POLITOPIX, an Open-source application for managing constraints by polytopes

Vincent Delos
Denis Teissandier

GRT – Genève, le 4 Juin 2014

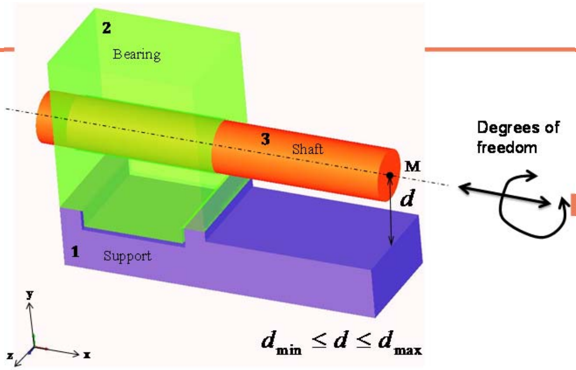
Summary

- Tolerancing analysis software developped at I2M
- Politopix
- Minkowski sum example
- Example of a complete tolerancing process
- Conclusion and perspectives

Tolerancing analysis software developped at I2M

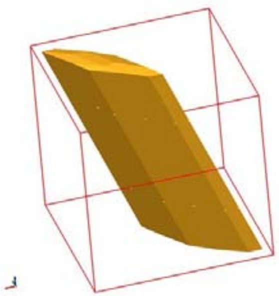
➔ Software structure

OpenCascade



Pre-processing

Calculated polytope



Post-processing

Solver Politopix

Computation of

- Operand polytopes
- Minkowski sums
- Intersections

What are polytopes ?

Minkowski-Weyl theorem states that both definitions are equivalent

Definition of the $\mathcal{V}$-description

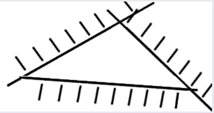
A polytope is the linear convex combination of a finite set of points

$$P = \left\{ x \in \mathbb{R}^n, x = \sum_{i=1}^{k} \alpha_i a_i \right\}$$

$$a_i \in \mathbb{R}^n, \alpha_i \in \mathbb{R}^+, \sum_{i=1}^{k} \alpha_i = 1$$

Definition of the $\mathcal{H}$-description

A polytope is the bounded intersection of a finite set of half-spaces



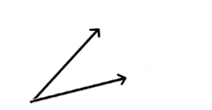
$$P = \bigcap_{u=1}^{l} \bar{H}_u^+$$

What are polyhedral cones ?

Minkowski-Weyl theorem states that both definitions are equivalent

Definition of the $\mathcal{V}$-description

A polyhedral cone is the positive linear combination of a finite set of vectors

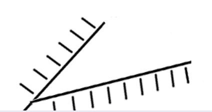


$$P = \left\{ x \in \mathbb{R}^n, x = \sum_{i=1}^{k} \alpha_i a_i \right\}$$

$a_i \in \mathbb{R}^n, \alpha_i \in \mathbb{R}^+$

Definition of the $\mathcal{H}$-description

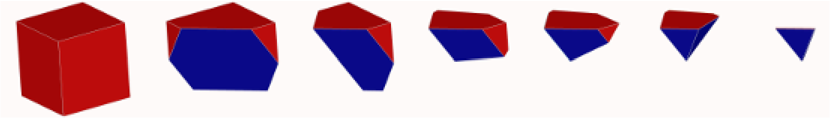
A polyhedral cone is the intersection of a finite set of half-spaces whose frontier contains the origin



$$P = \bigcap_{u=1}^{l} \bar{H}_u^+$$

How to compute polytopes ?

We use the Double Description algorithm to compute the $\mathcal{V}$-description from the $\mathcal{H}$-description in $\mathbb{R}^n$



- We initialize the algorithm with the 1-skeleton of a bounding cube
- We successively chop with the half-spaces of the $\mathcal{H}$-description
- At the last iteration we get the $\mathcal{V}$-description

Note : we can also compute with the same algorithm intersections between polytopes and intersections between polyhedral cones

Computing a polytope $\mathcal{HV}$-description

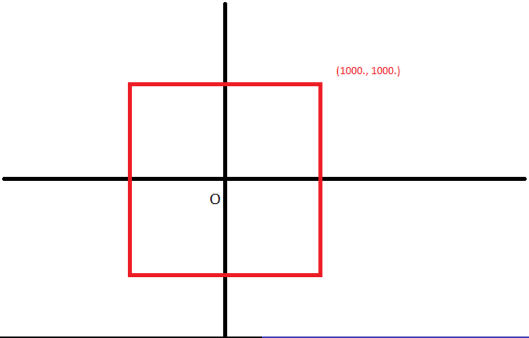
$\mathcal{H}$-description file constraints1.ptop

#*Dimension NumberOfHalfspaces NumberOfVertices*
3 8 0
#*HALFSPACES* : $a0 + a1.x1 + ... + an.xn \geq 0.$
21.6366 -0.432731 0.540914 -0.721218
21.6366 0.432731 -0.540914 0.721218
21.6366 0.432731 0.540914 -0.721218
21.6366 -0.432731 -0.540914 0.721218
21.6366 0.432731 -0.540914 -0.721218
21.6366 -0.432731 0.540914 0.721218
21.6366 -0.432731 -0.540914 -0.721218
21.6366 0.432731 0.540914 0.721218

Command line running politopix with the file constraints1.ptop

./politopix -p1 constraints1.ptop -d 3 -bb 1000.

- option -p1 : provide the polytope $\mathcal{H}$-description file
- option -d : the dimension of the space we work in
- option -bb : the bounding box size centered on the origin to initialize the algorithm



Computing a polytope $\mathcal{HV}$-description

$\mathcal{HV}$-description file outputconstraints1.ptop

#*Dimension NumberOfHalfspaces NumberOfVertices*
3 8 6
#*HALFSPACES* : $a0 + a1.x1 + \ldots + an.xn \geq 0.$
21.6366 -0.432731 0.540914 -0.721218
21.6366 0.432731 -0.540914 0.721218
21.6366 0.432731 0.540914 -0.721218
21.6366 -0.432731 -0.540914 0.721218
21.6366 0.432731 -0.540914 -0.721218
21.6366 -0.432731 0.540914 0.721218
21.6366 -0.432731 -0.540914 -0.721218
21.6366 0.432731 0.540914 0.721218
#*GENERATORS* : $V = (v1, \ldots, vn)$
0. 0. 30.00008319
0. 40.00007395 0.
50.00011555 0. 0.
-50.00011555 0 0
0. -40.00007395 0.
0. 0. -30.00008319
#*GENERATOR CONNECTION* : $Ha, Hb, \ldots$
0 2 4 6
1 3 4 6
0 3 5 6
1 2 4 7
0 2 5 7
1 3 5 7

Computing $\mathcal{HV}$-polytopes intersections

Intersections between polytopes in $\mathbb{R}^6$ at a given tolerance

./politopix -p1 polytope1.ptop -p2 polytope2.ptop -d 6 -t 0.0000001

- option -p1 : provide the first polytope $\mathcal{HV}$-description file
- option -p2 : provide the second polytope $\mathcal{HV}$-description file
- option -d : the dimension of the space we work in
- option -t : the minimum distance to differentiate 2 points

Intersections between polytopes with the output file and checks

./politopix -p1 polytope1.ptop -p2 polytope2.ptop -d 6 -o inter.ptop -ch

- option -o : for a specific output file instead of *outputpolytope1.ptop*
- option -ch : checks for all vertices inclusion in all the half-spaces, at least n facets per vertex in $\mathbb{R}^n$, at least n vertices per facet in $\mathbb{R}^n$

Other computations based on the double description

Checking equality between polytopes

```
./politopix -p1 polytope1.ptop -p2 polytope2.ptop -d 6 -EQ
```

- option -p1 : provide the first polytope $\mathcal{HV}$-description file
- option -p2 : provide the second polytope $\mathcal{HV}$-description file
- option -d : the dimension of the space we work in
- option -EQ : checks for all vertices of the first polytope are in the second polytope half-spaces and vice-versa

Intersections between polyhedral cones with the output file

```
./politopix -c1 cone1.pcon -c2 cone2.pcon -d 6 -o inter.pcon
```

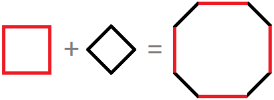
- option -c1 : provide the first polyhedral cone $\mathcal{HV}$-description file
- option -c2 : provide the second polyhedral cone $\mathcal{HV}$-description file
- option -d : the dimension of the space we work in
- option -o : the output file

All supported options

Option -h

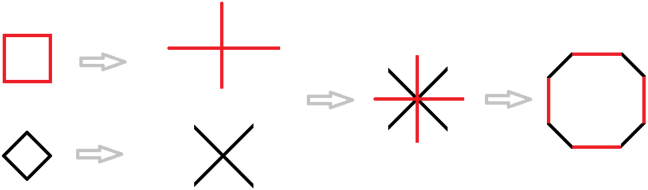
./politopix -h
Version 1.0.0
-d [–dimension] ARG : Set the cartesian space dimension
-t [–tolerance] ARG : Set the cartesian space tolerance [default: 1.e-06]
-o [–output] ARG : The optional output file (ptop or pcon extension)
-p1 [–polytope1] ARG : First polytope input file (ptop extension)
-p2 [–polytope2] ARG : Second polytope input file (ptop extension)
-pv [–polytopevolume] ARG : First polytope input file (ptop extension)
-c1 [–polyhedralcone1] ARG : First polyhedral cone input file (pcon extension)
-c2 [–polyhedralcone2] ARG : Second polyhedral cone input file (pcon extension)
-cf [–check-facets] ARG : Used to test when we know the final number of facets
-cg [–check-generators] ARG : Used to test when we know the final number of generators
-bs [–boundingsimplex] ARG : Bounding simplex size, containing the bounding box -bb (n+1 vertices)
-bb [–boundingbox] ARG : Bounding box size centered on the origin including the polytope ($2^n vertices$)
-ch [–check-all] : Used to perform all tests (no arguments, can be slow)
-MS [–MinkowskiSum] : Set the option to turn on Minkowski sums
-IN [–Intersection] : Set the option to turn on intersections (default option)
-EQ [–Equality] : Set the option to turn on the equality check between ptop or pcon
-v [–version] : Give the current version

Computing the Minkowski sum of two polytopes



The algorithm is based on the normal fans refinement

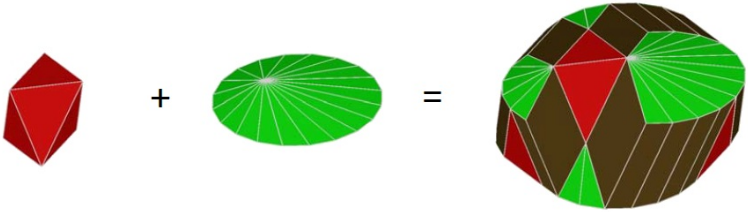
- It intersects dual cones from A and B
- It needs both polytopes $\mathcal{HV}$-description



Computing the Minkowski sum of two polytopes

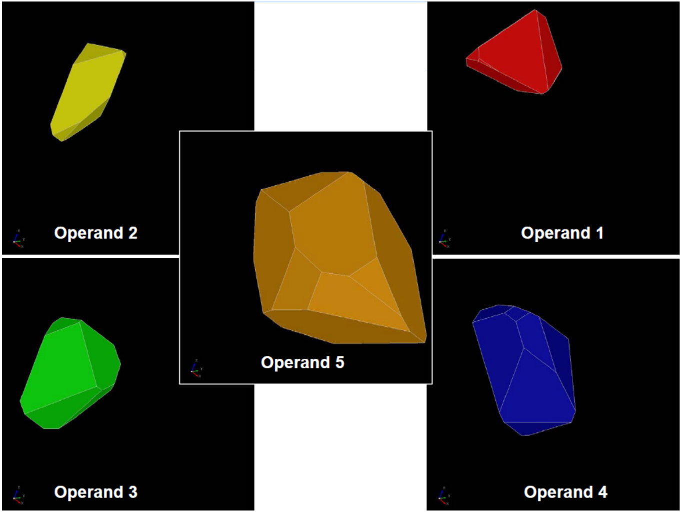
politopix summing polytopes

./politopix -p1 polytope1.ptop -p2 polytope2.ptop -d 3 -MS

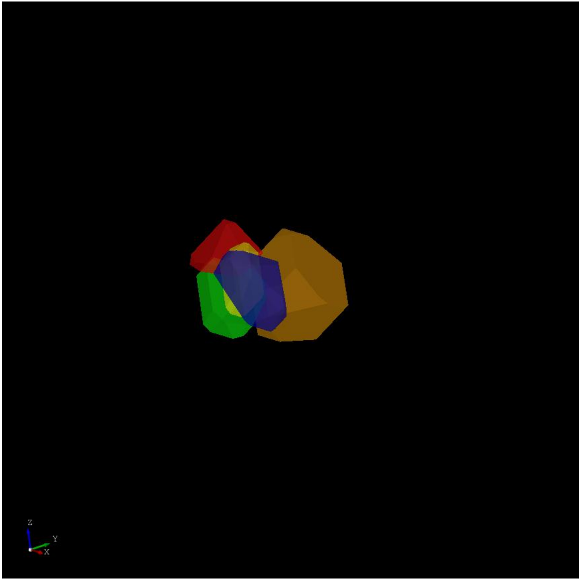


In $\mathbb{R}^3$ new half-spaces appear in the sum that are neither from the first polytope, nor from the second

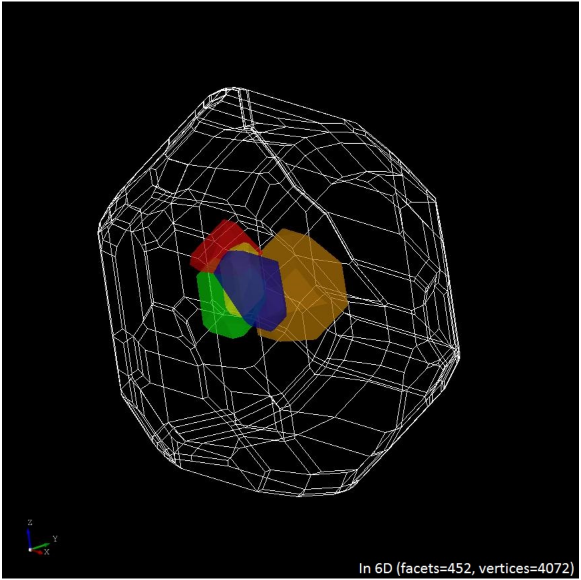
Minkowski sum of 5 polytopes in $\mathbb{R}^3$ (time $<$ 45 s)



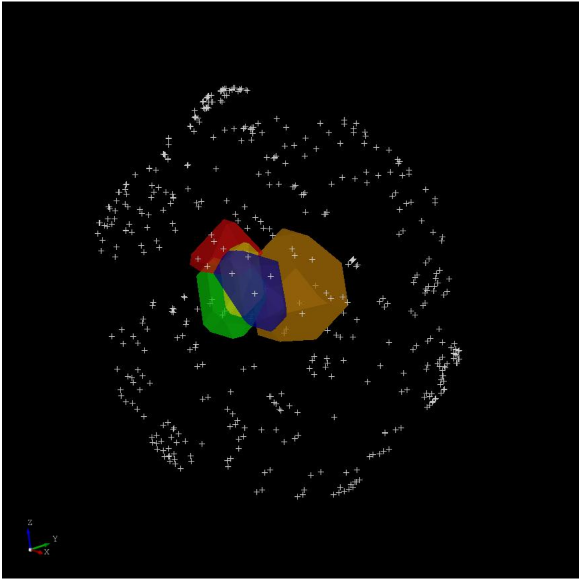
Minkowski sum of 5 polytopes in $\mathbb{R}^3$ (time $<$ 45 s)



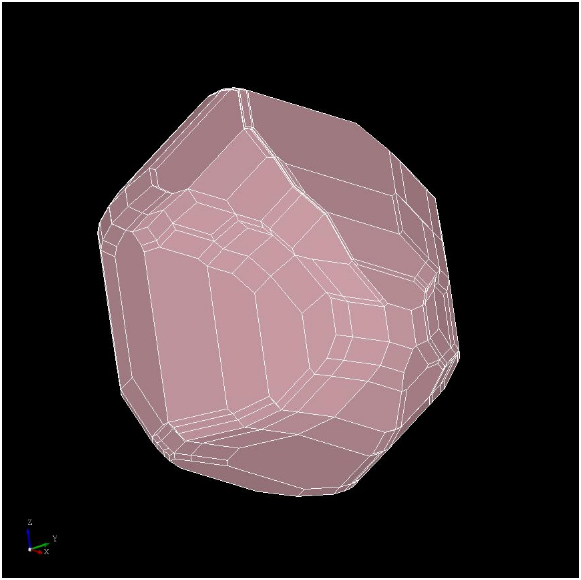
Minkowski sum of 5 polytopes in $\mathbb{R}^3$ (time $<$ 45 s)



Minkowski sum of 5 polytopes in $\mathbb{R}^3$ (time $<$ 45 s)



Minkowski sum of 5 polytopes in $\mathbb{R}^3$ (time < 45 s)



Validating politopix

- The software has been validated with a library of more than 250 tests, most of them coming from industrial cases.

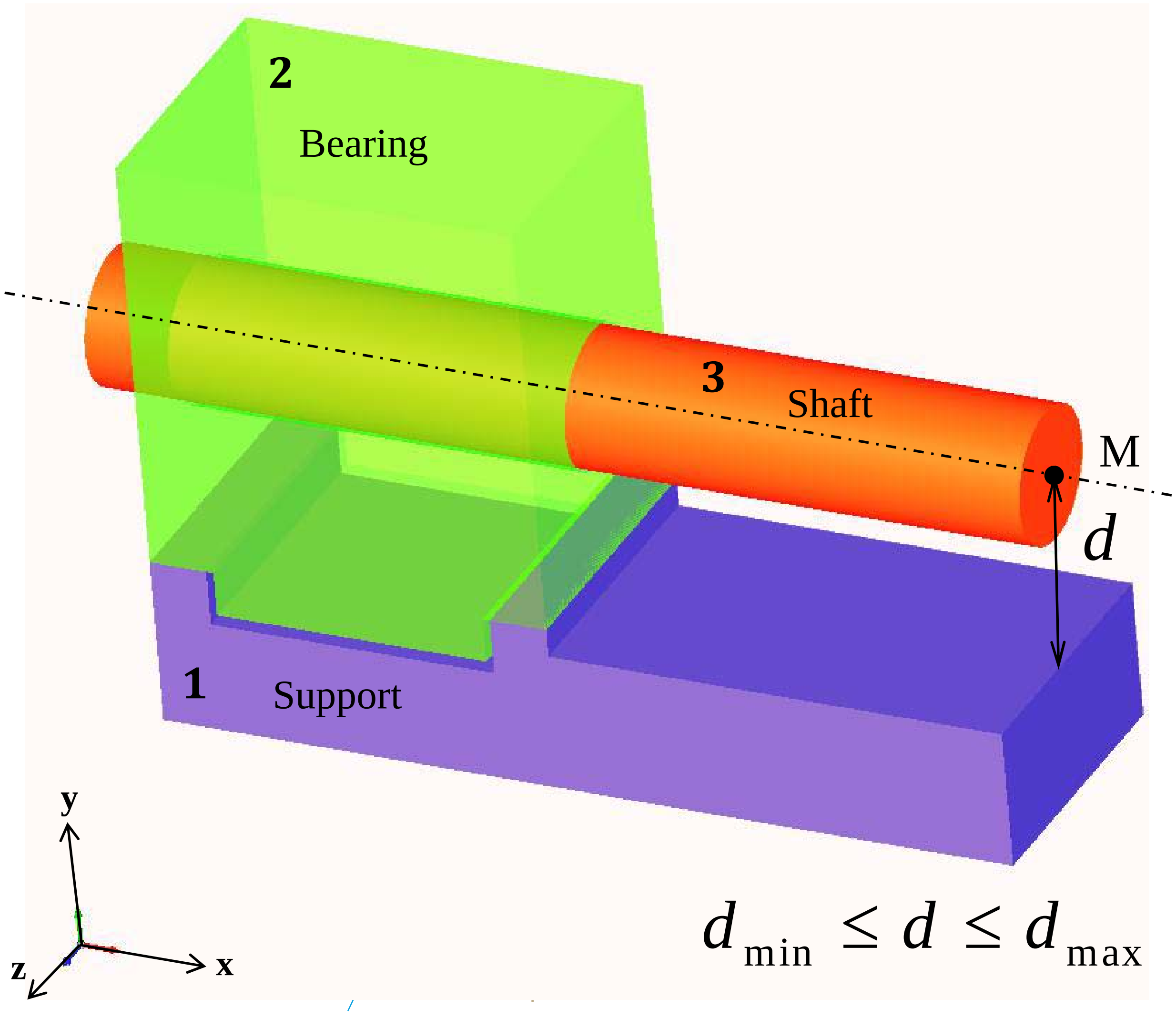
Examples of $\mathbb{R}^6$-sums with the numbers of half-spaces and vertices

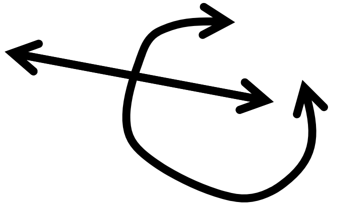
$P_1(26, 240) + P_2(26, 240) = P_3(592, 3168)$ in 6 s
$P_1(592, 3168) + P_2(26, 288) = P_3(4428, 18128)$ in 138 s
$P_1(4428, 18128) + P_2(26, 336) = P_3(19296, 65924)$ in 20 min 43 s

- The open source software qhull is used to check that the computed $\mathcal{V}$-description of polytopes and Minkowski sums are correct.
- politopix is under the GNU General Public Licence and can be downloaded at http://delosvin.perso.math.cnrs.fr, click on “politopix” and then “Telecharger”.

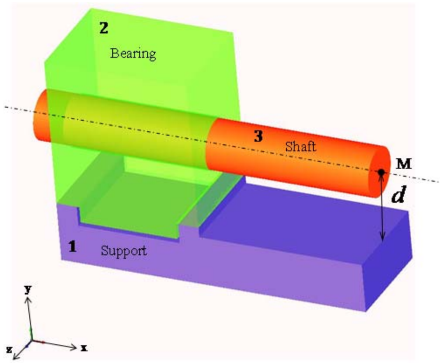
Application example of a complete tolerancing process

➔ Presentation of the mechanical system

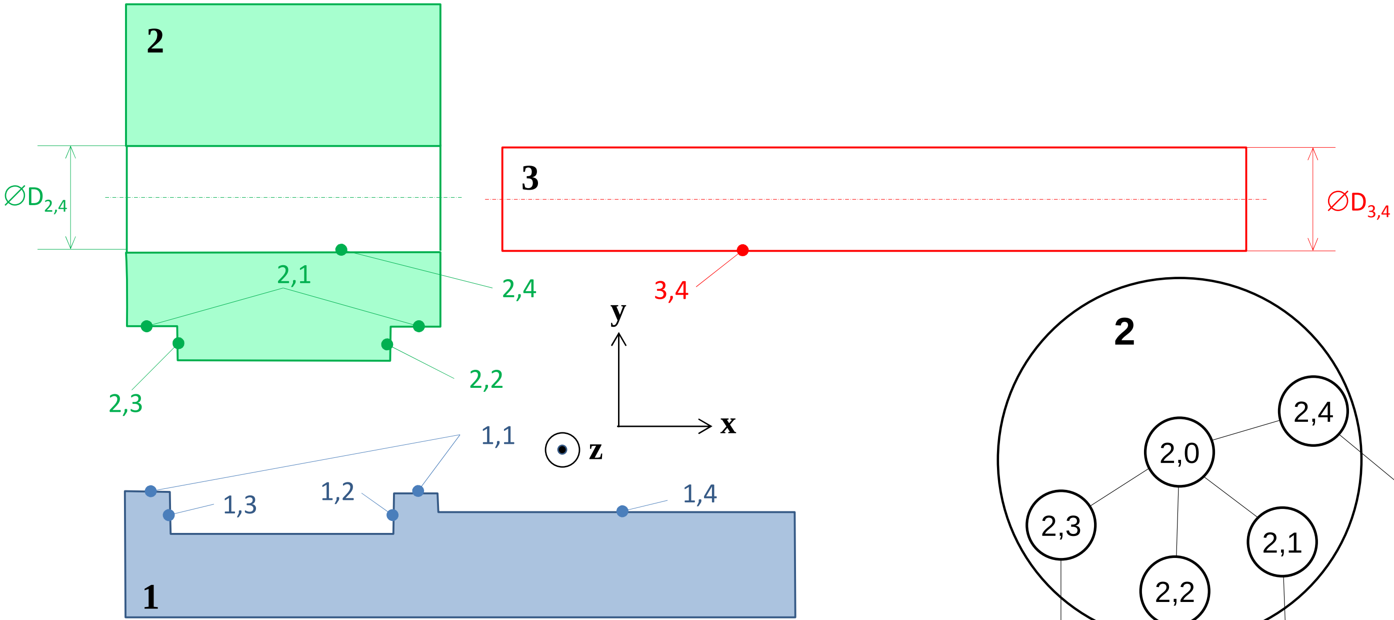


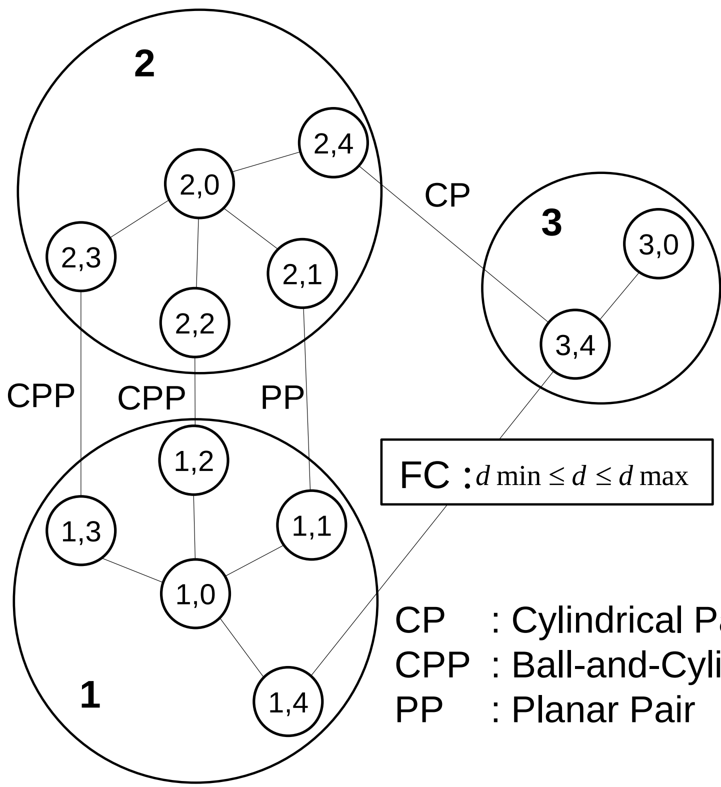


Issue of tolerancing analysis with polytopes



→ Operations on sets of constraints





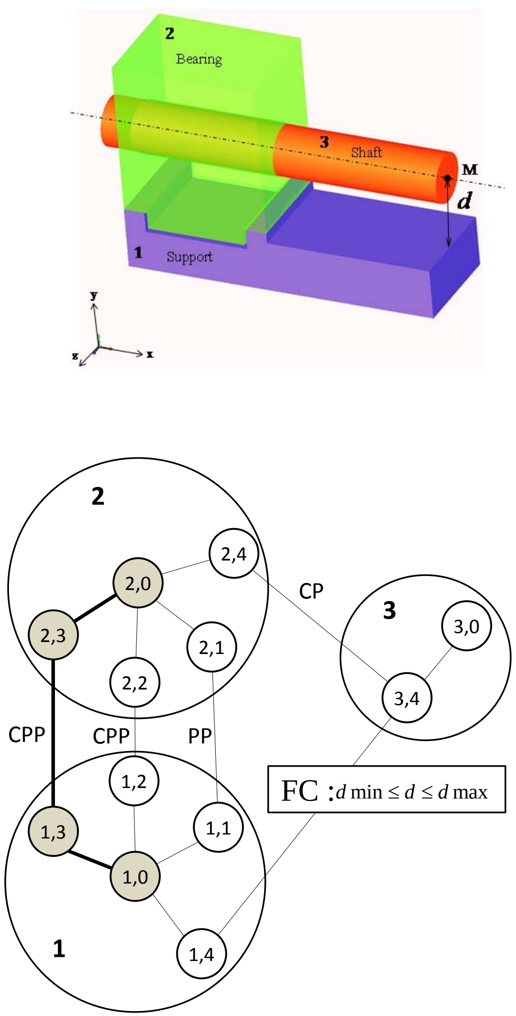
$$\mathcal{P}_{3,4/1,4} = \mathcal{P}_{3,4/2,4} \oplus \mathcal{P}_{2,4/2,0} \oplus \begin{bmatrix} \left(\mathcal{P}_{2,0/2,3} \oplus \mathcal{P}_{2,3/1,3} \oplus \mathcal{P}_{1,3/1,0}\right) \\ \cap\left(\mathcal{P}_{2,0/2,2} \oplus \mathcal{P}_{2,2/1,2} \oplus \mathcal{P}_{1,2/1,0}\right) \\ \cap\left(\mathcal{P}_{2,0/2,1} \oplus \mathcal{P}_{2,1/1,1} \oplus \mathcal{P}_{1,1/1,0}\right) \end{bmatrix} \oplus \mathcal{P}_{1,0/1,4}$$

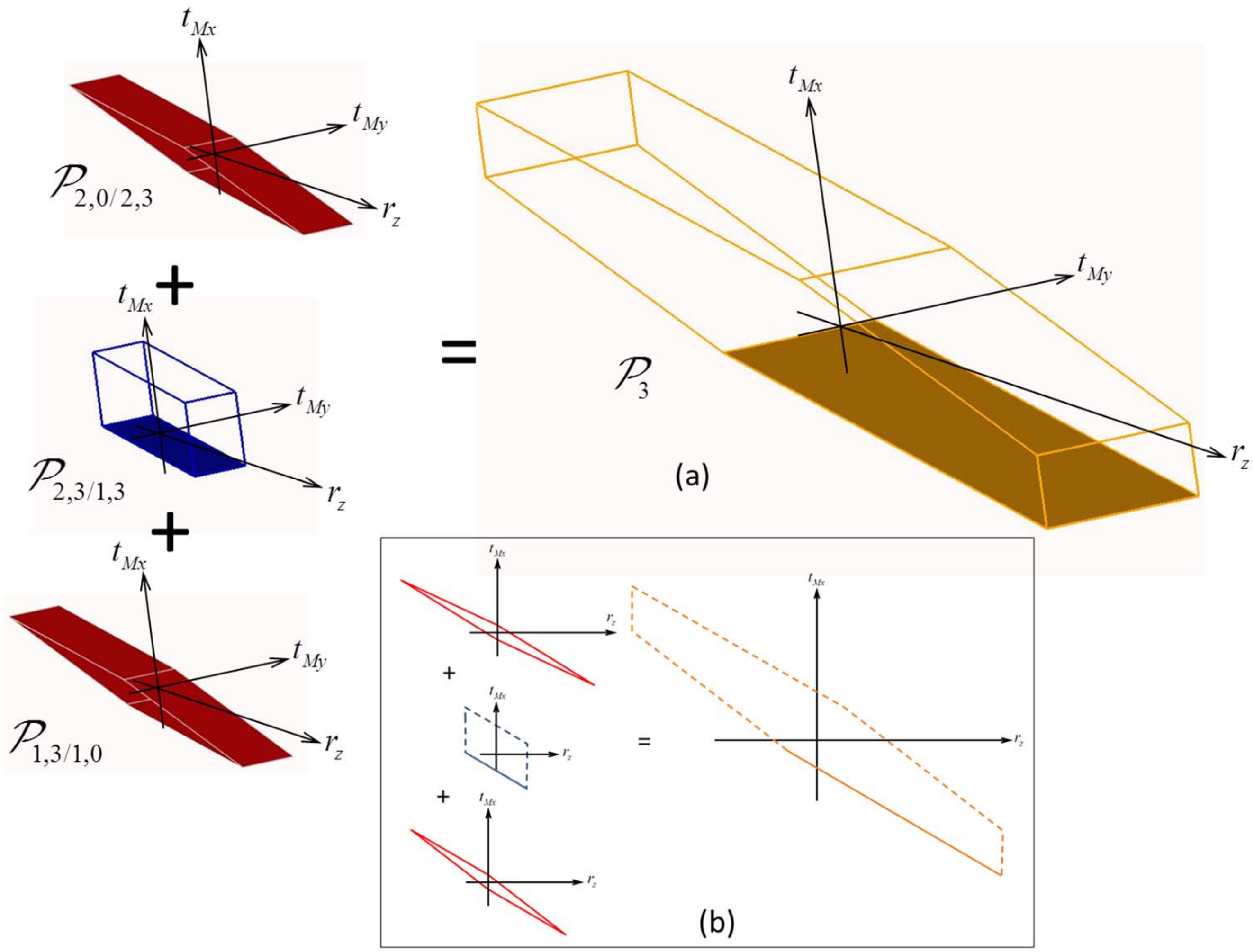
FC is satisfied ➡ $\mathcal{P}_{3,4/1,4} \subseteq P^{f}_{3,4/1,4}$

CP : Cylindrical Pair
CPP : Ball-and-Cylinder Pair
PP : Planar Pair

Management of cap-half spaces in operations on polytopes

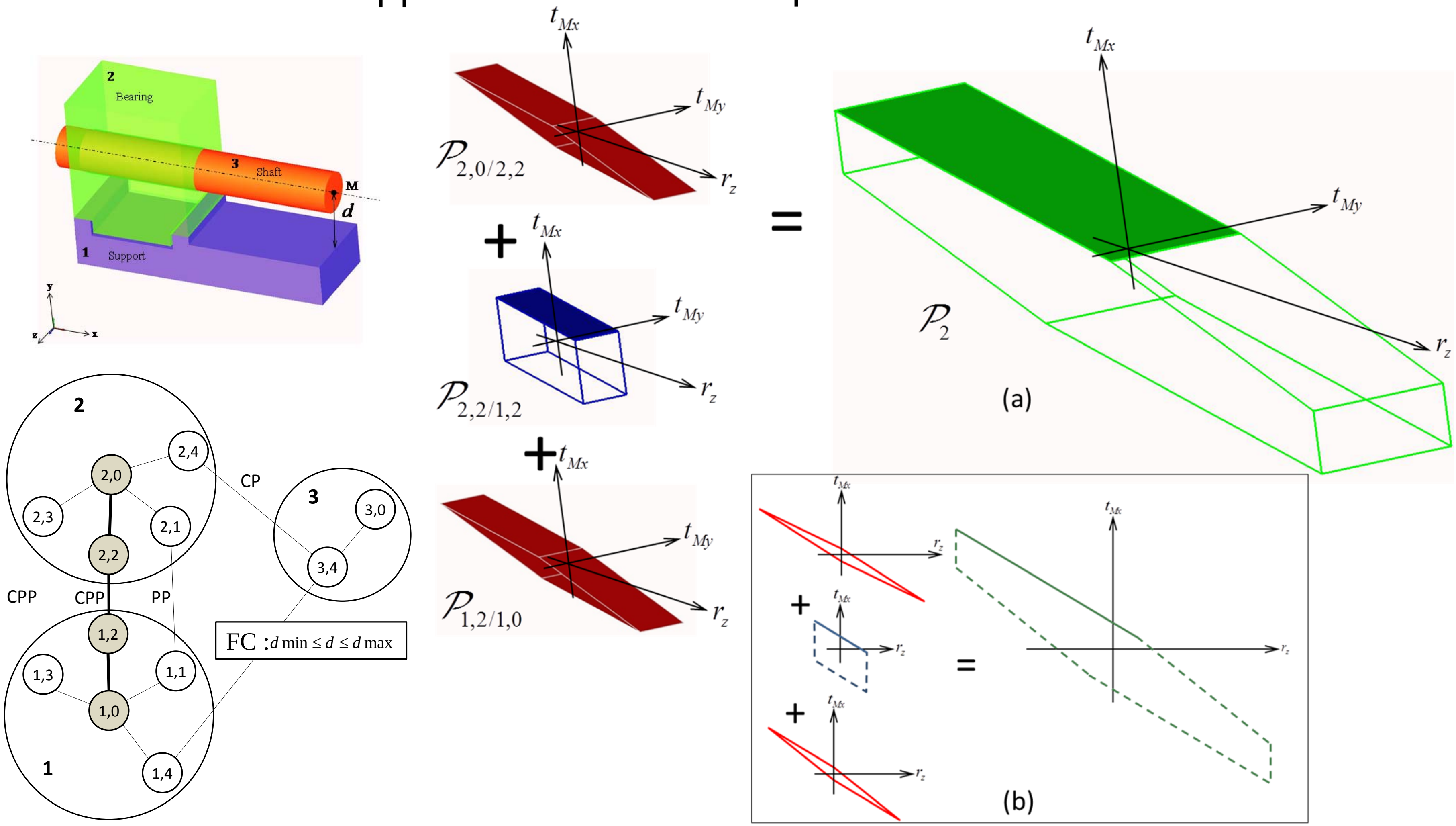
➔ Minkowski sum: application to the example





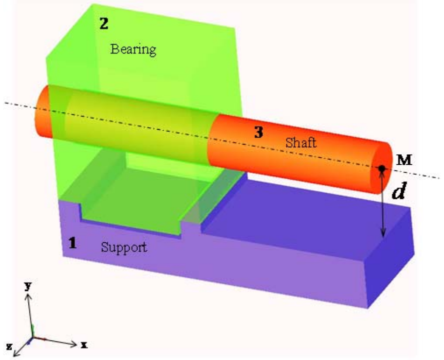
Management of cap-half spaces in operations on polytopes

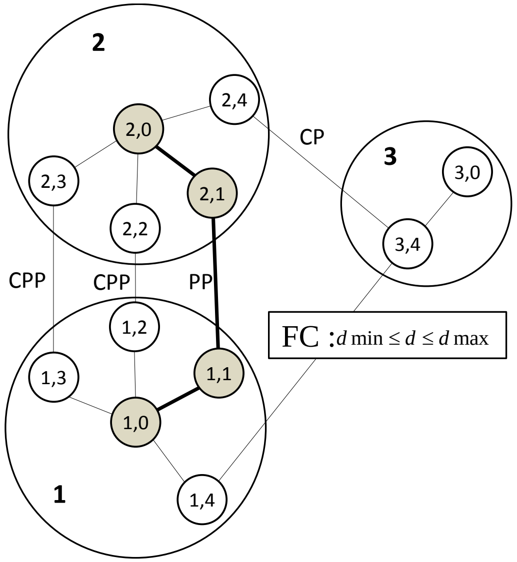
→ Minkowski sum: application to the example

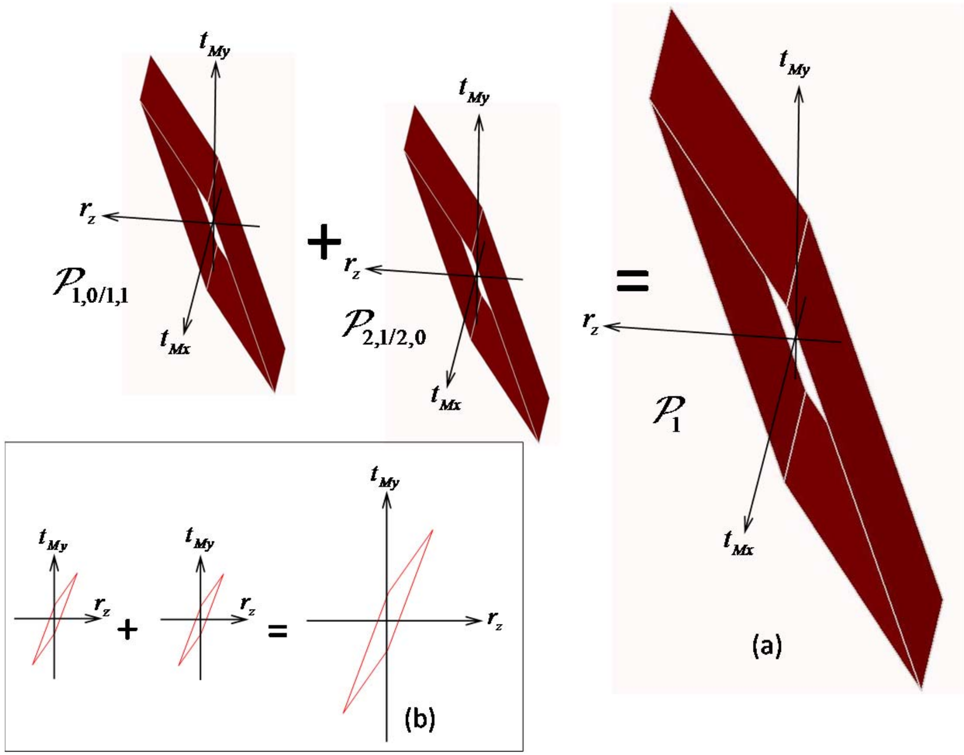


Management of cap-half spaces in operations on polytopes

➔ Minkowski sum: application to the example

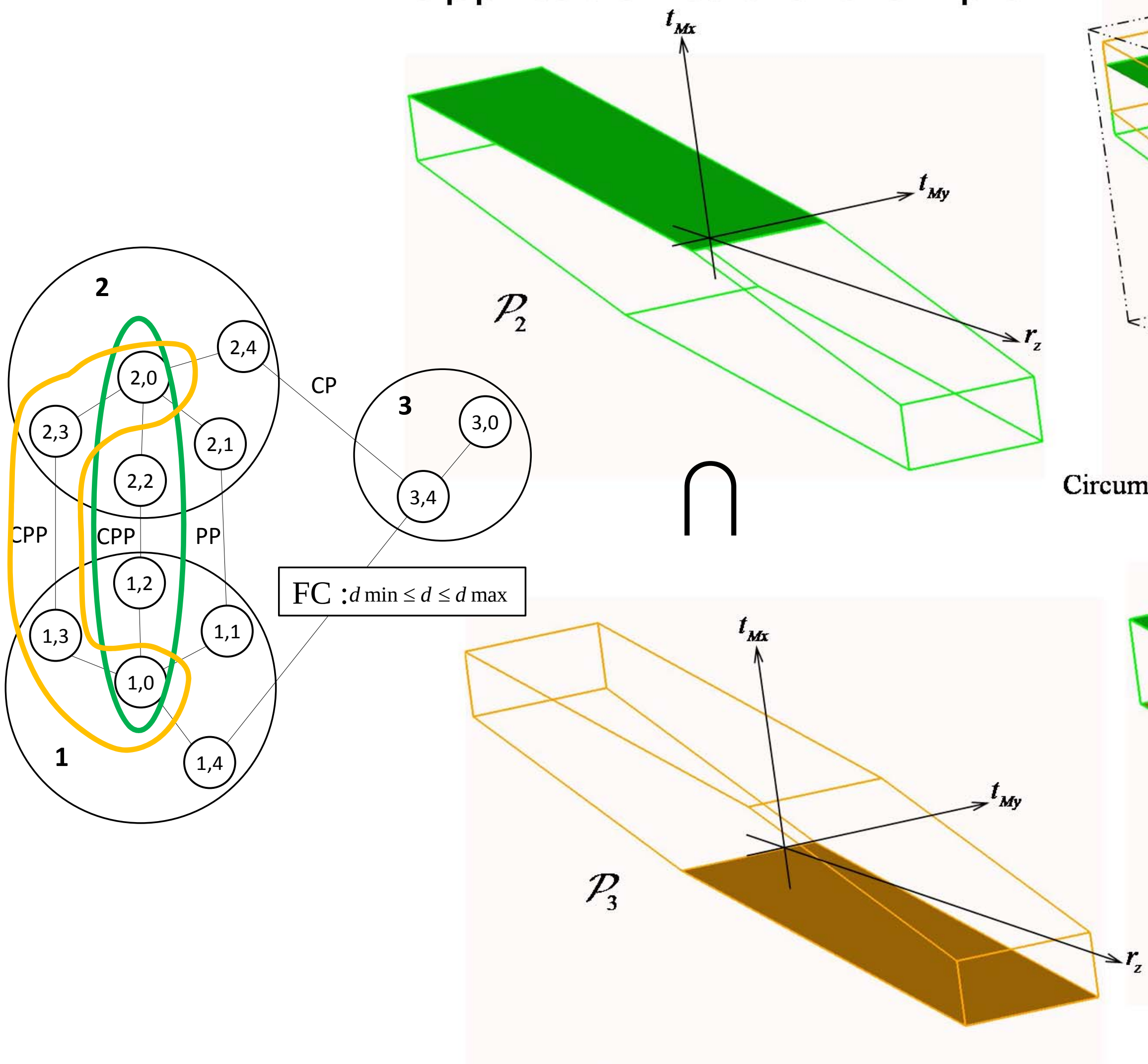


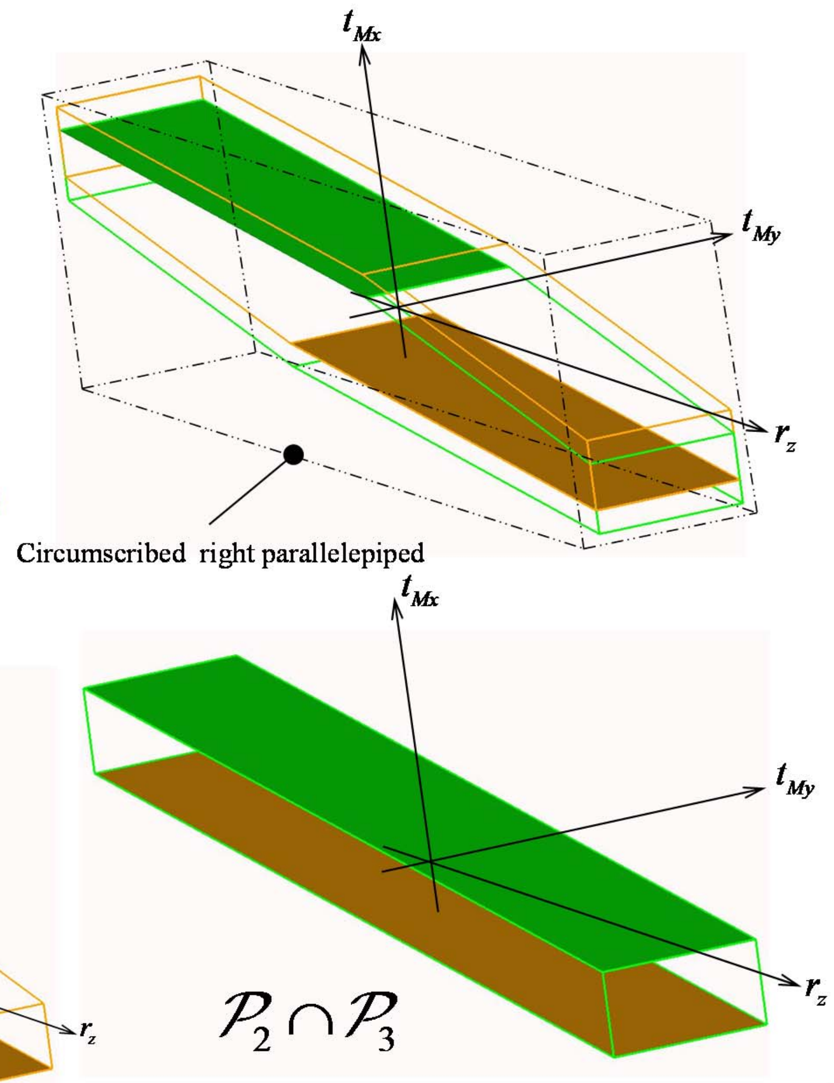




Management of cap-half spaces in operations on polytopes

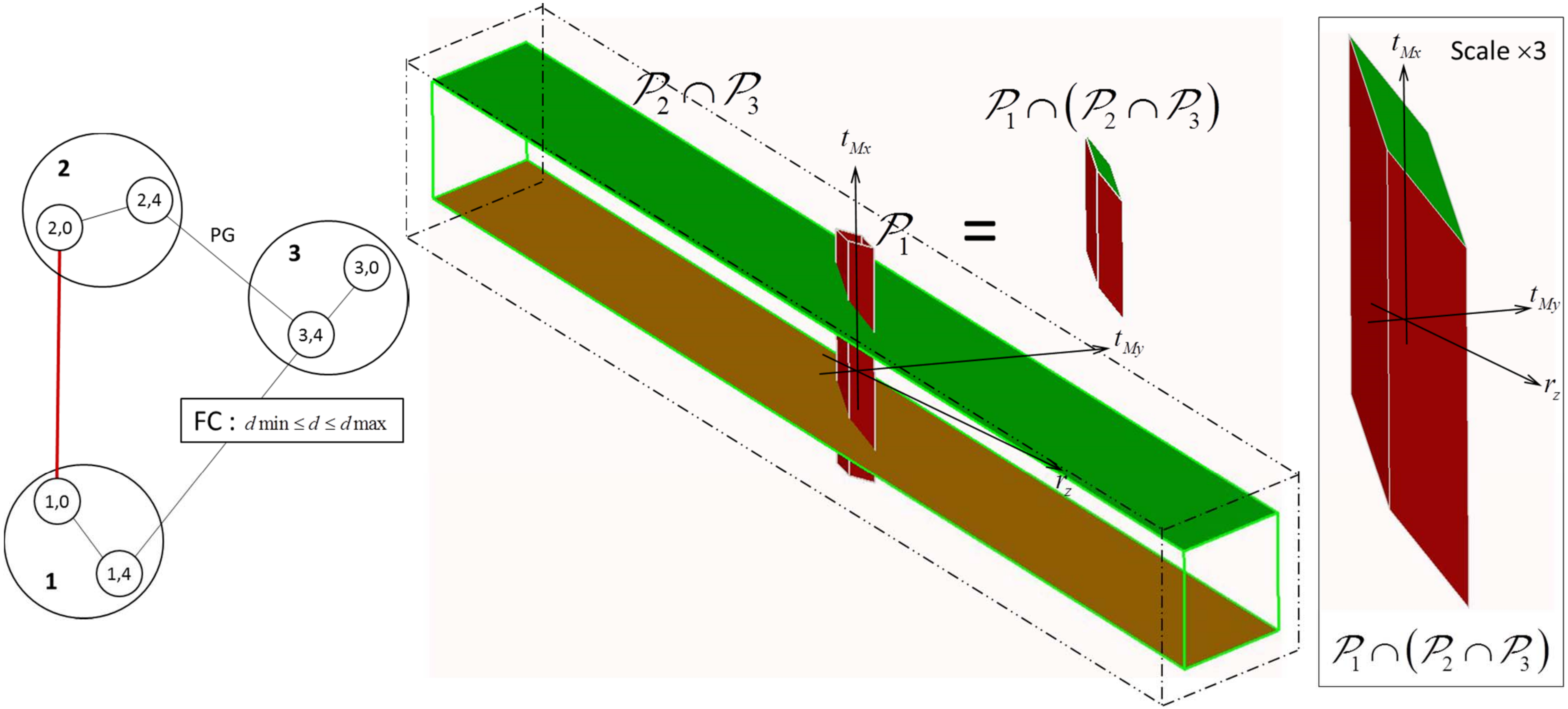
- ➔ Intersection: application to the example





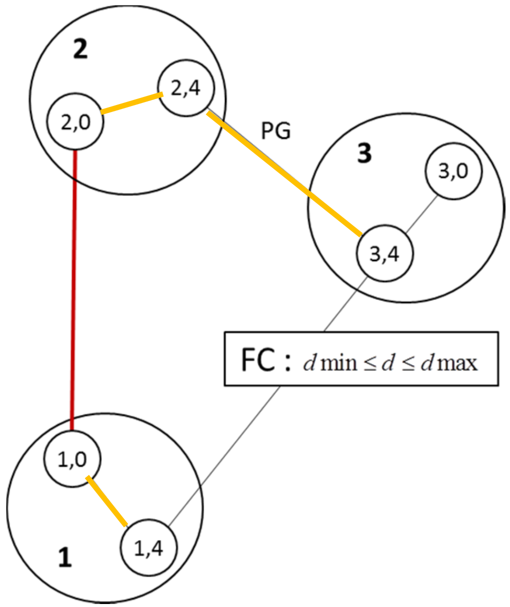
Management of cap-half spaces in operations on polytopes

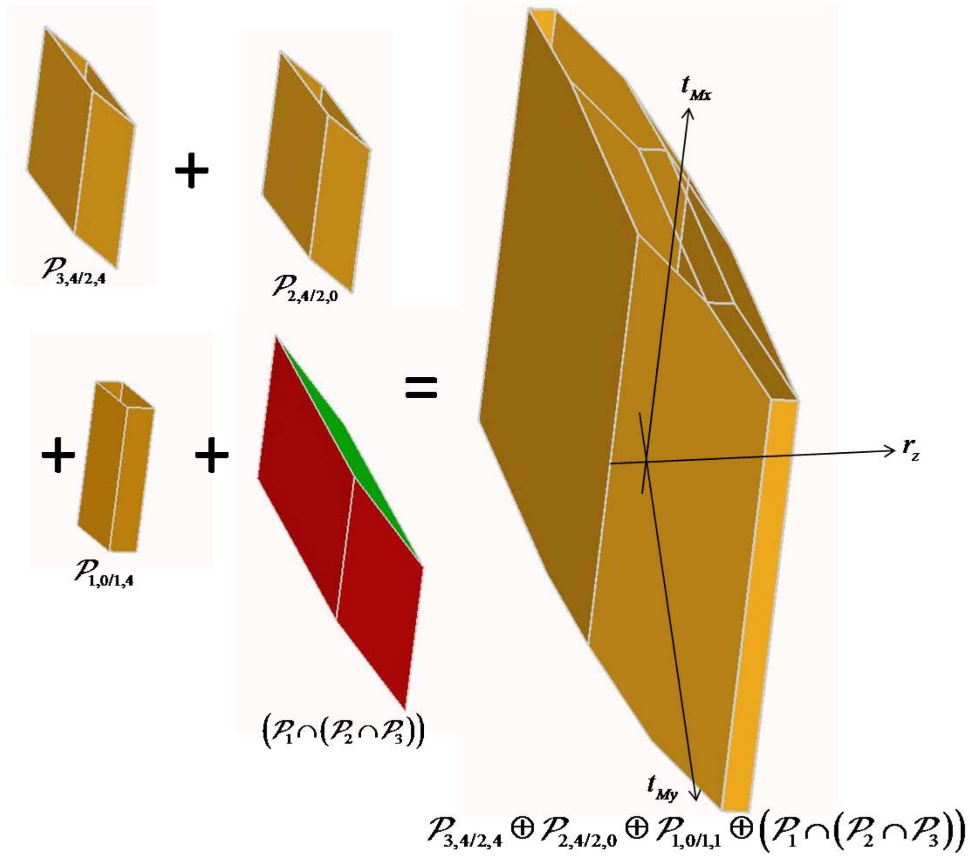
→ Intersection: application to the example



Management of cap-half spaces in operations on polytopes

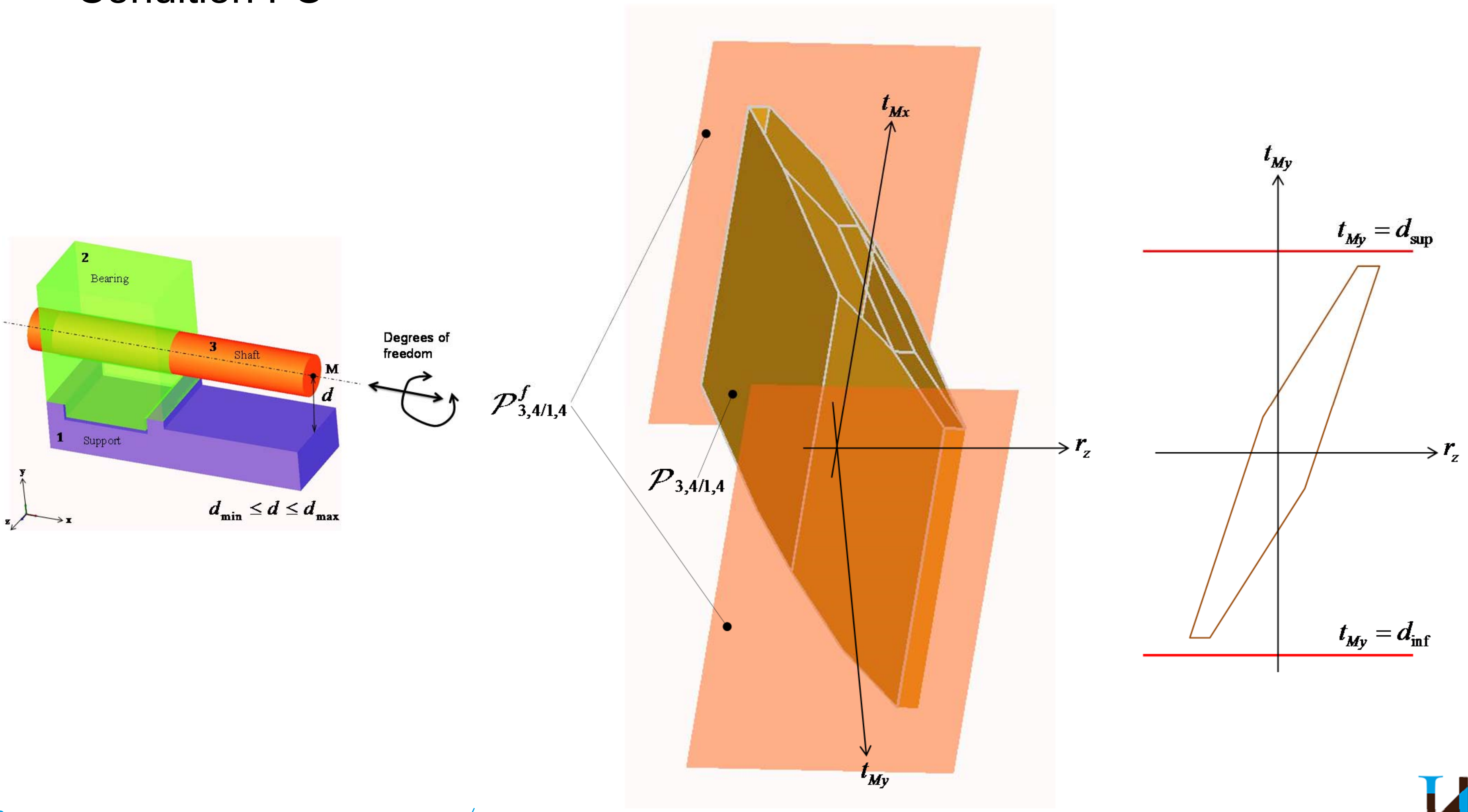
➔ Final result





Management of cap half-spaces in operations on polytopes

➔ Simulation of mechanical system compliance with the Functional Condition FC



Conclusion and perspectives

➔ Introduction of cap half-spaces to take into account the unbounded displacements

- the cap half-spaces bound the geometric constraints sets
- the cap half-spaces bound the contact constraints sets

➔ Management of cap half-spaces of derived polytopes from

- Minkowski sum
- intersection

➔ Developments of some strategies in order to improve

- the numerical precision
- the robustness of polytopes
- the computing time

Conclusion and perspectives

➔ Some references

- Teissandier D., Couétard Y., Gérard A., *A Computer Aided Tolerancing Model : Proportioned Assemblies Clearance Volume*, Computer-Aided Design, *Vol. 31: 805-817*, 1999.
- Teissandier, D., Delos, V. & Couétard, Y., 1999. *Operations on polytopes: application to tolerance analysis*. In *Global Consistency of Tolerances*. 6th CIRP Seminar on Computer Aided Tolerancing. Enschede (Netherlands): ISBN 0-7923-5654-3, Kluwer academic publisher, p. 425-433.
- Teissandier D., Delos V., *Algorithm for the calculation of the Minkowski sums of 3-polytopes based on normal fans*, Computer-Aided Design, *Vol. 43:1567-1576*, 2011.
- Teissandier D., Delos V., Algorithm to calculate the Minkowski sums of 3-polytopes dedicated to tolerance analysis, IMProve 2011, Venice (Italy), June 15-17, 2011
- Pierre L., Teissandier D., Nadeau J.P., *Integration of thermomechanical strains into tolerancing analysis*, Int J Interact Des Manuf, *Vol. 3:247-263*, 2009.
- Pierre L., Teissandier D., Nadeau J.P., *Qualification of turbine architectures in a multiphysical approach: application to a turboshaft engine*, Mechanism And Machine Theory, *Vol. 74: 82-101*, 2014.
- Homri L., Teissandier D., Ballu A., *Tolerance analysis by polytopes: taking into account degrees of freedom with cap half-spaces*, Computer-Aided Design, *accepted after minor revisions*, 2014.