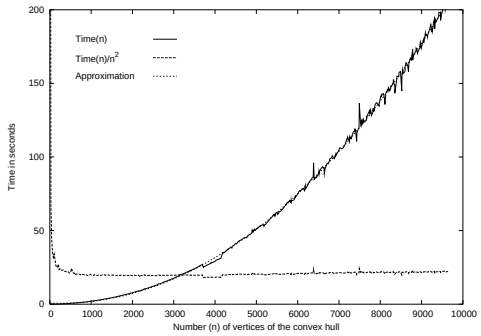

Figure 7: Execution time of Algorithm 2 on 499 sets of points randomly taken on a sphere.

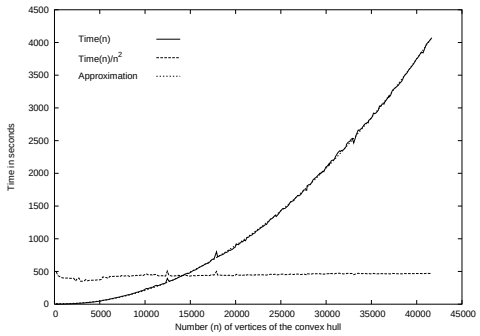

Figure 8: Execution time of Algorithm 2 on 248 sets of points randomly taken on a sphere.

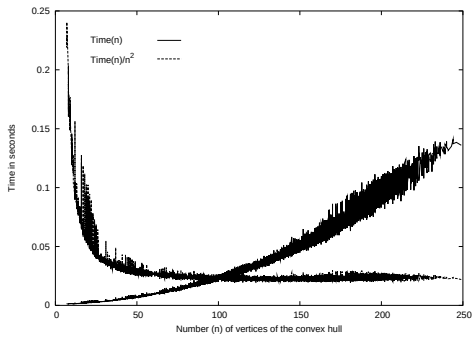

Figure 9: Execution time of Algorithm 2 on 15073 sets of random points.

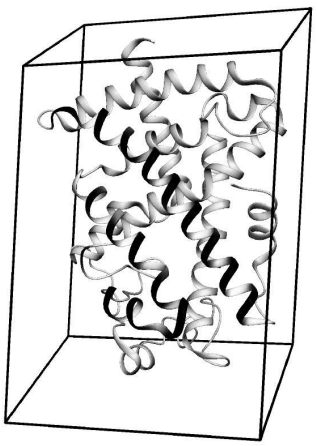

Figure 10: The PPAR protein with its minimal volume enclosing parallelepiped.

by water (hydrophile amino-acids) or repulsed (hydrophobic amino-acids) [2]. So we consider the composition of the faces of our parallelepiped in terms of electrically charged and hydrophobic amino-acids. The composition of a face is the set of the

[2] The hydrophobic amino-acids are: leucine, isoleucine, valine, methionine, phenylalanine, tyrosine, and tryptophan.

amino-acids whose center of gravity is close to the face (less than 2.4 Å away from the face in our model).

We chose to illustrate our work with a protein which is a nuclear receptor. A nuclear receptor initiates the transcription of some part of the DNA when it is activated by a certain molecule called its ligand. More important for us, nuclear receptors are known to have a large interaction face: we want to check whether we are able to rediscover this interaction face.

We chose the nuclear receptor protein called PPAR (*Peroxisome Proliferator-Activated Receptor*). This protein is involved in the metabolism of glucose, lipids, and cholesterol. PPAR is presented on Figure 10 with its minimal volume enclosing parallelepiped [3]. The composition of the parallelepiped faces is summarized in Figure 12 (the numbering of the parallelepiped faces is presented on Figure 11). From biological considerations, faces 5 and 6 do not "contain" enough amino-acids to be significant. Among the remaining faces, Face 1 is the one containing the smallest percentage of hydrophobic amino-acid and the one containing the biggest percentage of electrically charged amino-acids. Face 1 as thus an outstanding composition (the amino-acids belonging to Face 1 are drawn the darkest on Figure 11). Actually, Face 1 corresponds to the dimerisation interface of PPAR: thanks to this interface, PPAR can form an heterodimer with the protein RXR (*Retinoid X Receptor*). Therefore, we were able to re-discover PPAR interface.

We do not claim from the above example that our method enables us to predict anything: we only presented this example to give an insight to our motivation and application. In the general case, we cut a protein in sub-pieces (if necessary) and we approximate each sub-piece with its minimal volume enclosing parallelepiped. The whole description of this work goes far beyond the scope of this paper.

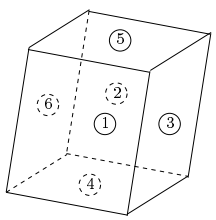

Figure 11: Numbering of the parallelepiped faces.

[3] We used the structure of PPAR proposed by Xu et al. [11] and denoted 1k74 in the *Protein Data Base*.

Face	1	2	3	4	5	6
Number of amino-acids	32	19	18	13	8	4
Hydrophobic amino-acids	6%	21%	22%	38%	0%	0%
Electrically charged amino-acids	50%	47%	44%	38%	62%	75%

Figure 12: Composition of the faces of the minimal volume parallelepiped enclosing PPAR (cf. Figure 10).

6 Conclusion

We presented two mathematical properties of the minimal volume parallelepiped enclosing a three-dimensional set of points S. Using these properties we designed two algorithms of theoretical complexity $O(n^6)$, where n is the size of S (the number of points it contains). Our experiments show that the practical complexity of our quickest algorithm is $O(n \log n + v^2)$ where n is the size of S and v the number of vertices of its convex hull, at least when v is smaller than 10^5. Finally, we applied our method to search for the interaction faces of a protein, our initial goal. Although the application of this research to structural biology is in the preliminary stages, the first results are promising.

References

[1] J.-D. Boissonnat and M. Yvinec. *Géométrie algorithmique*. Ediscience international, 1995.

[2] J. O'Rourke. Finding minimal enclosing boxes. *International Journal of Computer and Information Sciences*, 14(3):183–199, June 1985.

[3] Qhull library. `http://www.geom.umn.edu/software/qhull` or `http://www.thesa.com/software/qhull`.

[4] P. Quinton. *Automata Networks in Computer Science*, chapter The systematic design of systolic arrays. Manchester University Press, 1987.

[5] E. Schömer, J. Sellen, M. Teichmann, and C. Yap. Smallest enclosing cylinders. *Algorithmica*, 27:170–186, 2000.

[6] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, New York, 1986.

[7] C. Schwarz, J. Teich, A. Vainshtein, E. Welzl, and B. L. Evans. Minimal enclosing parallelogram with application. In *Proceedings of the eleventh annual symposium on Computational geometry*, pages 434–435, Vancouver, British Columbia, Canada, June 1995. ACM Press New York, NY, USA.

[8] C. Schwarz, J. Teich, E. Welzl, and B. Evans. On finding a minimal enclosing parallelgram. Technical Report 036, The International Computer Science Institute, Berkeley, 1994.

[9] A. Vainshtein. Finding minimal enclosing parallelograms. *Diskretknaya Matematika*, 2:72–81, 1990. In Russian.

[10] E. Welzl. Smallest enclosing disks (balls and ellipsoids). In H. Maurer, editor, *New Results and New Trends in Computer Science*, volume 555 of *LNCS*, pages 359–370. Springer-Verlag, 1991.

[11] H. Xu, M. Lambert, V. Montana, K. Plunket, L. Moore, J. Collins, J. Oplinger, S. Kliewer, J. Gampe, and D. McKee. Structural determinants of ligand binding selectivity between the peroxisome proliferator-activated receptors. *Proc Natl Acad Sci*, 98:13919–13924, 2001.

[12] Y. Zhou and S. Suri. Algorithms for minimum volume enclosing simplex in $\mathbb{R}^3$. *SIAM Journal on Computing*, 31:1339–1357, 2002.

A Additional results and proofs

Proof of Lemma 1 For the two pairs of parallel planes to have a chance to satisfy the property stated by Theorem 2, there must exist a point x in $\mathcal{P}_1 \cap \mathcal{C}$ and a point y in $\mathcal{P}_2 \cap \mathcal{C}$ and a direction d in $\mathcal{P}_3$ such that the projection of x on $\mathcal{P}_2$ along d is equal to y. In other words, the vector $y - x$ must be parallel to $\mathcal{P}_3$. This property is equivalent to $(y - x).n_{3,4} = 0$. We prove that this property is equivalent to the system of Lemma 1.

The points of $\mathcal{P}_1 \cap \mathcal{C}$ are exactly the convex combinations of the vertices of $\mathcal{P}_1 \cap \mathcal{C}$. We use this property for the point x of $\mathcal{P}_1 \cap \mathcal{C}$ and also for the point y in $\mathcal{P}_2 \cap \mathcal{C}$:

$$\exists \lambda_1 \geq 0, ..., \exists \lambda_{|\mathcal{V}_1|} \geq 0, \ \sum_{j=1}^{|\mathcal{V}_1|} \lambda_j = 1, x = \sum_{j=1}^{|\mathcal{V}_1|} \lambda_j v_j^1, \text{ and}$$

$$\exists \mu_1 \geq 0, ..., \exists \mu_{|\mathcal{V}_2|} \geq 0, \ \sum_{k=1}^{|\mathcal{V}_2|} \mu_k = 1, y = \sum_{k=1}^{|\mathcal{V}_2|} \mu_k v_k^2.$$

$$y - x = \sum_{k=1}^{|\mathcal{V}_2|} \mu_k v_k^2 - \sum_{j=1}^{|\mathcal{V}_1|} \lambda_j v_j^1 = \sum_{k=1}^{|\mathcal{V}_2|} \left(\sum_{j=1}^{|\mathcal{V}_1|} \lambda_j \right) \mu_k v_k^2 - \sum_{j=1}^{|\mathcal{V}_1|} \left(\sum_{k=1}^{|\mathcal{V}_2|} \mu_k \right) \lambda_j v_j^1 \Leftrightarrow$$

$$y - x = \sum_{k=1}^{|\mathcal{V}_2|} \sum_{j=1}^{|\mathcal{V}_1|} \mu_k \lambda_j (v_k^2 - v_j^1),$$

and $(y - x)$ is a convex combination of the values $(v_k^2 - v_j^1)$. We have three cases to consider:

- All the scalar products $(v_k^2 - v_j^1).n_{3,4}$ are (strictly) positive (resp. negative). Then, the scalar product $(y - x).n_{3,4}$ is also (strictly) positive (resp. negative) and the two pairs of parallel planes cannot satisfy the property stated by Theorem 2.

- At least one of the scalar products is null: the two pairs of parallel planes can obviously satisfy the property.

- No scalar product is null, but there exist some values k_1, k_2, j_1, j_2 such that $(v_{k_1}^2 - v_{j_1}^1).n_{3,4} > 0$ and $(v_{k_2}^2 - v_{j_2}^1).n_{3,4} < 0$. We define the points x and y as follows:
$$x = \frac{|(v_{k_2}^2 - v_{j_2}^1).n_{3,4}| v_{j_1}^1 + |(v_{k_1}^2 - v_{j_1}^1).n_{3,4}| v_{j_2}^1}{|(v_{k_2}^2 - v_{j_2}^1).n_{3,4}| + |(v_{k_1}^2 - v_{j_1}^1).n_{3,4}|} \text{ and}$$

$$y = \frac{|(v^2_{k_2} - v^1_{j_2}).n_{3,4}|v^2_{k_1} + |(v^2_{k_1} - v^1_{j_1}).n_{3,4}|v^2_{k_2}}{|(v^2_{k_2} - v^1_{j_2}).n_{3,4}| + |(v^2_{k_1} - v^1_{j_1}).n_{3,4}|}.$$

One can check that x belongs to $\mathcal{P}_1 \cap \mathcal{C}$, y to $\mathcal{P}_2 \cap \mathcal{C}$ and that $(y - x).n_{3,4} = 0$.

To obtain the desired property, we redo on the pair of planes $\{\mathcal{P}_3, \mathcal{P}_4\}$ what we have done on $\{\mathcal{P}_1, \mathcal{P}_2\}$. ■

The following lemma is used in the proof of Theorem 1.

Lemma 2 *Let A and B be two polytopes (bounded convex polyhedra) which have an empty intersection. There exists an hyperplane which strictly separates A and B. In other words, there exists an affine form $x \mapsto \lambda x + \mu$ which takes (strictly) negative values on A and (strictly) positive values on B.*

Proof Let $C = A - B = \{z | \exists x \in A, \exists y \in B, z = x - y\}$. C does not contain 0 as, by hypothesis, $A \cap B = \emptyset$.

C **is convex.** We take two points $z_1 = x_1 - y_1$ and $z_2 = x_2 - y_2$ of C ($x_1 \in A$, $x_2 \in A$, $y_1 \in B$ and $y_2 \in B$). Let α be any value in $[0; 1]$. $\alpha z_1 + (1 - \alpha)z_2 = (\alpha x_1 + (1 - \alpha)x_2) - (\alpha y_1 + (1 - \alpha)y_2)$. As A and B are convex $(\alpha x_1 + (1 - \alpha)x_2)$ belongs to A and $(\alpha y_1 + (1 - \alpha)y_2)$ to B. Therefore, $\alpha z_1 + (1 - \alpha)z_2$ belongs to C and C is convex.

C is a polytope. To prove it, we use Minkowski's representation of polytopes [6]: there exists a set $\{v_1, ..., v_p\}$ of vertices of A such that:

$$A = \left\{ x \middle| x = \sum_{i=1}^{p} \alpha_i v_i, \forall i \in [1, p] \alpha_i \geq 0, \sum_{i=1}^{p} \alpha_i = 1 \right\}.$$

Symmetrically, we denote by $\{w_1, ..., w_q\}$ the vertices of B. We show that C is a polyhedron by showing that it admits as (not necessarily minimal) Minkowski's representation the set $V = \{v_i - w_j\}_{1 \leq i \leq p, 1 \leq j \leq q}$. It is obvious that V is included in C. By convexity of C, the polytope generated by V, which is the convex hull of V, is included in C. We still have to show that C is included in this polyhedron. Let $z = x - y$ be any point of C. x belongs to A and y to B. Thus, there exist some values $\alpha_1, ..., \alpha_p$ and $\beta_1, ..., \beta_q$ such that:

$$\forall i \ \alpha_i \geq 0, \ \forall j \ \beta_i \geq 0, \ \sum_{i=1}^{p} \alpha_i = 1, \ \sum_{j=1}^{q} \beta_j = 1, \ x = \sum_{i=1}^{p} \alpha_i v_i \text{ and } y = \sum_{j=1}^{q} \beta_j w_j.$$

Hence:

$$z = \sum_{i=1}^{p} \alpha_i v_i - \sum_{j=1}^{q} \beta_j w_j = \sum_{i=1}^{p} \left(\sum_{j=1}^{q} \beta_j \right) \alpha_i v_i - \sum_{j=1}^{q} \left(\sum_{i=1}^{p} \alpha_i \right) \beta_j w_j.$$

Therefore:

$$z = \sum_{i=1}^{p} \sum_{j=1}^{q} \alpha_i \beta_j (v_i - w_j), \text{ with } \sum_{(i,j) \in [1;p] \times [1;q]} \alpha_i \beta_j = 1$$

and $\forall (i,j) \in [1;p] \times [1;q], \alpha_i \beta_j \geq 0$ and . Thus z is a point of the polyhedron generated by V.

There exists a vector λ such that $\forall \mathbf{x} \in \mathbf{C}, \lambda.\mathbf{x} < \mathbf{0}$. We show this property by contradiction. Thus, we suppose that:

$$\forall \lambda, \exists x \in C, \lambda.x \geq 0. \tag{3}$$

C is a polyhedron. Therefore, by definition, there exists a set of m hyperplanes $a_i x + b_i \geq 0$ such that: $C = \cap_{i=1}^{m} \{x | a_i x + b_i \geq 0\}$ [6]. We suppose that there exists a value $i \in [1;m]$ such that $b_i < 0$. Then, for any element x of C, $a_i x + b_i \geq 0 \Leftrightarrow (-a_i)x \leq b_i < 0$. Thus, for any element x of C, $(-a_i)x < 0$ and $\lambda = (-a_i)$ does not satisfy Equation 3 and there is a contradiction. Thus, for any value $i \in [1;m]$, $b_i \geq 0$ and $a_i.0 \geq 0 \geq -b_i$. Thus, for any value $i \in [1;m]$, 0 belongs to the set $\{x | a_i x + b_i \geq 0\}$ and 0 belongs to C, which is impossible.

Building the strictly separating hyperplane. Let λ be a vector satisfying the property we just proved: $\forall x \in C, \lambda.x < 0$. $\forall x \in C, \lambda.x < 0 \Leftrightarrow \forall y \in A, \forall z \in B, \lambda y < \lambda z$. A (resp. B) is a polyhedron. Thus any linear form over A (resp. B) reaches its maximum (resp. minimum) on a vertex, and thus a point, of A (resp. B) [4]. Let y_A (resp. z_B) be such a point. We have: $\forall y \in A, \forall z \in B, \lambda y \leq \lambda y_A < \lambda y_B \leq \lambda z$. The hyperplane defined by the equation: $\lambda x = \frac{\lambda y_A + \lambda y_B}{2}$ strictly separates A and B. ■

Lemma 3 (Another formula to compute the volume of a parallelepiped) *Let $ABCDEFGH$ be a parallelepiped. Let n_1 (resp. n_2) (resp. n_3) be a vector normal to the pair of planes $((DAEH),(CBFG))$ (resp. $((DCGH),(ABFE))$) (resp. $((ABCD),(HEFG))$) whose norm is equal to the distance between these two planes. Then, the volume of the parallelepiped $ABCDEFGH$ is equal to: $\mathcal{V} = \left| \frac{||n_1||^2 ||n_2||^2 ||n_3||^2}{n_1 \wedge n_2 . n_3} \right|$.*

Proof The volume of the parallelepiped is equal to: $\mathcal{V} = |(\boldsymbol{HG} \wedge \boldsymbol{HE}).\boldsymbol{HD}|$. We need to explicit the values of $\boldsymbol{HG}$, $\boldsymbol{HE}$, and $\boldsymbol{HD}$ as functions of n_1, n_2, and n_3. We start with $\boldsymbol{HD}$. Because n_1 and n_2 are perpendicular to $\boldsymbol{HD}$, the direction of $\boldsymbol{HD}$ is equal to $\pm\frac{n_1 \wedge n_2}{||n_1 \wedge n_2||}$. Let α be the angle defined by the vectors $\boldsymbol{HD}$ and n_3. As the triangle defined by H, D, and $H + n_3$ is rectangle, $||\boldsymbol{HD}|| = \frac{||n_3||}{|\cos\alpha|}$ hence: $\boldsymbol{HD} = \pm\frac{||n_3||}{\cos\alpha}\frac{n_1 \wedge n_2}{||n_1 \wedge n_2||}$. Besides, $\cos\alpha = \frac{n_1 \wedge n_2}{||n_1 \wedge n_2||}.\frac{n_3}{||n_3||}$, which implies: $\boldsymbol{HD} = \pm\frac{||n_3||^2 n_1 \wedge n_2}{n_1 \wedge n_2.n_3}$. Similarly we obtain the values of $\boldsymbol{HG}$ and $\boldsymbol{HE}$: $\boldsymbol{HG} = \pm\frac{||n_1||^2 n_2 \wedge n_3}{n_1 \wedge n_2.n_3}$ and $\boldsymbol{HE} = \pm\frac{||n_2||^2 n_3 \wedge n_1}{n_1 \wedge n_2.n_3}$. Collecting these results we have:

$$\begin{aligned}
\mathcal{V} &= \frac{||n_1||^2||n_2||^2||n_3||^2|(n_2 \wedge n_3) \wedge (n_3 \wedge n_1).(n_1 \wedge n_2)|}{|n_1 \wedge n_2.n_3|^3} \\
&= \frac{||n_1||^2||n_2||^2||n_3||^2|[((n_2 \wedge n_3).n_1)n_3 - ((n_2 \wedge n_3).n_3)n_1].(n_1 \wedge n_2)|}{|n_1 \wedge n_2.n_3|^3} \\
&= \frac{||n_1||^2||n_2||^2||n_3||^2|[((n_2 \wedge n_3).n_1)n_3].(n_1 \wedge n_2)|}{|n_1 \wedge n_2.n_3|^3} \\
&= \frac{||n_1||^2||n_2||^2||n_3||^2}{|n_1 \wedge n_2.n_3|}
\end{aligned}$$

(using the formula $u \wedge (v \wedge w) = (u.w)v - (u.v)w$). ■

Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399