Applying screw theory for summing sets of constraints in geometric tolerancing

Santiago Arroyave-Tobón, Denis Teissandier, Vincent Delos

Applying screw theory for summing sets of constraints in geometric tolerancing

Santiago Arroyave-Tobón, Denis Teissandier, Vincent Delos

Univ. Bordeaux, I2M, UMR 5295, F-33400 Talence, France

Abstract

In tolerance analysis, approaches based on sets of constraints (also called convex hull techniques) are able to study simultaneously all the possible extreme configurations of a mechanism when simulating manufacturing defects in its components. The accumulation of these defects can be calculated by summing and intersecting 6-dimensional sets of constraints, i.e. polyhedra. These approaches tend to be time-consuming, however, because of the complexity resulting from manipulating sets in $\mathbb{R}^6$. In this paper, polyhedra are decomposed into a bounded set (a polytope) and an unbounded set (a set of straight lines). The unbounded part of the polyhedra is characterized by the degrees of freedom of the toleranced feature or the joint. Therefore, the decomposition can be performed based on a kinematic analysis of the studied assembly using screw systems. The proposed decomposition is presented for the most common features used in geometric tolerancing. The idea behind this strategy is, instead of summing polyhedra in $\mathbb{R}^6$, to sum only their underlying polytopes by isolating the unbounded part of the operands. A slider-crank mechanism is used to show the gain in computational time of the proposed method in comparison with the strategy based on complete 6-dimensional sets of constraints.

Keywords: Manufacturing defects, Tolerance analysis, Screw theory, Degrees of freedom, Polyhedra, Minkowski sum

1. Introduction

Tolerance analysis consists in studying the impact of manufacturing defects on product behavior. The objective is to determine if the accumulation of defects in the different parts allows the ideal functioning of the product. One way to perform tolerance analysis of three-dimensional tolerance chains is by means of sets of constraints. Using these approaches, it is possible to simulate simultaneously all possible displacements (3 rotations and 3 translations) of a toleranced feature inside its tolerance zone or the relative movements of two features of two parts mated by a contact restriction [1]. They allow, additionally, the treatment of overconstrained (hyperstatic) mechanisms. Some models consider quadratic constraints (such as Domains [2] and T-Maps [3]), while others are based on linear constraints (such as Polytopes [4]) by considering manufacturing defects as small displacements [5]. A comparison of these methods is presented in [6]. As all possible configurations of each toleranced feature must be considered, the cumulative defect limits on parts in a serial configuration can be calculated by means of the Minkowski sum of sets of geometric constraints [1, 7]. In the case of parts with parallel contacts, all the tolerance zones must be satisfied and then the intersection of the respective sets of constraints is required. By means of these two operations, the stack-up of deviations in any tolerance chain can be computed [8].

The degrees of freedom of contact features or the degrees of invariance of geometric features imply unbounded displacements [9, 10]. In these cases, the respective set of constraints defines a polyhedron in $\mathbb{R}^6$. When the toleranced feature has no degrees of freedom or invariance (as in the case of a complex surface), the set of constraints represents a polytope (a bounded polyhedron).

On the one hand, Minkowski sums of polytopes have been a research subject explored with applications in biology [11], robotics [12, 13], tolerance analysis [14, 15], among others. Because of their complexity, however, the computation of sums in spaces of dimension greater than 3 is still a challenging matter. Mansuy et al. [16] propose a method for calculating separately only the most disadvantageous vertices with respect to a functional polytope. Even if this method avoids the computation of Minkowski sums, the set of computed vertices is only representative of a given functional condition. In addition, the authors just consider the case of tolerance chains made up of features of the same invariance class and in a particular relative position (i.e. a set of parallel planes). In these cases, the displacements linked to degrees of freedom or invariance can be trivially isolated from the computations.

On the other hand, for the Minkowski sums of polyhedra, just a few studies have been proposed. Homri et al. [17] suggested turning polyhedra into polytopes by introducing virtual boundaries, called cap half-spaces. This strategy has to cope with the multiplication of cap half-spaces during the computation of Minkowski sums. As a consequence, the time taken for computing cap facets (facets associated with cap half-spaces) is in general far greater than that needed for computing facets representing real limits of bounded displacements. Fukuda [18] presented an algorithm to compute Minkowski sums of polytopes, mentioning the possibility of applying the same procedure for the case of polyhedra with at least one vertex (pointed polyhedra) by treating infinite rays as points at 'infinity'. However, due to the degrees of freedom (or invariance),

the polyhedra manipulated in tolerancing usually do not have vertices. Each degree of freedom (or invariance) implies a sweeping operation of a polytope along a straight line, placing the vertices at infinity.

The set of displacements required to completely define the relative position between two features in a tolerance chain can be identified by a kinematic analysis. At a later stage, the objective of tolerance analysis is to determine the limits of these displacements. For this reason it is common to find tolerance analysis techniques in the literature involving kinematic analysis [19, 20, 21, 22]. However, it has been done just for the case of parametric techniques (which are not able to treat assemblies with redundantly suppressed degrees of freedom). The advantages of applying kinematic analysis in the case of techniques based on sets of constraints remain unexplored, and this is actually the objective of this paper.

One possible way to model rigid body motions is by means of the theory of Euclidean group displacements. This theory states that in the affine Euclidean space of dimension 3, the set of rigid transformations has an algebraic structure of a continuous group, more specifically it is a Lie group of dimension 6 [23]. Therefore, the union of two successive displacements is also a displacement [24]. Fanghella [25] presented an exhaustive list of products of subgroups. The author produced this classification by analyzing systematically all possible couples of subgroups and their geometric relations (coincidence, parallelism, intersection, etc.). Hervé [26] presented a more general procedure by using the exponential maps of the Lie groups enumerating 12 possible kinds of displacement subgroups. However, the 12 displacement subgroups cannot explain all rigid motions in space. Under most conditions, a rigid motion is simply a displacement manifold included in the displacement group but not a subgroup itself [27].

In a similar way, using the theory of screws [28] it is possible to perform spatial kinematic analysis. By computing systematically the union and intersection of screws the mobility condition of any two surfaces of a mechanical system can be determined [29, 30, 31]. Open serial and closed chains can be analyzed even if the former present redundantly suppressed degrees of freedom. For these purposes, the classification of screw systems plays an important role. Rico Martinez and Duffy [32, 33] proposed a classification based on the reciprocal basis of screw systems. A general approach for mechanisms analysis and synthesis based on this reciprocity is proposed by Dai and Jones [34].

This paper proposes the use of the theory of screws for modeling the mobility conditions of an assembly during tolerance simulation with sets of geometric constraints. It enables us to simplify the sets of constraints, decomposing them into the sum of a polytope (a bounded set) and a set of straight lines (an unbounded set). When the operand sets are decomposed, their sum can be calculated by dealing with the underlying polytopes. This implies a reduction in operand complexity and consequently a reduction in the computational time. In this work, the advantages of using the proposed model (decomposed polyhedra) are explored only in the case of sums, i.e. when the tolerance chain is made up of serial contacts.

The next section of the paper introduces the way to sum sets of constraints based on polytopes and cap half-spaces. In section 3 the new strategy based on kinematic analysis is presented. Next, in section 4, a case study is solved using the proposed method and the results are compared with those obtained by the method based on cap half-spaces. Finally, some conclusions and perspectives are discussed in section 5.

2. Tolerance analysis with polytopes of $\mathbb{R}^6$

In this section, the tolerance analysis method based on the operation with polytopes of $\mathbb{R}^6$ is presented with its mathematical support and illustrated with an example. For further details the reader can refer to [17].

Before going into the details of polyhedra and polytopes in the field of geometric tolerancing, let's present some definitions required for this paper. They are taken from [35].

Definition 2.1. *An $\mathcal{H}$-polyhedron is an intersection of many finitely closed half-spaces in some $\mathbb{R}^n$.*

Definition 2.2. *A polytope is a polyhedron that is bounded in the sense that it does not contain a ray $\{\boldsymbol{x} + t\boldsymbol{y} : t \geq 0,\ \boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n\}$ for any $\boldsymbol{y} \neq \boldsymbol{0}$. A $\mathcal{V}$-polytope is the convex hull of a finite set of points in some $\mathbb{R}^n$.*

According to the Minkowski-Weyl theorem, the $\mathcal{H}$-description (where $\mathcal{H}$ refers to half-space) and the $\mathcal{V}$-description (where $\mathcal{V}$ refers to vertex) of a polytope are equivalent. It makes the link between the two dual intuitive definitions of a polytope: one based on facets and the other based on vertices (see Figure 1).

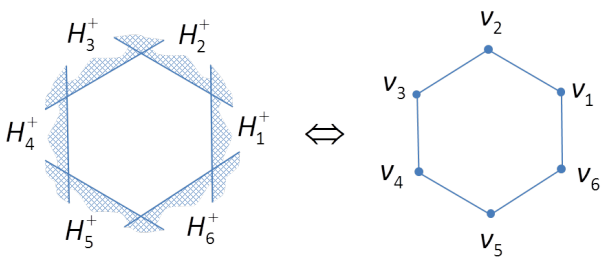


Figure 1: Two equivalent definitions of a polytope: facets or half-spaces versus vertices

Definition 2.3. *A polyhedron is pointed if it does not contain an entire line and then contains at least one vertex.*

2.1. Sets of constraints and polyhedra

Let S_1 be a substitute surface related to a nominal feature S_0 (see Figure 2). The position and orientation of S_0 can be restricted by a specification, which in turn defines a tolerance zone TZ, offsetting S_0 from d^{inf} to d^{sup}. This tolerance zone implies a restriction on the position of the points $\boldsymbol{N}_i \in S_0$:

$$S_1 \subseteq ZT \Leftrightarrow \forall \boldsymbol{N}_i \in S_0 : d^{inf} \leq \boldsymbol{t}_{N_i} \cdot \boldsymbol{n}_i \leq d^{sup} \tag{1}$$

where $\boldsymbol{t}_{N_i}$ is the translation displacement of S_1 in relation to S_0 at point $\boldsymbol{N}_i$.

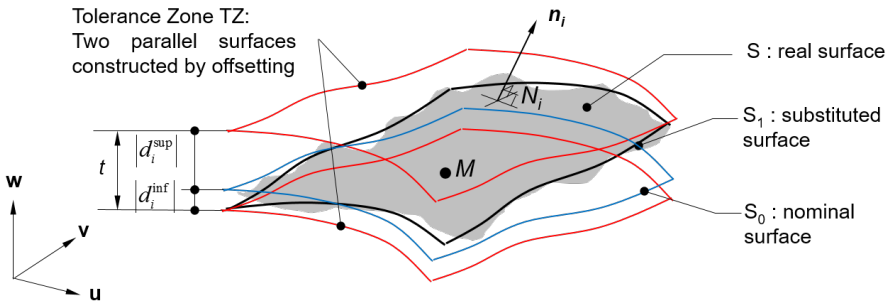


Figure 2: Toleranced feature and tolerance zone [17]

Expressing the constraints in (1) at any point $\boldsymbol{M}$ of the Euclidean space and linearizing the rotations under the assumption of small displacements we have:

$$d^{inf} \leq (\boldsymbol{t}_M + \boldsymbol{N}_i\boldsymbol{M} \times \boldsymbol{r}) \cdot \boldsymbol{n}_i \leq d^{sup} \tag{2}$$

where $\boldsymbol{r}$ is the rotation vector of S_1 in relation to S_0.

Considering a vectorial base $[u, v, w]$, we define the vectors $\boldsymbol{r} = [r_u, r_v, r_w]^T$, $\boldsymbol{t_M} = [t_u, t_v, t_w]^T$, $\boldsymbol{N_iM} = [d_{iu}, d_{iv}, d_{iw}]^T$, $\boldsymbol{n}_i = [n_{iu}, n_{iv}, n_{iw}]^T$. When expanding the vectorial and scalar products in inequality (2) and setting $r_u = x_1$, $r_v = x_2$, $r_w = x_3$, $t_u = x_4$, $t_v = x_5$ and $t_w = x_6$, we obtain:

$$d^{inf} \leq (n_{iv}d_{iw} - n_{iw}d_{iv})x_1 + (n_{iw}d_{iu} - n_{iu}d_{iw})x_2 + (n_{iu}d_{iv} - n_{iv}d_{iu})x_3 + n_{iu}x_4 + n_{iv}x_5 + n_{iw}x_6 \leq d^{sup} \tag{3}$$

These constraints can be modeled with half-spaces. For each point $\boldsymbol{N}_i \in S_0$, two parallel half-spaces are obtained. Then, if a finite set of m points $\boldsymbol{N}_i$ is considered, a set of $2m$ constraints is obtained [17]. The intersection of these constraints defines a convex $\mathcal{H}$-polyhedron in $\mathbb{R}^6$ [4]:

$$\Gamma = \bigcap_{k=1}^{2m} \bar{H}_k^- \text{ where } \bar{H}_k^- = \left\{\boldsymbol{x} \in \mathbb{R}^6 : b_k + a_{k1}x_1 + ... + a_{k6}x_6 \geq 0\right\} \text{ for all } k \in \{1, ..., 2m\} \tag{4}$$

In a similar way, a polyhedron representing the allowable displacements of a couple of features potentially in contact can be defined. In this case, the toleranced feature is defined in the nominal case of permanent contact between the features and the tolerance zone is obtained by offsetting the toleranced feature according to the clearance value.

Finally, a functional polytope can be created with the same procedure that defines a geometric specification between two surfaces of the same part.

Except for the case of complex surfaces, the set of constraints derived from a toleranced feature (Eq. (4)) defines an unbounded set in the 6-dimensional space of deviations, i.e. a polyhedron. This is a consequence of the degrees of invariance of the toleranced features or the degrees of freedom of the joints which define unbounded displacements. These polyhedra are generally not pointed (according to definition 2.3). As a consequence, they do not have vertices and thus, the computation of double descriptions and Minkowski sums becomes more complicated.

2.2. Polytopes and cap half-spaces

Due to the aforementioned difficulties treating polyhedra, Homri et al. [17] suggested turning them into polytopes introducing virtual boundaries called cap facets, $\bar{H}c$:

$$\Gamma' = \left(\bigcap_{k=1}^{2m} \bar{H}_k^-\right) \cap \left(\bigcap_{j=1}^{2d} \bar{H}c_j^-\right) = \Gamma \cap \left(\bigcap_{j=1}^{2d} \bar{H}c_j^-\right) \tag{5}$$

where d is the number of degrees of invariance (or freedom) of the toleranced (or kinematic) feature. Notice that the strict minimum of cap facets is introduced to each polyhedron. An example of the introduction of cap facets is presented in Figure 3. In this figure a 6D polyhedron derived from the geometric constraints of a plane surface is depicted in two partial 3D representations. The bottom left polyhedron is a partial visualization of the rotations and the right is one of the translations. As the tolerance zone imposes limits on r_x, r_y and t_z, cap facets are required to virtually limit the unbounded displacements

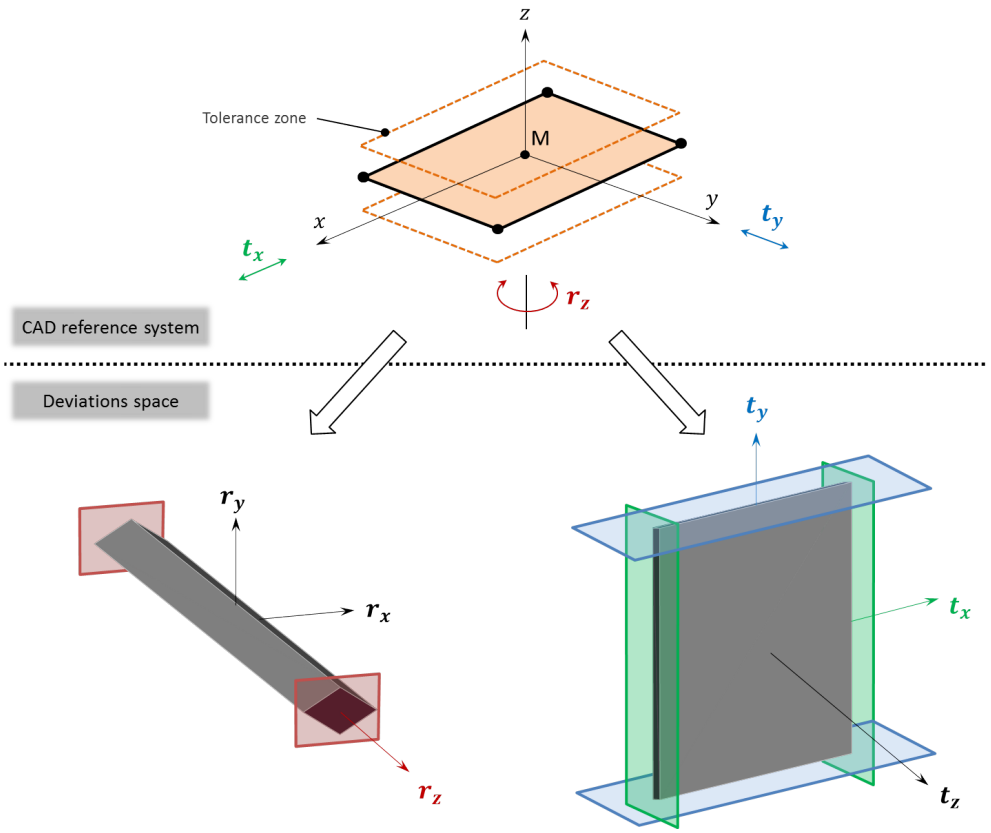


Figure 3: Cap half-spaces to virtually limit r_z, t_x and t_y depicted in red, blue and green respectively.

r_z (in the polyhedron at the right), t_x and t_y (in the polyhedron at the left). The decision to add cap facets to obtain bounded sets is only for to algorithmic reasons.

The only issue arising from the method based on cap facets is that the topology of the operand polytopes is affected, increasing its complexity. Although such an issue does not influence the results from the tolerancing point of view, it does involve spending time calculating meaningless information. During a simulation, this problem increases after each sum due to the accumulation of the degrees of freedom along the toleranced chains. Therefore, after the first sum, the definition of the set of cap facets of the operand polytopes is no longer minimal. As a consequence, the time for computing cap facets is in general far greater than that for computing significant facets.

2.3. Example of tolerance analysis with polytopes of $\mathbb{R}^6$

A tolerance analysis of a sub-assembly of a slider-crank mechanism is presented below in order to illustrate the method based on polytopes and cap half-spaces. Later, in section 4, the whole assembly is addressed comparing the actual approach with the new enhanced method. As depicted in Figure 4, the functional condition FC1 implies the control of the relative position of the inferior cylindrical surface of the connecting rod and the upper plane surface of the piston. These surfaces are designated 1,2 and 3,2 in Figure 5. Contact and geometrical defects in the involved parts are modelled with polytopes for calculating the stack-up of deviations.

The topological model of the mechanism is depicted by the contact graph in Figure 5b. The nodes designated as α, β represent the nominal model of a part when $\beta = 0$, and the substitute surfaces when $\beta \neq 0$. Each edge of the graph represents some deviations. These may be geometric deviations, in the case of inner edges, or deviations due to contacts, in the case of edges connecting two nodes from different parts. These deviations can be represented by geometric and contact polyhedra respectively [17].

By analyzing the contact graph, the set of operations required to simulate the relative position of the surfaces involved by the functional condition can be determined. The following relation can then be deduced:

$$\Gamma_{1,2/3,2} = \Gamma_{1,2/1,0} \oplus \Gamma_{1,0/1,1} \oplus \Gamma_{1,1/2,1} \oplus \Gamma_{2,1/3,1} \oplus \Gamma_{3,1/3,0} \oplus \Gamma_{3,0/3,2} \tag{6}$$

where $\Gamma_{f/g}$ is the polyhedron describing the relative position of the surface f with respect to the surface g.

In Eq. (6) some polyhedra are defined over the same surface and consequently they are homothetic. This is the case for polyhedra $\Gamma_{1,0/1,1}$ and $\Gamma_{1,1/2,1}$ and polytopes $\Gamma_{2,1/3,1}$ and $\Gamma_{3,1/3,0}$. The sum of homothetic polyhedra can be performed directly by homothetic transformations and no numerical computation is needed. Then, Eq. (6) can be simplified as:

$$\Gamma_{1,2/3,2} = \Gamma_{1,2/1,0} \oplus \Gamma_{1,0/2,1} \oplus \Gamma_{2,1/3,0} \oplus \Gamma_{3,0/3,2} \tag{7}$$

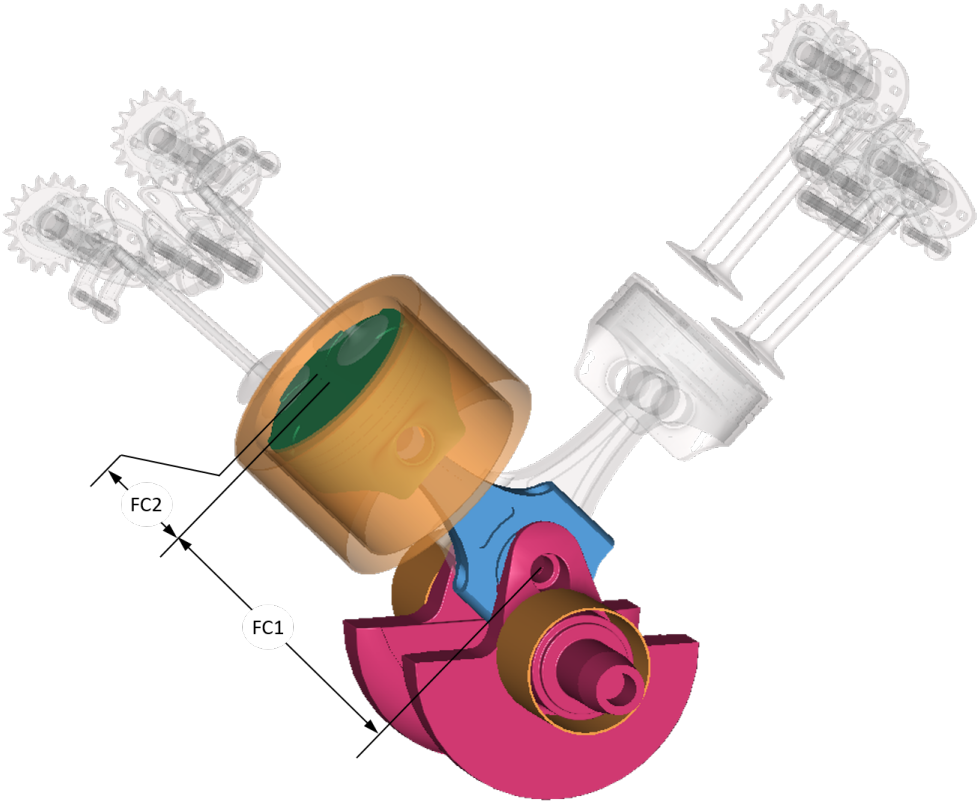
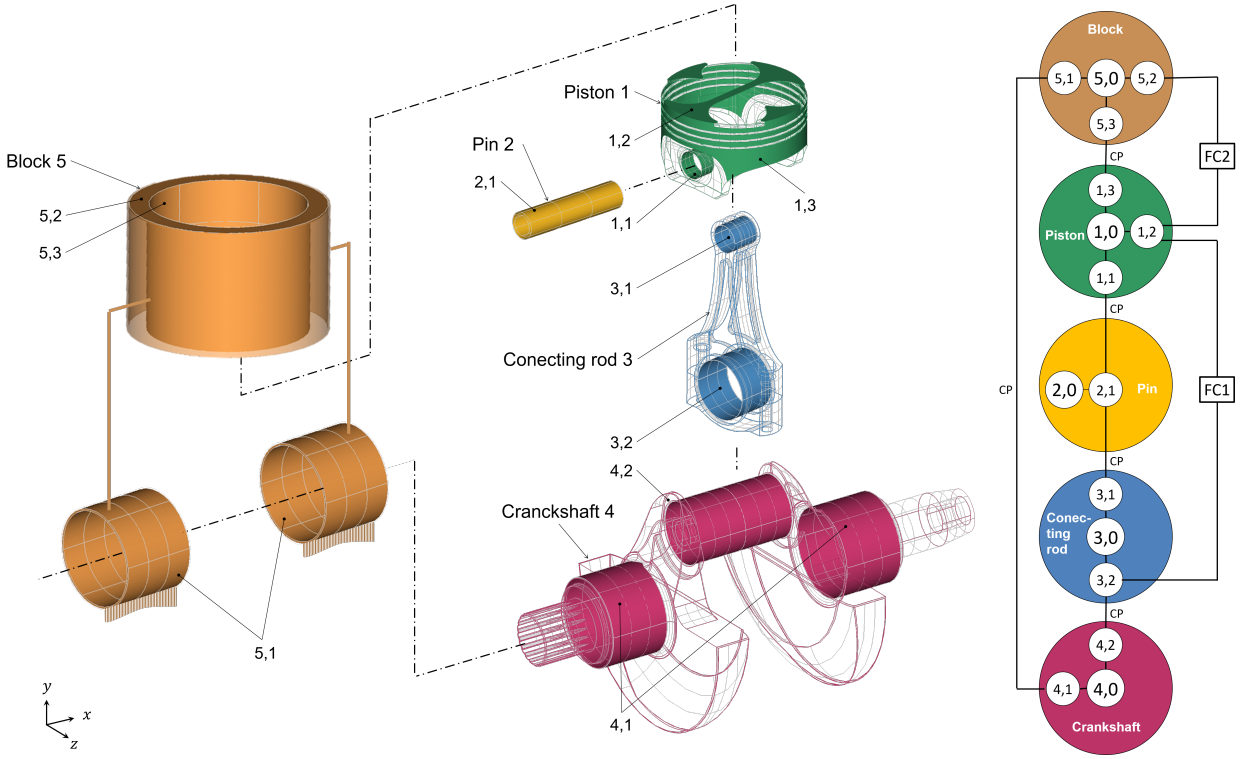


Figure 4: Proposed case study: slider-crank mechanism



(a) Parts and faces enumeration

(b) Contact graph

Figure 5: CAD and topological model of the assembly (CP: cylindrical pair)

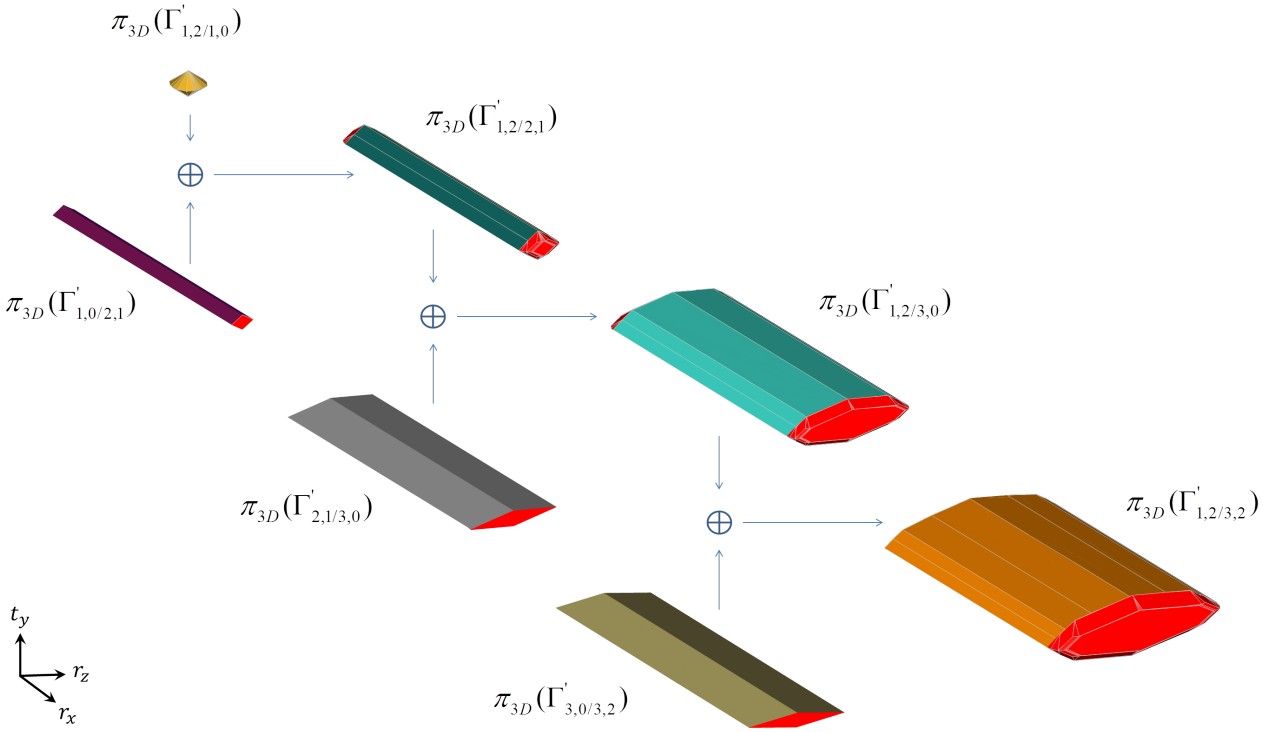


Figure 6: 3D representation of simulation of FC1 in 6D. Cap facets are shown in red.

In order to calculate $\Gamma_{1,2/3,2}$, the strategy proposed by Homri et al. [17] suggests the introduction of cap facets to the initial operand polyhedra in order to bound them and obtain polytopes. Polytopes $\Gamma'_{1,0/2,1}$, $\Gamma'_{2,1/3,0}$ and $\Gamma'_{3,0/3,2}$ (from CAD cylindrical surfaces oriented along the x-axis) required two pairs of cap facets to limit rotation and translation along the x-axis (hereafter r_x and t_x respectively). Three pairs of cap facets were introduced to $\Gamma'_{1,0/1,2}$ (defined from a plane surface with normal along y-axis) to impose virtual limits along t_x, t_z and r_y. The truncation of the operands was required to obtain their list of vertices (see [8] for further details). After obtaining all the operands in $\mathcal{V}$-description and belonging to the same space, the Minkowski sums of Eq. (7) were computed. The computations were performed with the open source library politopix: `http://i2m.u-bordeaux.fr/politopix.html`.

Figure 7 shows a 3D representation of the first sum of the simulation. It is worth mentioning that some of the figures showing 3D polytopes in this paper are just partial representations since their original polytopes belong to spaces with a dimension larger than three. A projection into a three-dimensional subspace was therefore carried out. For this case, the operands were calculated at the centroid of surface 2,1 and were then projected to the subspace spanned by $[r_x, r_z, t_y]$ for visualization purposes. Notice that from a strict minimum set of cap facets (facets in red) in the operands, originally required just to close the sets, many cap facets and consequently many unnecessary vertices appear in the calculated polytope $\Gamma_{1,2/2,1}$. The result of this sum, which models the possible deviations of a plane surface oriented along the y-axis (face 1,2) with respect to a cylindrical surface oriented along the x-axis (face 2,1), shows that only two displacements can be controlled: r_z and t_y (this conclusion is justified further on with a kinematic analysis). With this strategy, however, many vertices are computed along the directions of unbounded displacements: r_x, r_y, t_x and t_z in this case. This situation is clearly illustrated in Figure 7, where many unnecessary facets where calculated along r_x.

When the result of a sum is used again as an operand for a subsequent sum, this issue becomes worse since the operands are already highly 'contaminated' with vertices from cap facets. Notice that at the lats step of the simulation, for which the results are presented in Table 1, the calculated polytope is made up of 99.9% of cap facets. This can be justified by the accumulation and propagation of the degrees of freedom along the tolerance chain. The cap facets were identified following the strategy proposed in [36].

Table 1: Summary of the computations in $\mathbb{R}^6$ for FC1 performed with an Intel Core i7-3740QM.

Polytope	Total number of facets	Percentage of cap facets	Vertices	Time [s]
$\Gamma'_{1,2/1,0}$	50	12%	192	-
$\Gamma'_{1,0/2,1}$	20	20%	256	-
$\Gamma'_{2,1/3,0}$	20	20%	256	-
$\Gamma'_{3,0/3,2}$	20	20%	256	-
$\Gamma'_{1,2/2,1} = \Gamma'_{1,2/1,0} \oplus \Gamma'_{1,0/2,1}$	1144	99.6%	4032	25
$\Gamma'_{1,2/3,0} = \Gamma'_{1,2/2,1} \oplus \Gamma'_{2,1/3,0}$	5208	99.8%	16320	3730
$\Gamma'_{1,2/3,2} = \Gamma'_{1,2/3,0} \oplus \Gamma'_{3,0/3,2}$	19810	99.9%	55808	58870

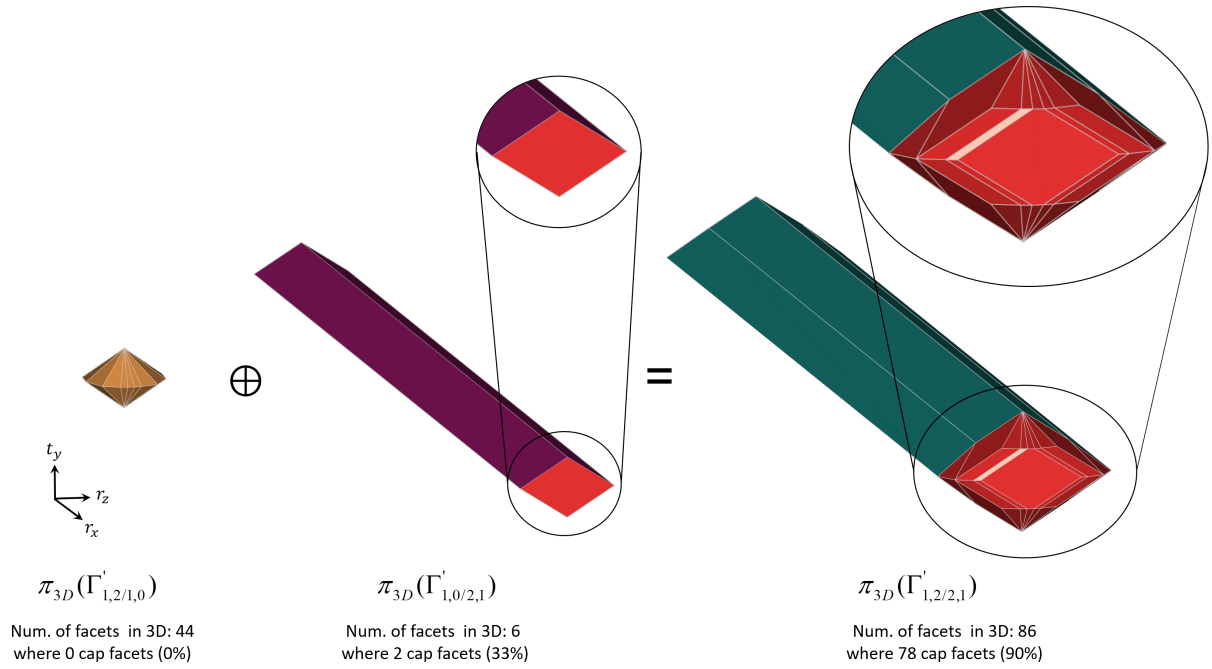


Figure 7: 3D representation of the first sum in $\mathbb{R}^6$ for FC1. Cap facets are shown in red.

The above results show clearly a situation to be considered to reduce the computational times. The main issue is how to avoid computing worthless information (cap facets) in tolerance analysis with sets of constraints. One possible solution is presented in the next section, where a new method is described for summing polyhedra based on the decomposition of the operand polytopes.

3. On the sum of decomposed polyhedra

According to the Minkowski-Weyl theorem [35], a polyhedron can be decomposed into the sum of a bounded part and an unbounded part, namely a polytope $\mathcal{P}$ and a polyhedral cone C:

$$\Gamma = \mathcal{P} \oplus C \tag{8}$$

The polyhedral cone of a polyhedron can be identified considering the degrees of invariance of the toleranced feature or the degrees of freedom of the joint. Each of these unbounded displacements characterizes a straight line Δ_j, in $\mathbb{R}^6$:

$$C = \sum_{j=1}^{d} \Delta_j \text{ with } d \leq 6 \tag{9}$$

It is along these straight lines that the underlying polytope (derived from geometric or contact constraints) is 'extruded' to generate a polyhedron (see Figure 8).

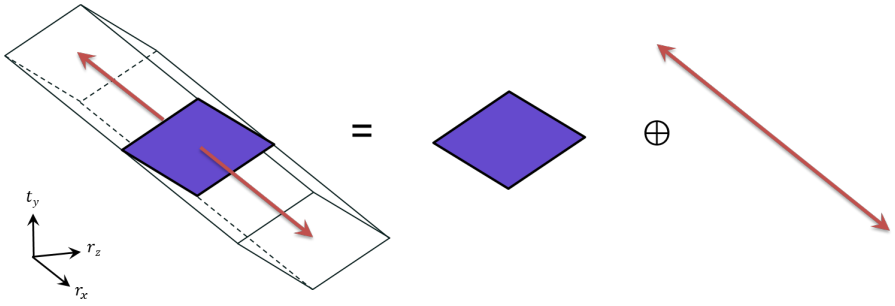


Figure 8: Decomposition of a polyhedron into the sum of a polytope and a straight line

3.1. Identification of unbounded displacements by kinematic analysis

Because of their linear properties, screw systems are a very suitable tool for analyzing the instantaneous kinematics of assemblies for tolerance analysis purposes. According to this theory, an allowable motion can be represented as a twist and

a restricted motion as a wrench [37]. Both twists $\hat{\boldsymbol{T}}$ and wrenches $\hat{\boldsymbol{W}}$ are 1x6 vectors, written as [38]:

$$\hat{\boldsymbol{T}} = [\boldsymbol{\omega} \mid \boldsymbol{v}] = [\boldsymbol{\omega} \mid \boldsymbol{r} \times \boldsymbol{\omega}] \tag{10}$$

$$\hat{\boldsymbol{W}} = [\boldsymbol{f} \mid \boldsymbol{m}] = [\boldsymbol{f} \mid \boldsymbol{r} \times \boldsymbol{f}] \tag{11}$$

where $\boldsymbol{\omega}$ is a unit angular velocity vector, $\boldsymbol{v}$ is a unit linear velocity vector, $\boldsymbol{f}$ is a unit force vector, $\boldsymbol{m}$ is unit moment vector about the point $\boldsymbol{r}$. The advantage of using screws is that the mobility analysis can be performed just by using the geometric parameters of the features.

The degrees of freedom of a joint characterize a subspace of $\mathbb{R}^6$: the twist-space. This subspace is defined by the set of d twists describing the mobility condition of the feature [39]:

$$\mathcal{T} = \begin{bmatrix} \hat{\boldsymbol{T}}_1 \\ \hat{\boldsymbol{T}}_2 \\ \dots \\ \hat{\boldsymbol{T}}_d \end{bmatrix} \tag{12}$$

$\mathcal{T}$ is usually called twist-matrix and its dual vector space, represented by a wrench-matrix, corresponds to the subspace of bounded displacements: the wrench-space. As the twist-space and the wrench-space are orthogonal subspaces, it is possible to compute a wrench-matrix from a twist-matrix and vice versa. This property, called reciprocity of screws, is one of the most important properties of this theory since changing from one subspace to the other is made easy. This duality is valid if and only if the virtual work of the wrenches on the twists is zero. It can be said if it is assumed that the constrained parts are rigid, the contact surfaces are frictionless and the contact between parts does not break [30].

The mobility conditions of a set of features can describe as well as it is done for a single one. If the features are assembled in a serial configuration, the twist-matrices must be joined together, as in Eq. (12). When the features are assembled in a parallel configuration (as in the case of parts mated with multiple contacts), the set of unbounded displacements can be calculated as the intersection of the twists, or more simply, as the union of the corresponding wrenches [39, 38]. Then, by means of unions and intersections of screws, the mobility conditions of any couple of faces in a tolerance chain can be characterized.

A screw can be interpreted as a vector in the 6-dimensional space of Euclidean motions [40]. For a given feature, a twist can be seen as the director vector of a straight line Δ_j in $\mathbb{R}^6$ characterizing a non-restricted displacement, as depicted in Figure 8. By duality, this vector represents a hyperplane. In turn, the intersection of the hyperplanes orthogonal to each of the twists of a feature characterizes the wrench-space. As the geometric constraints obtained in tolerancing are limits in the directions of bounded displacements, they define a polytope in the wrench-space. The sweeping of this polytope along the straight lines of the non-restricted displacement defines an operand polyhedron in $\mathbb{R}^6$.

3.2. Polyhedra decomposition

In accordance with the above, the mobility conditions of a kinematic joint can be identified according to its degrees of freedom. We distinguish three types of kinematic joints according to their type contact feature: surfaces, lines and point. The contact feature is defined in the nominal case of permanent contact between the surfaces of the kinematic joint [17].

According to the ISO standards [41], we consider in the case of surfacic contact features: spherical, planar, cylindrical, revolute and prismatic pairs. The straight lines characterizing the unbounded displacements for these cases are presented in Table 2. The helical pair is not treated as it is not of interest for geometric tolerancing. In the case of linear contact features, we consider ball-and-cylinder and cylinder-and-plane pairs. Their decomposition is given in Table 3. In the case of punctual contact feature, we consider the ball-and-plane pair (see Table 4). Even if the same reasoning can be applied for the exhaustive list of cases, we only address those of interest in the subject of tolerancing.

As a toleranced feature has no mobility, it is possible to see it as a joint (usually called internal joint) mating the nominal feature and the substitute one. The mobility in this case is characterized by the set of displacements leaving the toleranced feature invariant in the tolerance zone. In other words, the degree of invariance of an internal joint (i.e. a toleranced feature) can be modelled the same way we model the degree of freedom of a kinematic joint. Tolerance chains generally involve internal and kinematic joints alternatively. The general procedure for identifing the unbounded displacements in a tolerance chain is summarized in Figure 9.

The invariance degree of a surface is defined according to 7 invariance classes [42, 43]. We consider spherical, plane, cylindrical, of revolution, prismatic and complex classes (see Table 2). Like the helical pair, the helical class is not considered because it is not of interest in geometric tolerancing.

Once the straight lines of the polyhedron are identified, the underlying polytope can be obtained. This can be done by intersecting the original set of constraints (which is obtained by applying double inequality (3) to a set of points belonging to a toleranced feature) with a set of hyperplanes H_{Δ_j} such that $H_{\Delta_j} \perp \Delta_j \; \forall j \in \{1, ..., d\}$. Tables 2, 3 and 4 present this decomposition for sets of constraints derived from surfaces, lines and a point, respectively.

3.3. Summing the underlying polytopes of two polyhedra

The decomposition of operand polyhedra proposed in the previous section gives way to simplified techniques for calculating the accumulation of manufacturing defects in mechanical systems. Next, instead of operating polyhedra in $\mathbb{R}^6$ or

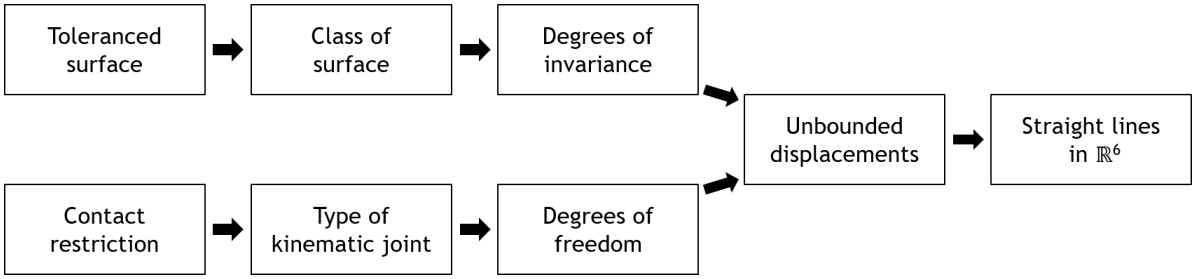


Figure 9: Flow chart of the decomposition of polyhedra

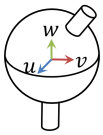
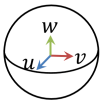
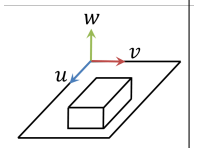
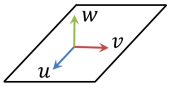
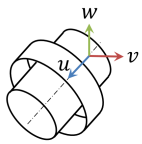
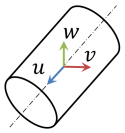
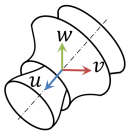
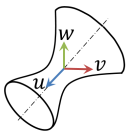
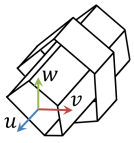
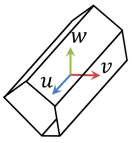
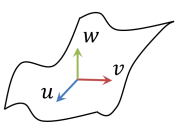
Kinematic joint	Class of toleranced surface	Tolerancing constraints	Lines of unbounded displacements
Spherical pair 	Spherical surface 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1} \bigcap H_{\Delta_2} \bigcap H_{\Delta_3}$	$\Delta_1 = \{\boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{u}]\}$ $\Delta_2 = \{\boldsymbol{x} = \lambda \hat{T}_{r_v} = \lambda[\boldsymbol{v} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{v}]\}$ $\Delta_3 = \{\boldsymbol{x} = \lambda \hat{T}_{r_w} = \lambda[\boldsymbol{w} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{w}]\}$
Planar pair 	Plane surface 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_3} \bigcap H_{\Delta_4} \bigcap H_{\Delta_5}$	$\Delta_3 = \{\boldsymbol{x} = \lambda \hat{T}_{r_w} = \lambda[\boldsymbol{w} \mid \boldsymbol{0}]\}$ $\Delta_4 = \{\boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}]\}$ $\Delta_5 = \{\boldsymbol{x} = \lambda \hat{T}_{t_v} = \lambda[\boldsymbol{0} \mid \boldsymbol{v}]\}$
Cylindrical pair 	Cylindrical surface 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1} \bigcap H_{\Delta_4}$	$\Delta_1 = \{\boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{u}]\}$ $\Delta_4 = \{\boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}]\}$
Revolute pair 	Surface of revolution 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1}$	$\Delta_1 = \{\boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{u}]\}$
Prismatic pair 	Prismatic surface 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_4}$	$\Delta_4 = \{\boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}]\}$
N/A	Complex surface 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^-$	N/A

Table 2: Decomposition of polyhedra derived from surfaces, $\boldsymbol{x} = \{x_1, ..., x_6\} \in \mathbb{R}^6, \lambda \in \mathbb{R}$.

its corresponding polytopes made up of cap facets, we propose the manipulation of simplified sets of constraints after the exclusion of the straight lines derived from the unbounded displacements (according to Tables 2, 3 and 4).

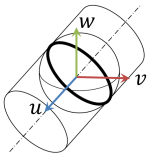
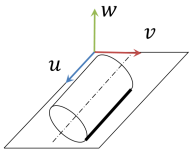
Kinematic joint	Tolerancing constraints	Lines of unbounded displacements
Ball-and-cylinder pair 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1} \bigcap H_{\Delta_2} \bigcap H_{\Delta_3} \bigcap H_{\Delta_4}$	$\Delta_1 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{u}] \right\}$ $\Delta_2 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_v} = \lambda[\boldsymbol{v} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{v}] \right\}$ $\Delta_3 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_w} = \lambda[\boldsymbol{w} \mid \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{w}] \right\}$ $\Delta_4 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}], \right\}$
Cylinder-and-plane pair 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1} \bigcap H_{\Delta_3} \bigcap H_{\Delta_4} \bigcap H_{\Delta_5}$ with $m = 2$	$\Delta_1 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u}, \boldsymbol{N}_i \boldsymbol{M} \times \boldsymbol{u}] \right\}$ $\Delta_3 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_w} = \lambda[\boldsymbol{w} \mid \boldsymbol{0}] \right\}$ $\Delta_4 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}] \right\}$ $\Delta_5 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{t_v} = \lambda[\boldsymbol{0} \mid \boldsymbol{v}] \right\}$

Table 3: Decomposition of polyhedra derived from linear features, $\boldsymbol{x} = \{x_1, ..., x_6\} \in \mathbb{R}^6, \lambda \in \mathbb{R}$.

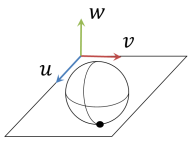
Kinematic joint	Tolerancing constraints	Lines of unbounded displacements
Ball-and-plane pair 	$\mathcal{P} = \bigcap_{k=1}^{2m} \bar{H}_k^- \bigcap H_{\Delta_1} \bigcap H_{\Delta_2} \bigcap H_{\Delta_3} \bigcap H_{\Delta_4} \bigcap H_{\Delta_5}$ with $m = 1$	$\Delta_1 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_u} = \lambda[\boldsymbol{u} \mid \boldsymbol{0}] \right\}$ $\Delta_2 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_v} = \lambda[\boldsymbol{v} \mid \boldsymbol{0}] \right\}$ $\Delta_3 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{r_w} = \lambda[\boldsymbol{w} \mid \boldsymbol{0}] \right\}$ $\Delta_4 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{t_u} = \lambda[\boldsymbol{0} \mid \boldsymbol{u}] \right\}$ $\Delta_5 = \left\{ \boldsymbol{x} = \lambda \hat{T}_{t_v} = \lambda[\boldsymbol{0} \mid \boldsymbol{v}] \right\}$

Table 4: Decomposition of polyhedra derived from punctual contacts, $\boldsymbol{x} = \{x_1, ..., x_6\} \in \mathbb{R}^6, \lambda \in \mathbb{R}$.

Let Γ_1 and Γ_2 be two polyhedra, then their sum can be calculated by summing their underlying polytopes in the smallest possible subspace. Therefore, according to Eq. (8) the following can be posed:

$$\Gamma_1 \oplus \Gamma_2 = \mathcal{P}_1 \oplus \mathcal{P}_2 \oplus C_1 \oplus C_2 \tag{13}$$

The idea is then to isolate the two polyhedral cones C_1 and C_2 from the sum to reduce the complexity. The sum $\mathcal{P}_1 + \mathcal{P}_2$ is, in general, a sum of two non-full-dimensional polytopes of $\mathbb{R}^n$ (with $1 \leq n \leq 6$). From Eq. (13) the synthesis of an algorithm to compute the sum of decomposed polyhedra is straightforward:

Algorithm 1 Sum of decomposed polyhedra

Require: Γ_A, Γ_B
Ensure: $\Gamma_S = \Gamma_A \oplus \Gamma_B$
decompose $\Gamma_A = \mathcal{P}_A \oplus C_A$
decompose $\Gamma_B = \mathcal{P}_B \oplus C_B$
if $\dim(C_A \oplus C_B) = 6$ **then**
  $\Gamma_S = \mathbb{R}^6$
else
  compute $\mathcal{V}$-description $\mathcal{V}_A = conv(a_1, ..., a_r)$
  compute $\mathcal{V}$-description $\mathcal{V}_B = conv(b_1, ..., b_s)$
  compute sum $\mathcal{V}_S = \mathcal{V}_A \oplus \mathcal{V}_B$
  $\Gamma_S = \mathcal{V}_S \oplus C_A \oplus C_B$
end if

Although the "natural way" to perform the sum of two polytopes is to compute the sum of all the vertices of the operands and calculate after its convex hull, this is a very time-consuming strategy. In addition, the points which are not vertices but are located on the boundary of the calculated polytope pose an additional challenge. There are some improved methods which can treat non-dimensional polytopes, such as those presented in [44] and in [18]. The first algorithm takes advantage of the property of the uniqueness of the Minkowski vertices decomposition for computing just the vertices defining the convex hull of the resulting polytope. The second one exploits the adjacency properties of the operands' vertices for computing just the Minkowski vertices of the resulting polytope.

4. Case study: slider-crank mechanism

4.1. FC1 simulation

The same case study already presented in section 2.3 was solved following the strategy based on the sum of decomposed polyhedra. The centroid of surface 2,1 was chosen as the point to express the screws for the mobility analysis. For convenience, it is the same point used to express the polytopes in Section 2.

Applying the decomposition proposed in Tables 2, 3 and 4 to the set of constraints derived from plane surface 1,2, the following can be obtained:

$$\Gamma_{1,2/1,0} = \mathcal{P}_{1,2/1,0} \oplus C_{1,2/1,0}$$

where the polyhedral cone $C_{1,2/1,0}$ is defined by the space spanned by the rows of the matrix:

$$\mathcal{T}_{1,2} = \begin{bmatrix} r_y \\ t_x \\ t_z \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

When isolating the 3-dimensional polyhedral cone $C_{1,2/1,0}$ from the operand $\Gamma_{1,2/1,0}$, the 3-dimensional polytope $\mathcal{P}_{1,2/1,0}$ was obtained. The truncation of $\mathcal{P}_{1,2/1,0}$ was performed in a 3D subspace in order to obtain its $\mathcal{V}$-description. Figure 10 shows a 3D representation of this operand. Notice in Table 5 that the number of vertices of $\mathcal{P}_{1,2/1,0}$ decreased 2^3 times with respect to the equivalent operand $\Gamma'_{1,2/1,0}$ from the strategy based on polytopes of $\mathbb{R}^6$ (see Table 1). This is due to the sweeping operation that must be performed recursively 3 times to obtain the polytope $\Gamma'_{1,2/1,0}$ in $\mathbb{R}^6$ from the polytope $\mathcal{P}_{1,2/1,0}$.

Similarly, the polyhedral cone of the operand related to surface 2,1 is characterized by the twist-matrix $\mathcal{T}_{2,1}$ presented below.

$$\mathcal{T}_{2,1} = \begin{bmatrix} r_x \\ t_x \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

In Figure 10, a 3D representation of the decomposition of this operand is given. As r_x is an unbounded displacement for the cylindrical surface 2,1, this polyhedron can be decomposed putting aside the straight line Δ_{r_x}. Finally, the list of vertices of $\mathcal{P}_{1,0/2,1}$ was calculated in a 4D subspace.

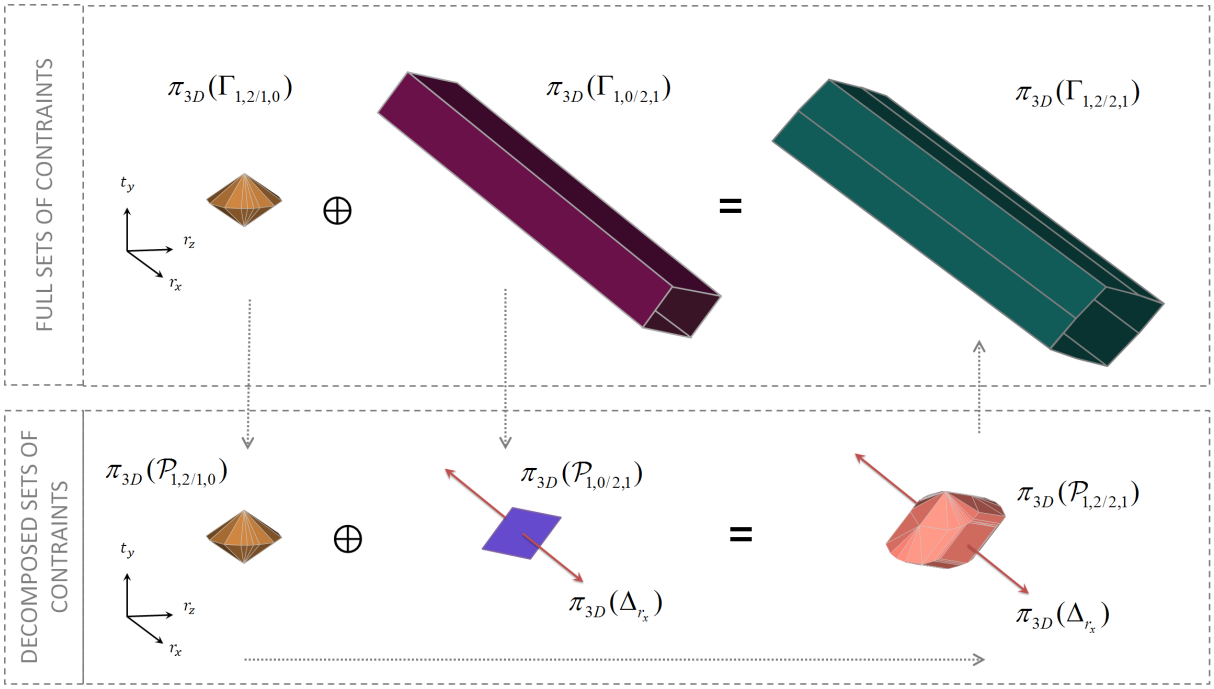


Figure 10: First sum of the computations for FC1 with decomposed polyhedra

Once we had the $\mathcal{V}$-description of the operands, the Minkowski sum $\mathcal{P}_{1,2/2,1} = \mathcal{P}_{1,2/1,0} \oplus \mathcal{P}_{1,0/2,1}$ could be performed. A summary of the results is presented in Table 5. As the manipulated operand polytopes were less complex, computing the sum took less time.

For the next sum, the set of straight lines (or the polyhedral cone) of the operand $\Gamma_{1,2/2,1}$ was calculated joining the twist-matrices $\mathcal{T}_{1,2}$ and $\mathcal{T}_{2,1}$. Then, $\mathcal{P}_{1,2/2,1}$ is the underlying polytope of the polyhedra $\Gamma_{1,2/2,1}$, of which the polyhedral cone is

characterized by $\mathcal{T}_{1,2/2,1}$.

$$\mathcal{T}_{1,2/2,1} = \begin{bmatrix} r_x \\ r_y \\ t_x \\ t_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The other two Minkowski sums in Eq. (7) were performed following the same strategy. A summary of all the simulation is presented in Table 5.

The main contribution of the proposed method can be observed in a reduction of the computation time by 97%. Furthermore, the manipulation of less complex polytopes reduces the probabilities of having numerical problems during the calculations.

It is worth mentioning that the simplification introduced by using the proposed approach keeps the set of constraints representing real limits in the associated tolerancing problem unaltered. The new method merely avoids calculating worthless information (constraints along axes of unbounded displacements). In fact, the equivalence of the results obtained by this method and the results obtained by the method based on polytopes of $\mathbb{R}^6$ was verified. This was done by projecting the results of both methods in the subspace of bounded displacements (the subspace spanned by $[r_z, t_y]$) and checking the equality of the two polytopes, as shown in Figure 11.

Table 5: Summary of the computations with decomposed polyhedra for FC1 performed with an Intel Core i7-3740QM.

Polytope	Vertices	Time [s]
$\mathcal{P}_{1,2/1,0}$	24	-
$\mathcal{P}_{1,0/2,1}$	64	-
$\mathcal{P}_{2,1/3,0}$	64	-
$\mathcal{P}_{3,0/3,2}$	64	-
$\mathcal{P}_{1,2/2,1} = \mathcal{P}_{1,2/1,0} \oplus \mathcal{P}_{1,0/2,1}$	960	0,3
$\mathcal{P}_{1,2/3,0} = \mathcal{P}_{1,2/2,1} \oplus \mathcal{P}_{2,1/3,0}$	5008	52
$\mathcal{P}_{1,2/3,2} = \mathcal{P}_{1,2/3,0} \oplus \mathcal{P}_{3,0/3,2}$	26280	1540

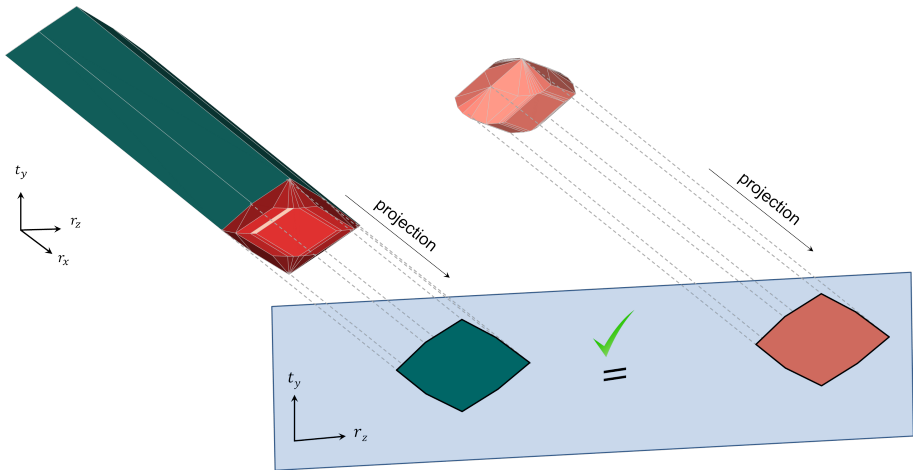


Figure 11: Comparison of the results obtained by both methods

4.2. FC2 simulation: whole crank slider mechanism

Tolerances along the whole crank slider mechanism were simulated to analyze FC2. This requirement requires control geometrical variations in the combustion chamber between the upper planar surface of the block and the upper planar surface of the piston, surfaces 5,2 and 1,5 respectively in figure 5a. The surfaces belonging to the block, surfaces 5,1, 5,2 and 5,3, are assumed rigidly fixed. As simulation had to be performed for a given position of the mechanism, we have chosen the extreme configuration with the piston in top dead centre.

The fact of having a closed loop implies the computation of an intersection, as it can be noticed in Eq. 14. Assuming some homothetic transformations, the operations required for the simulation of FC2 are:

$$\Gamma_{5,2/1,2} = \Gamma_{5,2/5,0} \oplus (\Gamma_{5,0/1,0-a} \cap \Gamma_{5,0/1,0-b}) \tag{14}$$

where

$$\Gamma_{5,0/1,0-b} = \Gamma_{1,0/2,1} \oplus \Gamma_{2,1/3,0} \oplus \Gamma_{3,0/4,0} \oplus \Gamma_{4,0/5,0} \tag{15}$$

The computation of $\Gamma_{5,0/1,0-b}$ can be performed in a 4D space isolating the straight lines associated to the DOFs of the operands, which in turn are related with the twists:

$$\mathcal{T}_{5,0/1,0-b} = \begin{bmatrix} \mathcal{T}_{1,0/2,1} \\ \\ \mathcal{T}_{2,1/3,0} \\ \\ \mathcal{T}_{3,0/4,0} \\ \\ \mathcal{T}_{4,0/5,0} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{1z} & -d_{1y} \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & d_{2z} & -d_{2y} \end{bmatrix} \tag{16}$$

where $d_1 = [d_{1x}\ d_{1y}\ d_{1z}]$ and $d_2 = [d_{2x}\ d_{2y}\ d_{2z}]$ are respectively the position vectors of the local reference system of the surfaces 3,2 and 4,2 regarding the point chosen for the mobility analysis (centroid of surface 2,1).

The next operation, the intersection, demands the union of the wrenches associated to the operands for modelling the mobility conditions of the surface 5,0 with respect to 1,0:

$$\mathcal{W}_{5,0/1,0} = \begin{bmatrix} \mathcal{W}_{5,0/1,0-a} \\ \\ \\ \mathcal{W}_{5,0/1,0-b} \\ \\ \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{17}$$

As it can be noticed, $\mathcal{W}_{5,0/1,0}$ can be reduced to the identity matrix, which means that $\Gamma_{5,0/1,0}$ is a 6D operand and can not be decomposed. Therefore, the final twist matrix is composed only of the mobility introduced by $\Gamma_{5,2/5,0}$:

$$\mathcal{T}_{5,2/1,2} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

The summary of the simulation for FC2 is presented in table 6. With the proposed strategy the computational time for computing the sums was reduced by 76% with respect to the approach based on 6D operands. The results of FC1 could not be used directly for the simulation of FC2, because the first one is not exactly a subset of the second one. Comparing the time of both simulations, it can be noticed than that for FC1 is greater. This can be justified by the fact that, in this simulation, the operand of the planar surface 1,2 is summed directly with the ones derived from the cylindrical surfaces. For FC2, on the contrary, the first sum is made up of operands derived from cylindrical surfaces whose axes are parallel between them and belong to the same plane. This implies operands polytopes with similar topology, and then, their sum take much less time. Additionally, the intersection performed before the final sum, implies a significant reduction in complexity due to the restriction of all the DOFs in the operand $\Gamma_{5,0/1,0}$ (as shown in Eq. 17).

Table 6: Summary of the sums for the simulation of FC2

	Current method		Proposed method	
Polytope	Vertices	Time [s]	Vertices	Time [s]
$\Gamma_{1,0/2,1} \oplus \Gamma_{2,1/3,0}$	3304	5,6	512	0,7
$\Gamma_{1,0/3,0} \oplus \Gamma_{3,0/4,0}$	10864	69,3	1728	16,3
$\Gamma_{1,0/4,0} \oplus \Gamma_{4,0/5,0}$	34480	522,4	4096	160,4
$\Gamma_{4,0/5,0-a} \cap \Gamma_{4,0/5,0-b}$	4288	-	4288	-
$\Gamma_{4,0/5,0} \oplus \Gamma_{5,2/5,0}$	48440	1207	29544	251,3

5. Conclusions and future work

In this paper, a new method for calculating the cumulative stack up of deviations with sets of constraints was presented. The method is based on the decomposition of geometric and contact polyhedra into the sum of polytopes (the bounded part

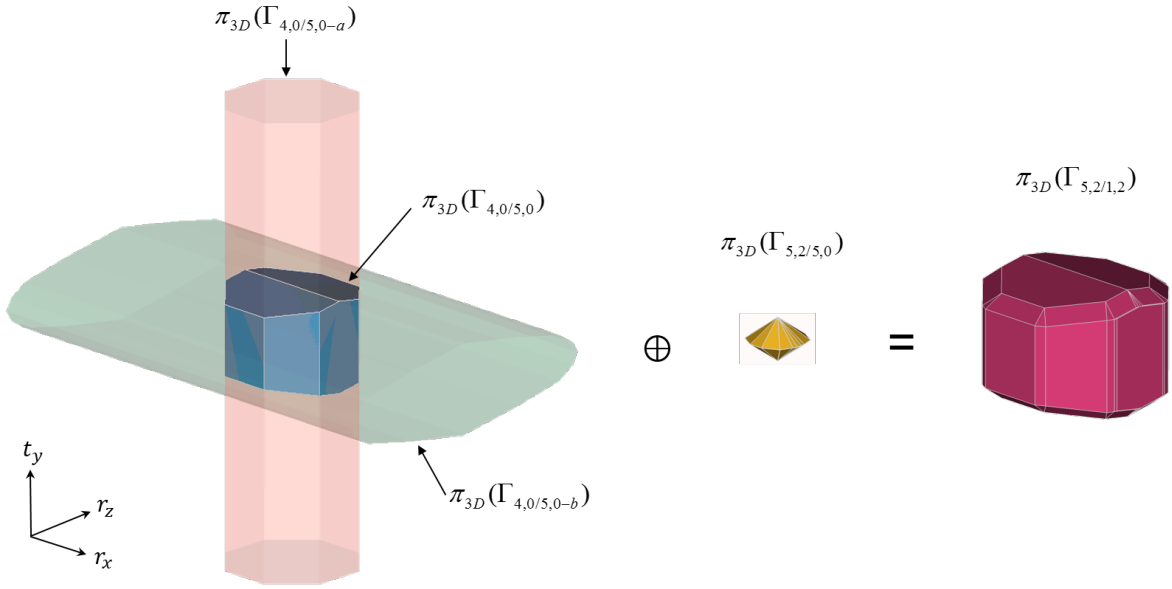


Figure 12: 3D representation of simulation of FC2 in 6D

of the polyhedron) and polyhedral cones (the unbounded part of the polyhedron). When performing a sum of polyhedra, it is possible to exploit this property and take just the bounded parts to calculate the sum. This new way of operating polyhedra was detailed in an algorithm.

The proposed decomposition of polyhedra is based on kinematic analysis of the toleranced features. Using the theory of screws it is possible to model the mobility conditions of a feature with respect to a reference or to another feature. In this way, the set of unbounded displacements can be identified, which in turn characterizes the polyhedral cone of the polyhedron. As the main interest of tolerance analysis lies in calculating the limits of the controllable displacement, which are characterized by the underlying polytopes of the operands, the polyhedral cones of the operands can be isolated from the simulation to reduce the complexity of the manipulated models.

The procedure and the results from applying the method were shown in a case study. A slider-crank mechanism was analyzed. The direct consequence of applying the new method was a 76% reduction in the computational time for the sums. In addition to the reduction in computational time, the method based on the decomposition of polyhedra reduces the complexity of the operands, and hence, the probabilities of having numerical problems during the calculations.

Due to the computational cost of summing of sets of constraints, the advantages of manipulating decomposed polyhedra were explored only for the case of serial assembly. Therefore, further work is required to formalize a strategy when the tolerance chain also contains parallel contacts.

When dealing with unilateral contacts, the polyhedra obtained from the sets of constraints are pointed (according to definition 2.3). As a consequence, the operand polyhedra cannot be completely decomposed as proposed in this paper. In these cases, the strategy based on cap half-spaces can be useful for simulating bounded sets in order to perform the computations.

In product design, the fact of recognizing the most influential features in a tolerance chain is essential. This information enables tolerances to be allocated in an optimal way in order to reduce production costs. In tolerance analysis, sets of constraints ensure that this can be achieved thanks to the traceability of the vertices and facets through the whole computational process. In this respect, the method presented in this paper is a first filter to identify among all the facets of the operands those that really influence the FC. However, further work is still required to develop a strategy with which to qualify the level of influence of the facets of the operands on the facets of the functional polytope (the polytope representing the FC).

References

[1] Alan Fleming. Geometric relationships between toleranced features. *Artificial Intelligence*, 37(13):403 – 412, 1988.

[2] M. Giordano, D. Duret, S. Tichadou, and R. Arrieux. Clearance space in volumic dimensioning. *CIRP Annals - Manufacturing Technology*, 41(1):565 – 568, 1992.

[3] J.K Davidson, A Mujezinovic, and JJ Shah. A new mathematical model for geometric tolerances as applied to round faces. *Journal of mechanical design*, 124(4):609–622, 2002.

[4] D. Teissandier, V. Delos, and Y. Coutard. Operations on polytopes: application to tolerance analysis. In *Global Consistency of Tolerances*, pages 425–433, Enschede (Netherlands), 1999. Kluwer academic.

[5] Pierre Bourdet, Luc Mathieu, Claire Lartigue, and Alexandre Ballu. The concept of the small displacement torsor in metrology. *Series on advances in Mathematics for Applied Sciences*, 40:110–122, 1996.

[6] Jami J. Shah, Gaurav Ameta, Zhengshu Shen, and Joseph Davidson. Navigating the tolerance analysis maze. *Computer-Aided Design and Applications*, 4(5):705–718, 2007.

[7] Vijay Srinivasan. The role of sweeps in tolerancing semantics. In *Inter. For. Dimensional Tolerancing and Metrology*, volume 27, pages 68–78. CRTD, 1993.

[8] Santiago Arroyave-Tobón, Denis Teissandier, and Vincent Delos. Tolerance analysis with polytopes in HV-description. In *Proceedings of ASME IDETC-CIE*, Charlotte, NC, USA, August 2016.

[9] A. Clément, A. Rivière, and P. Serré. *A Declarative Information Model for Functional Requirements*, pages 3–16. Springer Netherlands, Dordrecht, 1996.

[10] Max Giordano and Daniel Duret. Clearance space and deviation space. In *3rd. CIRP Seminars on Computer Aided Tolerancing, ENS Cachan, France, Apr*, pages 27–28, 1993.

[11] Thorsten Bernholt, Friedrich Eisenbrand, and Thomas Hofmeister. Constrained minkowski sums: A geometric framework for solving interval problems incomputational biology efficiently. *Discrete & Computational Geometry*, 42(1):22–36, 2009.

[12] Efi Fogel and Dan Halperin. Exact and efficient construction of minkowski sums of convex polyhedra with applications. *Computer-Aided Design*, 39(11):929 – 940, 2007.

[13] S.V. Rakovic, P. Grieder, M. Kvasnica, D.Q. Mayne, and M. Morari. Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances. In *43rd IEEE Conference on Decision and Control (CDC)*, volume 2, pages 1418–1423, Dec 2004.

[14] Yanyan Wu, Jami 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.

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

[16] Mathieu Mansuy, Max Giordano, and Pascal Hernandez. A new calculation method for the worst case tolerance analysis and synthesis in stack-type assemblies. *Comput. Aided Des.*, 43(9):1118–1125, September 2011.

[17] Lazhar Homri, Denis Teissandier, and Alex Ballu. Tolerance analysis by polytopes: Taking into account degrees of freedom with cap half-spaces. *Computer-Aided Design*, 62:112 – 130, 2015.

[18] Komei Fukuda. From the zonotope construction to the minkowski addition of convex polytopes. *Journal of Symbolic Computation*, 38(4):1261 – 1272, 2004.

[19] A. Frisoli, M. Solazzi, D. Pellegrinetti, and M. Bergamasco. A new screw theory method for the estimation of position accuracy in spatial parallel manipulators with revolute joint clearances. *Mechanism and Machine Theory*, 46(12):1929 – 1949, 2011.

[20] U. Kumaraswamy, M.S. Shunmugam, and S. Sujatha. A unified framework for tolerance analysis of planar and spatial mechanisms using screw theory. *Mechanism and Machine Theory*, 69:168 – 184, 2013.

[21] Alain Desrochers, Walid Ghie, and Luc Laperrière. Application of a unified jacobian-torsor model for tolerance analysis. *Journal of Computing and Information Science in Engineering*, 3(1):2–14, 2003.

[22] L. Laperrière and ElMaraghy H.A. Tolerance analysis and synthesis using jacobian transforms. *CIRP Annals - Manufacturing Technology*, 49(1):359 – 362, 2000.

[23] Jacques Marie Hervé. Intrinsic formulation of problems of geometry and kinematics of mechanisms. *Mechanism and Machine Theory*, 17(3):179 – 184, 1982.

[24] JM Hervé. The Lie group of rigid body displacements, a fundamental tool for mechanism design. *Mechanism and Machine Theory*, 34(5):719–730, July 1999.

[25] Pietro Fanghella. Kinematics of spatial linkages by group algebra: A structure-based approach. *Mechanism and Machine Theory*, 23(3):171–183, 1988.

[26] Jacques M. Hervé. The mathematical group structure of the set of displacements. *Mechanism and Machine Theory*, 29(1):73–81, January 1994.

[27] Qinchuan Li, Zhen Huang, and Jacques Marie Hervé. Displacement manifold method for type synthesis of lower-mobility parallel mechanisms. *Science in China Series E: Technological Sciences*, 47(6):641–650, 2004.

[28] R.S. Ball. *A Treatise on the Theory of Screws*. Cambridge University Press, Cambridge, UK, 1998.

[29] J.D. Adams and D.E. Whitney. Application of screw theory to constraint analysis of assemblies of rigid parts. In *Assembly and Task Planning, 1999. (ISATP '99) Proceedings of the 1999 IEEE International Symposium on*, pages 69–74, 1999.

[30] S. Gerbino and F. Arrichiello. How to investigate constraints and motions in assemblies by screw theory. In *CIRP ICME'04*, volume 4, pages 331–336, 2004.

[31] J.M. Rico, J. Gallardo, and J. Duffy. Screw theory and higher order kinematic analysis of open serial and closed chains. *Mechanism and Machine Theory*, 34(4):559 – 586, 1999.

[32] José M. Rico Martínez and Joseph Duffy. Classification of screw systemsi. one- and two-systems. *Mechanism and Machine Theory*, 27(4):459 – 470, 1992.

[33] José M. Rico Martínez and Joseph Duffy. Classification of screw systemsii. three-systems. *Mechanism and Machine Theory*, 27(4):471 – 490, 1992.

[34] Jian S. Dai and John Rees Jones. Interrelationship between screw systems and corresponding reciprocal systems and applications. *Mechanism and Machine Theory*, 36(5):633 – 651, 2001.

[35] Günter M. Ziegler. *Lectures on polytopes*, volume 152. Springer Science & Business Media, 1995.

[36] Vincent Delos, Denis Teissandier, and Santiago Arroyave-Tobón. How to trace the significant information in tolerance analysis with polytopes. In *Lecture Notes in Mechanical Engineering*, volume 2, pages 1003–1012, Catania, Italy, September 2016. Springer International Publishing.

[37] H.J. Su and C. Yue. Type synthesis of freedom and constraint elements for design of flexure mechanisms. *Mechanical Sciences*, 4(2):263–277, 2013.

[38] Jeffrey D Adams. Feature based analysis of selective limited motion in assemblies. Master's thesis, Massachusetts Institute of Technology, 1998.

[39] R. Konkar and M. Cutkosky. Incremental Kinematic Analysis of Mechanisms. *Journal of Mechanical Design*, 117(4):589–596, December 1995.

[40] Chintien Huang, Koichi Sugimoto, and Ian Parkin. The correspondence between finite screw systems and projective spaces. *Mechanism and Machine Theory*, 43(1):50 – 56, 2008.

[41] ISO 3952-1. Kinematic diagrams - graphical symbols, part 1, 1981.

[42] ISO 5459. Geometrical tolerancing datums and datum systems, 2011.

[43] A. Desrochers and A. Clément. A dimensioning and tolerancing assistance model for cad/cam systems. *The International Journal of Advanced Manufacturing Technology*, 9(6):352–361, 1994.

[44] Vincent Delos and Denis Teissandier. Minkowski sum of polytopes defined by their vertices. *Journal of Applied Mathematics and Physics (JAMP)*, 3(1):62–67, January 2015.