HAL open science

Minkowski sum of HV-polytopes in Rn

Vincent Delos, Denis Teissandier

To cite this version:

Vincent Delos, Denis Teissandier. Minkowski sum of HV-polytopes in Rn. 4th Annual International Conference on Computational Mathematics, Computational Geometry and Statistics, Jan 2015, Singapore, Singapore. hal-01092060

HAL Id: hal-01092060

https://hal.science/hal-01092060v1

Submitted on 8 Dec 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Minkowski sum of $\mathcal{HV}$-polytopes in $\mathbb{R}^n$

Vincent Delos* and Denis Teissandier**

University of Bordeaux
CNRS, National Center for French Research
I2M, UMR 5295
Talence, F-33400, France
*E-mail: v.delos@i2m.u-bordeaux1.fr
**E-mail: d.teissandier@i2m.u-bordeaux1.fr

Abstract

Minkowski sums cover a wide range of applications in many different fields like algebra, morphing, robotics, mechanical CAD/CAM systems ... This paper deals with sums of polytopes in a n dimensional space provided that both $\mathcal{H}$-representation and $\mathcal{V}$-representation are available i.e. the polytopes are described by both their half-spaces and vertices. The first method uses the polytope normal fans and relies on the ability to intersect dual polyhedral cones. Then we introduce another way of considering Minkowski sums of polytopes based on the primal polyhedral cones attached to each vertex.

keywords: Computational Geometry, Convex Polytope, Minkowski Sum, Normal Fan, Polyhedrical Cone.

1 Introduction

In mechanical design, tolerancing analysis consists in simulating the geometric variations due to the manufacturing process. A common way to simulate the variations of an over-constrained mechanical system is to manipulate sets of constraints in $\mathbb{R}^6$, to limit the 6 degrees of freedom (3 translations and 3 rotations), see [1]. In order to compute the cumulative stack-up of variations we need to calculate the Minkowski sums [2] and intersections of sets of contraints modelled by polytopes in $\mathbb{R}^6$. Two algorithms were developped in this direction in [3] and [4] but only in $\mathbb{R}^3$. In [5] and [6], Delos and Fukuda introduce other methods summing polytopes in $\mathbb{R}^n$ but they only work with the polytopes $\mathcal{V}$-description and in tolerancing analysis, the polytopes are first defined by half-spaces. We can take advantage of this important property in order to set up the following algorithm. Moreover, making use of the $\mathcal{H}$-representation can speed up the algorithm and is a key element in computing intersections further on. And finally, as we are in small dimensions, computing the double description is not a problem as stated in [6] we can find cases where: *"the number of facets of the convex hull of a set of k points in Euclidean n-space can be of order $\mathcal{O}(k^{\lfloor n/2 \rfloor})$ even when n is considered fixed"*.

The goal of this paper is to describe two ways of computing Minkowski sums of polytopes, whether we choose to work in the primal or dual spaces. The dual space approach has come to an algorithm implemented and tested in C++ while the other one is still under investigation. We finally introduce some promising perspectives to reach the objective of having a stable algorithm in the field of tolerancing analysis.

2 Basic properties

2.1 Minkowski sums

Given two sets A and B, let C be the Minkowski sum of A and B

$$C = A + B = \{c \in \mathbb{R}^n, \exists a \in A, \exists b \in B / c = a + b\}. \tag{1}$$

2.2 Polytopes

A *polytope* is defined as the convex hull of a finite set of points, called the $\mathcal{V}$-representation, or as the

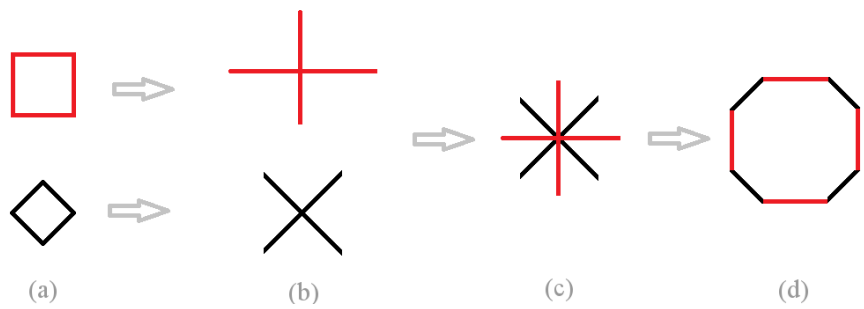

Figure 1: Sums of two polytopes step by step: a) polygons, b) normal fans, c) common refinement, d) sum.

bounded intersection of a finite number of half-spaces, called the $\mathcal{H}$-representation. The Minkowski-Weyl theorem states that both definitions are equivalent.

2.3 Normal fans

For each vertex v of a polytope, we define the set E of its edges oriented towards its neighbours. With $E = \{e_1, ..., e_l\}$ we build a polyhedral cone $C(v)$ named the primal cone of v:

$$C(v) = \{u_1 e_1 + ... + u_l e_l, \forall u_j \geq 0\}. \tag{2}$$

We can note that a polytope can be written as the intersection of all the primal cones attached to its vertices. Let A be a $\mathbb{R}^n$-polytope and $\mathcal{V}_A$ the list of its vertices such as $card(\mathcal{V}_A) = m$:

$$A = \bigcap_{i=1}^{m} C(v_i). \tag{3}$$

For each vertex v of a polytope, we define the set N of the outer normals of its corresponding facets. With $N = \{n_1, ..., n_k\}$ we build a polyhedral cone $C_D(v)$ named the dual cone of v:

$$C_D(v) = \{t_1 n_1 + ... + t_k n_k, \forall t_i \geq 0\}. \tag{4}$$

It is also the set of hyperplane outer normals which find their maximum on this specific vertex:

$$C_D(v) = \{y \in \mathbb{R}^n : \langle v, y \rangle = sup_{x \in A} \langle x, y \rangle\}. \tag{5}$$

$N(A)$ is defined as the set of all the dual cones of a given polytope, it forms a partition of the whole space $\mathbb{R}^n$ which is called the *normal fan* of the polytope.

3 Dual algorithm

As stated in [7] by Weibel *"The normal fan of polytopes contains all of their combinatorial organization. It is therefore enough to compute the normal fan of a Minkowski sum to have its combinatorial properties. We can then easily deduce the polytope itself by combining these informations with the summand polytopes. We know that the normal fan of a Minkowski sum is the common refinement of the normal fans of its summands."* This is why we developped such an approach in a previous article named *Algorithm to calculate the Minkowski sums of 3-polytopes based on normal fans* in $\mathbb{R}^n$. To extend the results in $\mathbb{R}^n$ we need some theoritical results.

3.1 Main properties

Ziegler and Gritzmann give the normal fan of the sum of two $\mathbb{R}^n$-polytopes in [8] and [9]. Let A and B with their respective lists of vertices $\mathcal{V}_A$ and $\mathcal{V}_B$:

$$N(A + B) = N(A) \wedge N(B). \tag{6}$$

where $N(A) \wedge N(B) = \{C_D(a_i) \cap C_D(b_j) : C_D(a_i) \in N(A)\ \forall a_i \in \mathcal{V}_A, C_D(b_j) \in N(B)\ \forall b_j \in \mathcal{V}_B\}$ is called the common refinement. So it is clear that computing the sum of two polytopes can be performed by intersecting polyhedral cones. This is illustrated in Fig. 1.

In the following we will emphasize on how finding the Minkowski vertices i.e. the vertices of the two operands sum.

3.2 Minkowski vertices

Let A, B and C be three $\mathbb{R}^n$-polytopes such as $C = A + B$, let $\mathcal{V}_A$, $\mathcal{V}_B$ and $\mathcal{V}_C$ be their lists of vertices.

Let $c \in \mathcal{V}_C$, from [6] we know that the vertex c can only be the sum of a face from A and a face from B. For reasons of dimension, c is necessarily the sum of two vertices $a \in \mathcal{V}_A$ and $b \in \mathcal{V}_B$. Let's characterize now the dual cone of c.

$$c = (a + b) \in \mathcal{V}_C \Rightarrow C_D(c) = C_D(a) \cap C_D(b). \quad (7)$$

Let $u \in C_D(a) \cap C_D(b)$ then $\langle u, a\rangle = sup_{x\in A}\langle u, x\rangle$ and $\langle u, b\rangle = sup_{y\in B}\langle u, y\rangle$. So $\langle u, a + b\rangle = sup_{x\in A,y\in B}\langle u, x + y\rangle$ which means that $\langle u, c\rangle = sup_{z\in C}\langle u, z\rangle$ i.e. $u \in C_D(c)$.

On the other side let $u \in C_D(c)$, by definition $\langle u, c\rangle = sup_{z\in C}\langle u, z\rangle$. If we decompose c in a sum of vertices a and b we get $\langle u, a\rangle+\langle u, b\rangle = sup_{x\in A}\langle u, x\rangle+ sup_{y\in B}\langle u, y\rangle$, but we know that in general $\langle u, a\rangle \leq sup_{x\in A}\langle u, x\rangle$ and $\langle u, b\rangle \leq sup_{y\in B}\langle u, y\rangle$ so for compatibility reasons with the formula of decomposition of c, the inequalities must be equalities. It gives $\langle u, a\rangle = sup_{x\in A}\langle u, x\rangle$ and $\langle u, b\rangle = sup_{y\in B}\langle u, y\rangle$ which means $u \in C_D(a) \cap C_D(b)$.

Now we have to ask ourselves what are the conditions to get a Minkowski vertex when we compute the intersection $C_D(a) \cap C_D(b)$?

Let A and B be $\mathbb{R}^n$-polytopes of full dimension n, $\mathcal{V}_A$ and $\mathcal{V}_B$ be their vertices lists. Let $a \in \mathcal{V}_A$, and $b \in \mathcal{V}_B$:

$$c = (a + b) \in \mathcal{V}_C \Leftrightarrow dim(C_D(a) \cap C_D(b)) = n. \quad (8)$$

In [10], Fukuda and Weibel indicate that *"Faces of a polytope and their normal cones have dual properties. In particular, if F is a i-dimensional face of A, then the normal cone $C_D(F)$ is a $(n-i)$-dimensional cone of $\mathbb{R}^n$."* So $\exists a \in \mathcal{V}_A, \exists b \in \mathcal{V}_B / c = (a + b) \in \mathcal{V}_C \Rightarrow dim(C_D(a) \cap C_D(b)) = n$ as c is a 0-face.

Reciprocally if $dim(C_D(a) \cap C_D(b)) = n$ then $\exists u \in Int(C_D(a)) \cap Int(C_D(b))$ such as $\forall x \in A, \langle u, a\rangle > \langle u, x\rangle$ and $\forall y \in B, \langle u, b\rangle > \langle u, y\rangle$. So $\langle u, a\rangle + \langle u, b\rangle = \langle u, c\rangle > \langle u, x + y\rangle, \forall(x + y) \in C$. Hence $dim(C_D(a) \cap C_D(b)) = n \Rightarrow c \in \mathcal{V}_C$.

Following the same idea we can find the facets of the polytope C from $N(C)$ edges. We now have all the tools to build Minkowski vertices and facets.

3.3 A first dual algorithm

3.4 Complexity and implementation

To perform such an operation, in [11] Fukuda gives interesting insights and efficient-to-use strategies despite the fact, as the author says, *"that we can hardly state any interesting theorems on its time and space complexities"*. The underlying physical problem we designed this algorithm for, is in low dimension so obtaining the polytopes double description is not a problem. In tolerancing analysis, it can even be done in an analytical way but beware that the number of vertices and facets can be exponential according to the dimension of the space we work in. As a example a tetraedon in $\mathbb{R}^n$ has only $(n+1)$ vertices and $(n+1)$ facets but a cube has $2n$ facets and 2^n vertices. In [12] one can find an optimum algorithm, when n is constant, to compute convex hulls that runs in time $\mathcal{O}(k^{\lceil n/2\rceil})$ for $n \geq 4$, k being the number of vertices. Such an upper bound cannot be reduced as it is of the order of the larger output. So in high dimensions the kind of polytopes you handle can have a very strong impact on the performances of the Minkowski sum algorithm. In [13] we can find a good introduction on families of polytopes and the way they behave in algorithmic contexts.

Algorithm 1 Calculate $C = A + B$ with A and B, two $\mathbb{R}^n$-polytopes of full dimension n

Require: List of dual cones of A $\{C_D(a_i), a_i \in \mathcal{V}_A\}$
Require: List of dual cones of B $\{C_D(b_j), b_j \in \mathcal{V}_B\}$
  for all $a_i \in \mathcal{V}_A$ and $b_j \in \mathcal{V}_B$ **do**
    Compute $C_D(a_i) \cap C_D(b_j)$
    if $dim(C_D(a_i) \cap C_D(b_j)) = n$ **then**
      $(a_i + b_j) \in \mathcal{V}_C$
      Get the half-spaces passing through $(a_i + b_j)$ from $C_D(a_i) \cap C_D(b_j)$ edges
    end if
  end for

This algorithm has been coded in C++ and is available under the Gnu General Public Licence v3.0 at `http://i2m.u-bordeaux.fr/politopix`. It has been tested and is now fully operational in $\mathbb{R}^6$ in the frame of tolerance analysis in mechanical engineering, as well as it is in any dimension given the limitations we previously described.

3.5 An optimized dual algorithm

The basic idea behind this algorithm is quite simple. As soon as we get $C_D(a_i)$ and $C_D(b_j)$ such as $dim(C_D(a_i) \cap C_D(b_j)) = n$, we do not pick dual cones from A in a random way but we rather select the list of $C_D(a_i)$ neighbours to intersect them with $C_D(b_j)$. While we find intersections of dimension n, we keep on picking up the neighbours of the neighbours and

so on.

Let's assume that the two dual cones $C_D(a_i)$ and $C_D(b_j)$ intersects with each other such as, at least one half-space hyperplane $\bar{H}$ of $C_D(a_i)$ separates $C_D(b_j)$. Then it is obvious that the neighbour dual cone $C_D(a_k)$ that shares $\bar{H}$ with $C_D(a_i)$ has also a non empty intersection with $C_D(b_j)$. We can take advantage of this neighbourdhood property to speed up the algorithm.

So we introduce the notion of polyhedral cap. Let A, B and $C = A+B$ be $\mathbb{R}^n$-polytopes and $\mathcal{V}_A$, $\mathcal{V}_B$, $\mathcal{V}_C$ their respective lists of vertices. For a given vertex $a \in \mathcal{V}_A$ we want to find the list of all the vertices of B that will give a Minkowski vertex in C. We define the polyhedral cap of the vertex a_i in the polytope B this way $\mathcal{V}_B^+(a_i) = \{b_j \in \mathcal{V}_B / (a_i + b_j) \in \mathcal{V}_C\}$, its complementary list in $\mathcal{V}_B$ is $\mathcal{V}_B^-(a_i) = \{b_j \in \mathcal{V}_B / (a_i + b_j) \notin \mathcal{V}_C\}$.

Let A et B be two $\mathbb{R}^n$-polytopes, $a_i \in \mathcal{V}_A$:

$$\forall i, \mathcal{V}_B = \mathcal{V}_B^+(a_i) \cup \mathcal{V}_B^-(a_i) \tag{9}$$

$$\forall i, \mathcal{V}_B^+(a_i) \neq \emptyset, \mathcal{V}_B^+(a_i) \text{ is connected.} \tag{10}$$

We want to show that $\forall i, \mathcal{V}_B^+(a_i) \neq \emptyset$. By definition $a_i \in \mathcal{V}_A$ so $dim(C_D(a_i)) = n$, let l be $Card(\mathcal{V}_B)$, so $\forall j \in \{1, ..., l\}$ we have $dim(C_D(b_j)) = n$. Let's admit that for the $l-1$ first dual cones of B we have $\forall j \in \{1, ..., l-1\}$ $dim(C_D(a_i) \cap C_D(b_j)) < n$, which means that they do not intersect with $C_D(a_i)$ or only with its frontier. Given that $\bigcup_{j=1}^{l} C_D(b_j) = \mathbb{R}^n$ it must only be the last dual cone in $N(B)$ that intersects with $C_D(a_i)$ interior. As a consequence $dim(C_D(a_i) \cap C_D(b_l)) = n$ so $a_i + b_l \in \mathcal{V}_C$. In all the cases we have at least one vertex in B that verifies this property so $\mathcal{V}_B^+(a_i) \neq \emptyset$.

We shall proove now that $\mathcal{V}_B^+(a)$ is connected. Let $c_1 = a + b_1 \in \mathcal{V}_C$ and $c_2 = a + b_2 \in \mathcal{V}_C$ i.e. $b_1 \in \mathcal{V}_B^+(a)$ and $b_2 \in \mathcal{V}_B^+(a)$. Is there a path of vertices of B, leading from b_1 to b_2 in $\mathcal{V}_B^+(a)$? We know that $C_D(c_1) = C_D(a) \cap C_D(b_1)$ and $C_D(c_2) = C_D(a) \cap C_D(b_2)$. We choose in the interiors $u_1 \in Int(C_D(a) \cap C_D(b_1))$ and $u_2 \in Int(C_D(a) \cap C_D(b_2))$ so $[u_1, u_2] \subset Int(C_D(a))$. We build the list L of dual cones in $N(B)$ whose intersection is not empty with the segment $[u_1, u_2]$ and that verify the following property: once we leave one of the bounding half-spaces of the current cone $C_D(b_j)$, we add to L its neighbour cone that shares with $C_D(b_j)$ the frontier of this current half-space. We can note that once we leave a cone, we will never process it again, as we remain on the segment $[u_1, u_2]$ we can never re-enter a bounding half-space we previously left. So L is a finite ordered list. We can also say that if $[u_1, u_2]$ passes through the interior of $C_D(b_j)$ then $(a + b_j)$ is a Minkowski vertex in C. On the other side if $[u_1, u_2]$ intersects $C_D(b_j)$ only with its frontier then as $[u_1, u_2] \subset Int(C_D(a))$ it is easy to build a point still inside $C_D(a)$ that will also be in the interior of $C_D(b_j)$, we only have to shift away from $[u_1, u_2]$ by a very small quantity. In all the cases $(a + b_j) \in \mathcal{V}_C$, moreover two consecutive dual cones in L are connected, so are their corresponding vertices in B. So L is a list having for first and last elements b_1 and b_2 connected through neighbour vertices $\{b_j, b_j \in \mathcal{V}_B\}$ such that $(a + b_j) \in \mathcal{V}_C$. So $\mathcal{V}_B^+(a)$ is connected.

Algorithm 2 Calculate $C = A + B$ with A and B, two $\mathbb{R}^n$-polytopes of full dimension n

Require: List of dual cones of A $\{C_D(a_i), a_i \in \mathcal{V}_A\}$
Require: List of dual cones of B $\{C_D(b_j), b_j \in \mathcal{V}_B\}$
  for all $a_i \in \mathcal{V}_A$ **do**
    $findMinkVertex = false$
    while $b_j \in \mathcal{V}_B$ and $findMinkVertex == false$ **do**
      Compute $C_D(c_{ij}) = C_D(a_i) \cap C_D(b_j)$
      if $dim(C_D(c_{ij})) = n$ **then**
        $findMinkVertex = true$
        $c_{ij} = (a_i + b_j) \in \mathcal{V}_C$
        // Now find a_i polyhedral cap.
        ProcessNeighbours$(C_D(a_i), C_D(b_j), C_D(c_{ij}))$
      end if
    end while
  end for

// Find neighbours through common facets
PROCEDURE
ProcessNeighbours$(C_D(a_i), C_D(b_j), C_D(c_{ij}))$
Mark $C_D(b_j)$ as processed
for all $C_D(b_k) \in N(B)$, F_l facet of $C_D(c_{ij})$ such that $F_l = C_D(b_k) \cap C_D(c_{ij})$ **do**
  // At this step b_j and b_k are neighbours in B
  if $C_D(b_k)$ is not marked as processed **then**
    Compute $C_D(c_{ik}) = C_D(a_i) \cap C_D(b_k)$
    if $dim(C_D(c_{ik})) = n$ **then**
      $c_{ik} = (a_i + b_k) \in \mathcal{V}_C$
      ProcessNeighbours$(C_D(a_i), C_D(b_k), C_D(c_{ik}))$
    end if
  end if
end for
END PROCEDURE

4 Minkowski sum of polytopes with primal cones

We will now consider the primal cones of the polytopes we handle that is to say, for each vertex $a \in \mathcal{V}_A$, the cones defined by the intersection of the half-spaces $\{\bar{H}_u^+(a)\}$ attached to this current vertex. In an equivalent manner, we can say that, given $a \in \mathcal{V}_A$, such cones can be generated by all the edges attached to a and pointing towards its neighbour vertices $\{a_i\}$. Let's write $\mathcal{V}_A(a)$ the set of the vertices of $\mathcal{V}_A$ adjacent to a, with $\alpha_i \in \mathbb{R}^+, a_i \in \mathcal{V}_A(a)$

$$C(a) = a + \sum_i \alpha_i(a_i - a) = \bigcap_u \bar{H}_u^+(a). \tag{11}$$

4.1 Main properties

Let C_1 and C_2 be two cones attached to the origin:

$$C_1 + C_2 = Conv(C_1, C_2). \tag{12}$$

Let $x \in C_1 + C_2$ then $\exists y \in C_1, \exists z \in C_2$ such as $x = y + z = \frac{1}{2}2y + \frac{1}{2}2z$. Yet $2y \in C_1$ and $2z \in C_2$ which means that any point in C_1+C_2 can be written as a point included in the convex hull of C_1 and C_2 hence $C_1 + C_2 \subset Conv(C_1, C_2)$.

Let $x \in Conv(C_1, C_2)$ then $\exists y \in C_1, \exists z \in C_2$ and $\theta \in [0, 1]$ such as $x = \theta y + (1 - \theta)z$. Yet $\theta \geq 0$ so $\theta y \in C_1$, $1 - \theta \geq 0$ hence $(1 - \theta)z \in C_2$ which means that x can be written as the sum of two elements from A and B so $Conv(C_1, C_2) \subset C_1 + C_2$.

It is easy to transpose this property to the case of two cones attached to the vertices a and b in the polytopes A and B provided that we translate them first in $a + b$ and hereafter compute the convex hull:

$$C(a) + C(b) = Conv\Big(b + C(a), a + C(b)\Big). \tag{13}$$

If $c = a + b$ is a Minkowski vertex of $C = A + B$ then:

$$C(c) = Conv\Big(b + C(a), a + C(b)\Big). \tag{14}$$

From [6] if c and c' are two adjacent vertices in C with their given decomposition in elements of A and B, $c = a + b$ et $c' = a' + b'$ then a et a' are either equal or adjacent (respectively b and b'). We deduce that the list of edges defining $C(c)$ is a sublist of $L(C(a), C(b))$, defined as the list of edges of both $C(a)$ and $C(b)$ translated in c. So $C(c) \subset Conv(b + C(a), a + C(b))$ because $Conv(b + C(a), a + C(b))$ is entirely defined by $L(C(a), C(b))$.

For the reciprocal we use the dual, we know that $C_D(c) = C_D(a) \cap C_D(b)$ so

$$\begin{cases} C_D(c) \subset C_D(a) \\ C_D(c) \subset C_D(b). \end{cases}$$

These relations are reversed in the primal space:

$$\begin{cases} b + C(a) \subset C(c) \\ a + C(b) \subset C(c). \end{cases}$$

Hence $Conv(b + C(a), a + C(b)) \subset C(c)$ because $C(c)$ is convex. Now we can give the property linking primal and dual cones respectively attached to the vertices $a_i \in \mathcal{V}_A$ and $b_j \in \mathcal{V}_B$ when their sum provide a Minkowski vertex:

$c \in \mathcal{V}_C \Rightarrow$

$$\Big(C_D(a) \cap C_D(b)\Big)_D = Conv\Big(b + C(a), a + C(b)\Big). \tag{15}$$

The proof is quite straightforward, as c is a Minkowski vertex then $C_D(c) = C_D(a) \cap C_D(b)$ and the dual of the dual is the primal so $(C_D(c))_D = C(c) = Conv(b + C(a), a + C(b))$.

This property is fundamental in the sense that it can make the connection between a polyhedra intersection problem on one side, and a polyhedra convex hull computation on the other. In the context of the sums of polytopes we're aware that if $c = a + b$ is a Minkowski vertex of $C = A + B$ then computing the convex hull of the two primal cones $C(a)$ and $C(b)$ is equivalent to computing the intersection between their corresponding duals $C_D(a)$ and $C_D(b)$, see Fig. 2. As a consequence we can compute $C(c)$ $\forall c \in \mathcal{V}_C$ - which means we can find all facets of C - with data coming only from the primal space. Given that a polytope is entirely determined by its vertices or facets we can write the following property for A and B two $\mathbb{R}^n$-polytopes with respectively k and l vertices. Let's note K the set of indices that provide a Minkowski vertex in $C = A + B$ i.e. $(i, j) \in K_{A+B} \Leftrightarrow (a_i + b_j) \in \mathcal{V}_C$.

$$A + B = \bigcap_{i=1}^{k} C(a_i) + \bigcap_{j=1}^{l} C(b_j) = \bigcap_{(i,j)\in K} C(a_i) + C(b_j) \tag{16}$$

We can generalize this property to all the sums of vertices as it is easy to proove that whatever $a_i \in \mathcal{V}_A$

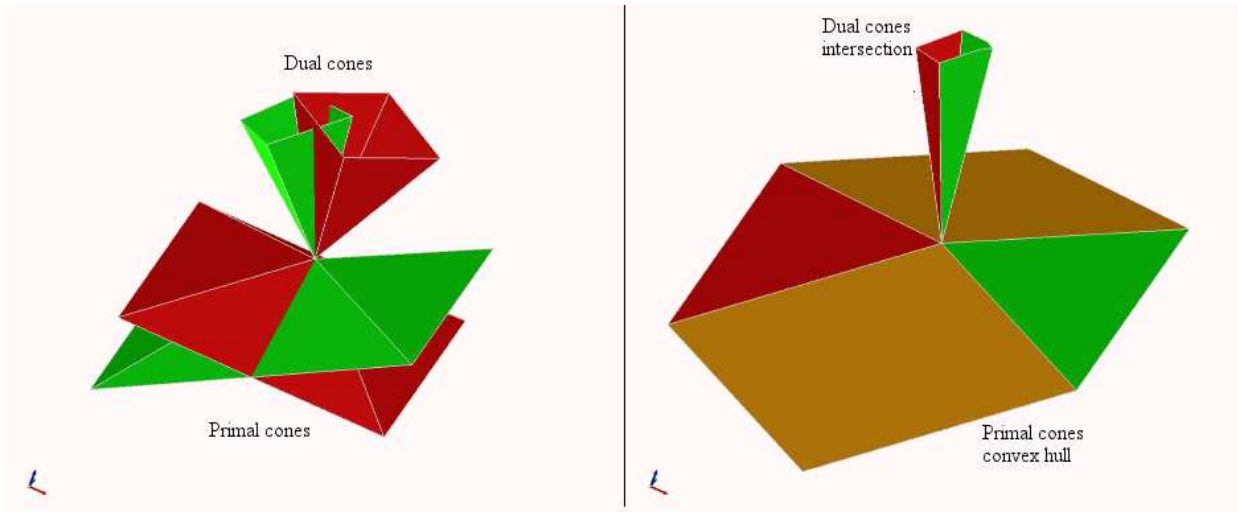

Figure 2: On the left two primal cones on the bottom with their duals on top of the drawing, on the right the primal cones convex hull and the duals intersection.

and $b_j \in \mathcal{V}_B$, $A+B \subset C(a_i)+C(b_j)$ because we know that $A \subset C(a_i)$ and $B \subset C(b_j)$. So it is clear that:

$$A + B = \bigcap_{i=1,j=1}^{k,l} \Big(C(a_i) + C(b_j)\Big). \tag{17}$$

4.2 A primal algorithm

With the property 14, it is quite easy to set up an algorithm computing the polytope $C = A + B$.

Algorithm 3 Calculate $C = A + B$ with A and B, two $\mathbb{R}^n$-polytopes of full dimension n

Require: List of primal cones of A $\{C(a_i), a_i \in \mathcal{V}_A\}$
Require: List of primal cones of B $\{C(b_j), b_j \in \mathcal{V}_B\}$
Require: $a_u \in \mathcal{V}_A$ and $b_v \in \mathcal{V}_B$ such as $a_u + b_v \in \mathcal{V}_C$
ProcessCones$(C(a_u), C(b_v))$

Starting from a first Minkowski vertex $c_1 = a_1 + b_1$ we just need to compute its edges making sure they belong to the convex hull of the cone $C(c_1)$. Following such edges will leed to c_1 neighbours where we will compute their corresponding convex hulls. At this step it is important to note that, to identify Minkowski vertices, if $c_2 = a_2 + b_2$ is a neighbour if c_1 then a_1 and a_2 are either equal or adjacent in A, b_1 and b_2 are either equal or adjacent in B. The edges of C are either parallel to an edge of A or an edge of B.

PROCEDURE
ProcessCones$(C(a_i), C(b_j))$
Mark $c_{ij} = (a_i + b_j) \in \mathcal{V}_C$ as processed
Find facets, edges of $C(c_{ij}) = Conv(C(a_i), C(b_j))$
for all edges of $C(c_{ij})$ leading to $c_{uv} \in \mathcal{V}_C$ **do**
  if c_{uv} is not marked as processed **then**
    ProcessCones$(C(a_u), C(b_v))$
  end if
end for
END PROCEDURE

4.3 Complexity

We have reduced our problem to a convex hull algorithm. However as stated in [14] *"we are still very far from knowing the best ways to compute the convex hull for general dimensions"* and the author adds *"in the general case, there is no known polynomial algorithm"*. Despite the current state-of-the-art we believe we do not have only a theoritical achievement with property 17. The reason is due to the fact that it is not just about computing the convex hull of a set of edges coming from cones C_1 and C_2 but rather computing the convex hull of two sets of edges, each one of them being already convex. As building a convex set from two convex sets is easier, we plan to explore this track in the future.

5 Conclusion

We have developped and implemented a dual algorithm based on the intersection of dual cones used in the field of tolerance analysis where it behaves very well in terms of robustness and computation time. However the fact that we handle the double description to sum polytopes could possibly be a limitation if one needs to work in high dimensions. In the second part, we introduced new properties to remain in the primal space and proposed another way to perform the operation. Now we plan to work on a parallel version of the first as well as improving the theoritical background of the second.

References

[1] Lazhar Homri, Denis Teissandier, and Alex Ballu, *Tolerancing analysis by operations on polytopes*, Design and Modeling of Mechanical Systems, Djerba (Tunisia), 597:604, 2013

[2] Vijay Srinivasan, *Role of Sweeps in Tolerancing Semantics*, in ASME Proc. of the International Forum on Dimensional Tolerancing and Metrology, TS172.I5711, CRTD, 27:69-78, 1993

[3] Denis Teissandier and Vincent Delos, *Algorithm to calculate the Minkowski sums of 3-polytopes based on normal fans*, Computer-Aided Design, 43:12:1567-1576, 2011

[4] Yanyan Wu, Jamie J. Shah, and Joseph K. Davidson, *Improvements to algorithms for computing the Minkowski sum of 3-polytopes*, Computer-Aided Design, 35:13:1181-1192, 2003

[5] Vincent Delos and Denis Teissandier, *Minkowski sum of polytopes defined by their vertices*, Proceedings of the Conference on Geometry, Topology and Applications, Shangai, 2015

[6] Komei Fukuda, *From the zonotope construction to the Minkowski addition of convex polytopes*, Journal of Symbolic Computation, 38:4:1261-1272, 2004

[7] Christophe Weibel, *Minkowski sums of polytopes*, PhD Thesis, EPFL, 2007

[8] Gunter Ziegler, *Lectures on Polytopes*, Graduate Texts in Mathematics, Springer Science, 1995

[9] Peter Gritzmann and Bernd Sturmfels, *Minkowski addition of polytopes: computational complexity and applications to Grbner bases*, Siam J. Discrete Math., 6:2:246-269, 1993

[10] Komei Fukuda and Christophe Weibel, *Computing all faces of the Minkowski sum of V-Polytopes*, Proceedings of the 17th Canadian Conference on Computational Geometry, 253-256, 2005

[11] Komei Fukuda and Alain Prodon, *Double Description Method Revisited*, Combinatorics and computer science, Lecture Notes in Computer Science, 1120:91-111, 1996

[12] Bernard Chazelle, *An optimal convex hull algorithm in any fixed dimension*, Discrete and Computational Geometry, 10:377-409, 1993

[13] David Avis and David Bremner and Raimund Seidel, *How good are convex hull algorithms?*, Computational Geometry, 7:5-6:265-301, 1997

[14] Komei Fukuda, *Frequently Asked Questions in Polyhedral Computation*, `ftp://ftp.ifor.math.ethz.ch/pub/fukuda/reports/polyfaq041121.pdf`, 2004