ISSN 1831-9424

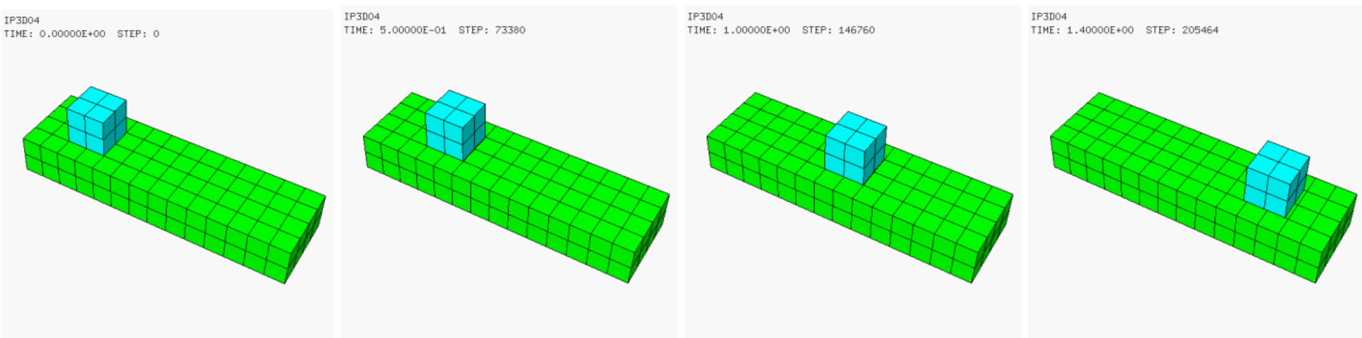
European Commission

JRC Technical Report

Correction of friction implementation in the LM-based PINB contact model of EUROPLEXUS

Casadei, F., Markovic, D., Valsamos G., Larcher M.

2023



Joint Research Centre

EUR 31597 EN

Contact information
Name: Martin Larcher
Address: JRC Ispra
Email: martin.larcher@ec.europa.eu

EU Science Hub
https://joint-research-centre.ec.europa.eu

JRC134224

EUR 31597 EN

PDF ISBN 978-92-68-05668-4 ISSN 1831-9424 doi:10.2760/180734 KJ-NA-31-597-EN-N

Luxembourg: Publications Office of the European Union, 2023

How to cite this report: Casadei, F., Markovic, D., Valsamos G., Larcher M., *Correction of friction implementation in the LM-based PINB contact model of EUROPLEXUS*, Publications Office of the European Union, Luxembourg, 2023, doi:10.2760/180734, JRC134224.

Correction of friction implementation in the LM-based PINB contact model of EUROPLEXUS

F. Casadei[1,2], D. Markovic[2], G. Valsamos[3], and M. Larcher[2]

[1] *Active Senior*
[2] *European Commission, Joint Research Centre*
Directorate for Space, Security and Migration
Safety and Security of Buildings Unit
[3] *European Commission, Joint Research Centre*
Directorate for Space, Security and Migration
Technology Innovation in Security Unit

Sunday 6th August, 2023 @ 16:24

Contents

List of Tables

List of Figures

Abstract

This report presents some notes on the use of pinball-based contact-impact models with friction by the method of Lagrange Multipliers (LM) in EUROPLEXUS [1] (also abbreviated as EPX). The code is jointly developed by the French Commissariat à l'Energie Atomique (CEA DMT Saclay) and by EC-JRC. Its application domain is the numerical simulation of fast transient phenomena such as explosions, crashes and impacts in complex three-dimensional fluid-structure systems.

Recently, EPX is being used to simulate vehicle crash against obstacles such as road barriers for safety and security studies. These studies involve complex contact-impact scenarios, which in EPX are typically modelled by the method of Lagrange Multipliers (LM), although other methods (e.g. penalty-based) are also available in the code. In some test cases, the effect of friction may be important and it must be included in the numerical simulations.

Recent simulations concerning (elastic) impact tests with friction using the LM-based version of the pinball contact model have revealed some malfunctionings, which may be attributed to the friction model. Therefore, it has been proposed to completely review the formulation and implementation of the friction algorithm in the LM-based pinball contact model. The model used a single-tangent formulation that could lead to problems. It has now been replaced by the same friction model used by the sliding surface model (GLIS), which uses a more robust two-tangent formulation.

Despite this improvement, and the correction of other minor issues, the LM version of the PINB contact model remains somewhat fragile in cases with friction, and the penalty-based version should preferably be used in industrial applications. Although not all issues have been solved yet, the present document may serve as a useful reference for the users, to highlight the problematic cases.

Keywords: *Pinballs, Lagrange Multipliers, Contact-Impact, Friction.*

Foreword

This report is part of a large series of scientific-technical documents that are meant to provide essential information and documentation to users and developers of the EUROPLEXUS code. EUROPLEXUS (also abbreviated as EPX) is a computer code jointly developed by the French Commissariat à l'Energie Atomique (CEA DMT Saclay) and by EC-Joint Research Centre (JRC Ispra) within the framework of contractual agreements between the two research bodies.

EPX is a mature, general-purpose Finite Element and Finite Volume explicit code under active development since 1999, for the simulation of fast transient dynamic events in complex fluid-structure systems. It is an evolution of its ancestor PLEXIS-3C, which was also jointly developed by CEA and JRC in the 1980s and early '90s.

The code has been traditionally used in safety studies, ranging from nuclear reactors, to energy plants, to chemical and industrial plants, off-shore structures, car and road barrier crashes, among others. More recently it has proven a very useful tool in providing certified and independent numerical solutions in support of EC policies regarding the security of critical infrastructures and public spaces (like buildings, train and metro stations and carriages, etc.), which may be vulnerable to terrorist attacks or to natural disasters.

While being mainly of technical nature, the information contained in this series of reports is an invaluable source of reference for the users (as a complement to the User's manual) but also in particular for the developers of EPX. New models made available in the code are described in detail from the theoretical viewpoint. Several verification and application examples are also usually provided, in order to illustrate the practical use and to verify the correct functioning of the models.

Usually, at the end of each report an Appendix lists the input files that were used to produce the examples presented in the report. This allows users to re-run the test cases with EPX at any time and to use them as a basis for their own numerical simulations.

A complete list of the reports (produced both at JRC and at CEA) in this series can be found in the Bibliography section of the EPX User's manual [1].

1 Introduction

This report presents some corrections that have been recently implemented in the friction part of the Lagrange Multiplier (LM) based pinball (PINB) contact model of EUROPLEXUS.

EUROPLEXUS [1] (also abbreviated as EPX) is a computer code jointly developed by the French Commissariat à l'Energie Atomique (CEA DMT Saclay) and by EC-JRC. The code application domain is the numerical simulation of fast transient phenomena such as explosions, crashes and impacts in complex three-dimensional fluid-structure systems. The Cast3m [2] software from CEA is used as a pre-processor to EPX when it is necessary to generate complex meshes.

1.1 Contact

The most popular contact algorithms available in Finite Element (FE) computer codes are probably the so-called slide line (in 2D) and slide surface (in 3D) algorithms proposed by Hallquist and Benson [3,4]. They are based on the notion of penetration of slave nodes into master segments (in 2D) or into master surfaces (in 3D), like e.g. the `GLIS` contact model in EPX. These algorithms suffer from a number of geometrically pathological cases in which physical penetration is not detected.

1.1.1 The pinball (`PINB`) method

The pinball method proposed by Belytschko and co-workers from the late 80's [5–12] for application in impact problems with perforation is much more robust as concerns penetration detection. The pinball contact-impact method has been implemented in EPX in [20–24,26], initially based on strong coupling via a Lagrange-multiplier (LM) based solution strategy of the contact constraints (see [13] for details of the method) and more recently by using weak coupling based on a penalty approach, see [25]. The latter report also contains a description of the implementation of Assembled Surface Normals (ASNs) in the pinball model, by an algorithm inspired to the one proposed by Belytschko in reference [5].

In reference [28] the pinball (`PINB`) model is generalized in order to be compatible with mesh adaptivity, i.e. with automatic mesh refinement and un-refinement. Both adaptivity-driven contact and contact-driven adaptivity paradigms are examined in detail.

The original pinball model of Belytschko and its implementation in EPX (`PINB` keyword) are based upon spherical pinballs. This approach is extremely robust and efficient in detecting penetration, but suffers from certain drawbacks when it comes to imposing contact constraints.

When dealing with slender elements (highly deformed continuum elements or structural members such as bars, beams and shells) the basic pinball method which associates a sphere with each element is no longer applicable and a hierarchic pinball method, consisting of splitting each penetrating pinball into a series of smaller pinballs (recursively, until a certain minimum size is reached) must be used instead, to enhance the resolution of penetration detection. Apart from the complexity and relative inefficiency of a recursive approach, difficulties arise when imposing multiple contacts between sub-pinballs with the LM method, because in this case redundant constraints may be generated and the system of constraints may become singular (see e.g. Section 7 of reference [21]). Various techniques have been devised in order to get rid of redundant constraints, but this is a complex and inefficient task, very difficult to perform in general terms. The penalty approach does not suffer from this limitation, but it needs some tuning parameters, which render solutions more laborious (besides being non-unique) and less reliable for the user, than the LM method.

Another difficulty in dealing with spherical-only pinballs is the determination of the local contact direction. The simplistic approach of using the line joining the two contacting pinball centres is sufficient in some cases (e.g. in fast impact with perforation) but may introduce large errors in other situations (such as sliding-dominated contact between smooth bodies). Recent (re-)developmeent of the ASN technique (see [25]) which associates a unique normal direction with each pinball or subpinball and then introduces rules for computing a "better" contact direction than the simple centres-joining line can alleviate these problems in many, but not in all, cases.

The pinball model originally implemented in EPX (until late 2016) assumed friction-less contact. However, friction is a very important phenomenon in many realistic applications. In reference [29] a first attempt was made to include friction in pinball-based (`PINB`) contacts. However, only the decoupled (penalty-based) version of the pinball model (`LINK DECO PINB PENA`) was detailed in reference [29] and in the numerical examples presented therein.

It is believed that, on the same occasion, friction had been tentatively added also to the coupled (LM-based) version of the pinball model (since it is still present in the code as of this writing). However, the LM-based pinball model with friction was never thoroughly tested so far and the decoupled version was used in all applications ever since, whenever friction was present. Furthermore, as of this writing (22 May 2023), there exist no tests involving LM-based pinball contact with friction in the EPX non-regression suite (while there are some tests verifying the penalty based pinballs with friction).

The fact that the contact normals are not accurately computed by the standard pinball (`PINB`) model, and that this model may lead to multiple and possibly redundant constraints, represents a serious drawback in the formulation and implementation of a friction model to be associated with the `PINB` contact algorithm.

1.1.2 The generalized pinball (`GPIN`) method

In a recent and still ongoing work [36] we explore the possibility of using so-called *generalized* pinballs (GPINs), of various geometrical shapes (not only spherical), in an attempt to avoid the problems highlighted above (especially those affecting the LM method), while retaining as far as possible the robustness and simplicity of the original pinball approach. The use of a variety of shapes allows to get rid of the necessity of a hierarchic procedure at the expense, however, of more complex penetration checks than with spheres. This should ensure that no (or fewer) redundant constraints appear and therefore their elimination is no longer necessary, with positive effects also on the treatment of rebound and of friction.

In the present work, the friction model introduced in `PINB` in reference [29], is corrected along the lines recently proposed by Bung and Potapov [39]. The same algorithm is implemented also in `GPIN`, by taking advantage of the fact that contact normals are much better defined in the `GPIN` model, and also that redundant constraints are avoided by construction.

2 Formulation of the friction model

The friction model adopted here is inspired by the simple Coulomb-type (dry) friction model already available in EPX for the sliding lines and sliding surfaces contact model (`GLIS`) developed at CEA, see references [14–19, 27].

2.1 Coulomb model

The classical Coulomb friction model, suitable for contact between dry (non-lubricated) solids, distinguishes between *static* friction and *kinetic* friction between two contacting bodies.

The static case occurs when the relative velocity of the two bodies along the contacting surface is null. In this case, friction generates a force directed along the tangent to the contact interface, which contrasts the mutual sliding of the bodies. The friction force is equal and opposite to the relative internal force along the tangent (i.e. the force that would tend to create sliding between the bodies), but only up to a limiting value equal to μ_s times the normal contact force:

$$F_t^s = \min\left(F_t^{\text{int}}, \mu_s F_n\right) \tag{1}$$

The kinetic case occurs when there is relative sliding of the two bodies along the contact surface, i.e. when there is a non-zero relative velocity of the two bodies along the tangent to the contact interface. In this case, friction generates a tangential force component along the direction of the relative velocity and opposed to it, whose magnitude is independent of the relative velocity, and equal to μ_k times the normal contact force:

$$F_t^k = \mu_k F_n \tag{2}$$

2.1.1 Interpolation of the friction coefficient

Experience shows that for most materials it is $\mu_k < \mu_s$ (indicating that once two previously sticking bodies start sliding with respect to each other it becomes easier for them to continue sliding) and that the passage between static and kinetic friction regime is rather abrupt. However, dealing with a discontinuous state of friction in a numerical model may be challenging and may lead to numerical instabilities. Therefore, it is preferred to use a single friction regime in the model, where the friction coefficient is continuously interpolated between the static and the kinetic values. Thus, the kinetic value of the coefficient will be reached only for very large (ideally infinite) relative velocity. From reference [15], the friction coefficient μ is expressed by:

$$\mu = \mu_k + (\mu_s - \mu_k)\, e^{-\gamma |v_{rt}|} \tag{3}$$

where v_{rt} is the tangential component of the relative velocity vector $\boldsymbol{v}_r$ and γ is a decay parameter having as dimension the inverse of a velocity.

When $v_{rt} = 0$ we have $\mu = \mu_s$ while, for increasing values of $|v_{rt}|$, the value of μ tends asymptotically to μ_k which, as already noted, is typically smaller than μ_s. The greater is the decay parameter γ, the faster is the decay of μ towards the limit value μ_k.

A practical procedure for choosing the value of γ is as follows. By indicating with α the value of the exponential:

$$\alpha = e^{-\gamma |v_{rt}|} \tag{4}$$

eq. (3) becomes:

$$\mu = \mu_k + \alpha\,(\mu_s - \mu_k) \tag{5}$$

From this one obtains for α:

$$\alpha = \frac{\mu - \mu_k}{\mu_s - \mu_k} \tag{6}$$

For example, if $\alpha = 0.01$ then the value of μ becomes equal to the asymptotic value μ_k within 1 % of the difference $(\mu_s - \mu_k)$ for the values of γ and v_{rt} which, combined, give α according to (4).

To determine the γ parameter the user may proceed as follows. From (4) one has:

$$-\gamma\,|v_{rt}| = \ln\alpha \tag{7}$$

or:

$$\gamma = -\frac{\ln\alpha}{|v_{rt}|} \tag{8}$$

By choosing the desired tolerance α and the corresponding value of the relative tangential velocity $|v_{rt}|$, one can compute γ from (8). For example, assume that we want the friction coefficient to reach the kinetic value with a tolerance of 1 % (of the difference between the static and kinetic coefficients) at a relative velocity of 2 m/s or higher. Then we get from (8):

$$\gamma = -\frac{\ln 0.01}{2.0} = 2.3025 \tag{9}$$

As an example, Figure 1 shows the variation of μ as a function of $|v_{rt}|$ (in the range 0.0 to 2.0 m/s) for $\mu_s = 0.3$, $\mu_k = 0.1$ and various values of γ. Note that, for $\gamma = 0$, from (3) one obtains $\mu = \mu_S$, *independently* from the velocity $|v_{rt}|$ *and* from the value of μ_k that has been chosen.

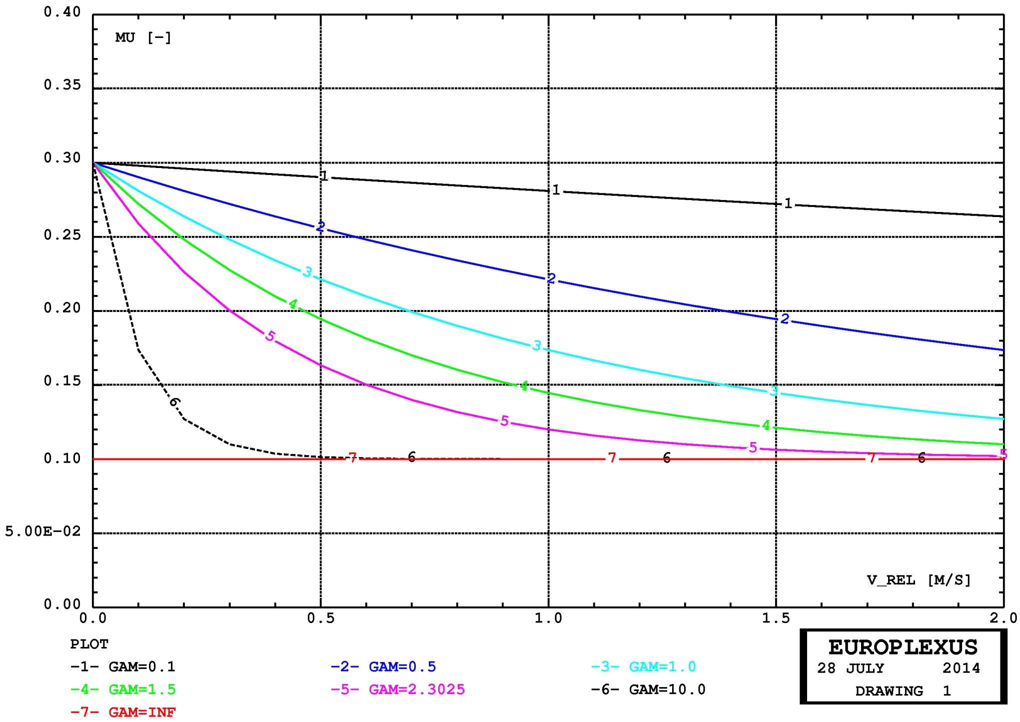


Figure 1: Friction coefficient *vs.* relative velocity for $\mu_s = 0.3$, $\mu_k = 0.1$ and various values of γ.

By the input directives, different friction values can be independently assigned to each potentially contacting body. If no `FRIC` keyword is specified in a `BODY` or `SELF` directive, then no friction is associated by default to the corresponding body. If two bodies come into contact, and none or only one of them has an associated friction, then no friction is computed for the contact. If *both* bodies have an associated friction, then the values μ_1 and μ_2 are evaluated with (3) for each body and then the *minimum* of the two results is retained for the present contact:

$$\mu = \min(\mu_1, \mu_2) \tag{10}$$

Note that this rule corrects and replaces the strategy that was adopted in the `PINB` model prior to the present work, namely that the values μ_s, μ_k and γ of the body with the *lowest* μ_k were retained for the contact between these two bodies.

3 Observed malfunctionings

Recently some malfunctionings in contact applications with friction using the “strong” or “coupled” LM-based version of the pinball contact algorithm were noticed.

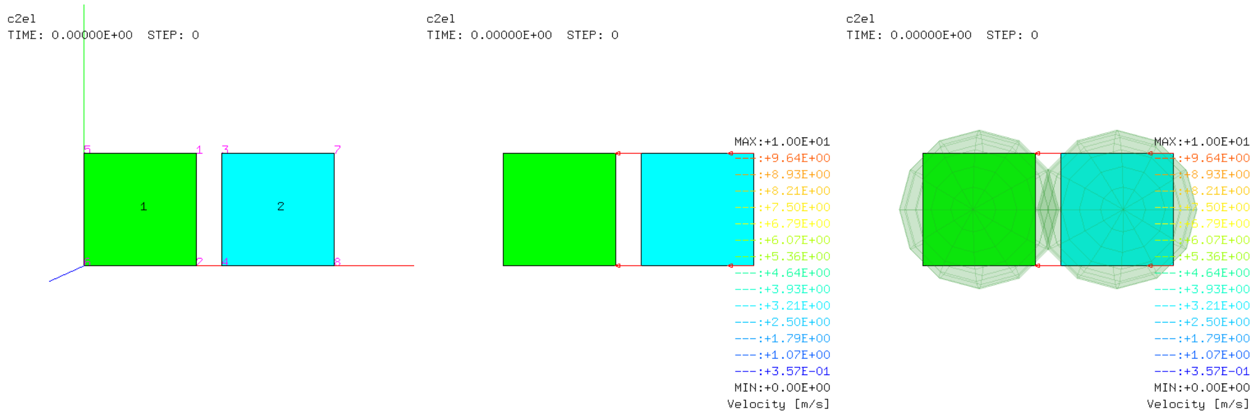
The application was then downsized to a very simple test involving just two elements, in order to study the origin of the malfunctioning. The simulations performed are summarized in Table 1.

Test	Mesh	Impact	Fric.	Split	Renu.	Contact @ step	Groups	Links	Solution
C2EL	2 Q42L	Normal	YES	`NONE`	(`RENU`)	18	1	2	Correct
C2EL_B	2 Q42L	Oblique	YES	`NONE`	(`RENU`)	11	1	4	Wrong
C2EL2	2 Q42L	Normal	YES	(`DOF`)	(`RENU`)	18	1	4	Wrong
C2EL_B2	2 Q42L	Oblique	YES	(`DOF`)	(`RENU`)	11	1	4	Wrong
C2EL03	2 Q42L	Normal	YES	`NONE`	`NORE`	18	1	2	Correct
C2EL04	2 Q42L	Oblique	YES	`NONE`	`NORE`	11	1	4	Correct
C2EL05	2 Q42L	Normal	NO	`NONE`	`NORE`	18	1	2	Correct
C2EL06	2 Q42L	Oblique	NO	`NONE`	`NORE`	11	1	2	Correct

Table 1: Simulations performed with the standard version of EPX.

3.1 Case C2EL

The first test considered is named C2EL and consists of just two unconnected square elements in 2D (plane stress). The second element has an initial velocity and hits the first element. Figure 2 shows the initial geometry and the initial conditions of the problem.



(a) Mesh (b) Velocities (c) Parent pinballs

Figure 2: Initial geometry and initial conditions of test C2EL.

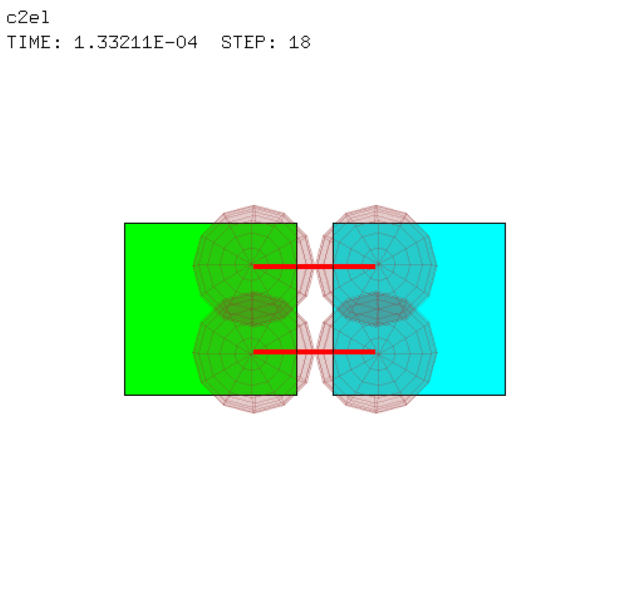
The first element (target) is assigned a very dense and stiff material ($\rho = 7.8 \times 10^6$ kg/m^3, $E = 2.1 \times 10^{12}$ Pa, resulting in $c = 518.9$ m/s) while the second element (projectile) has values typical of steel. Each element is set to represent a separate body as far as contact is concerned. Hierarchic pinballs (`MLEV 1`) are applied, with the penalty method (`PENA`). Friction is included, with $\mu_s = \mu_d = 0.9$ and $\gamma = 0.0$, which causes the friction coefficient μ to be equal to the static one μ_s, irrespective of the relative tangential velocity between the contacting bodies.

The initial velocity of the projectile is set as shown in Figure 2(b), so it will result in a centered and perfectly normal impact. The `SPLT NONE` sub-directive is activated in the `LINK COUP` directive. The `LIAJ` option is set, but with `ALPH 1` (which is the default), thus resulting in the standard way of imposing constraints in EPX, where the constrained velocities are those at $n + 3/2$.

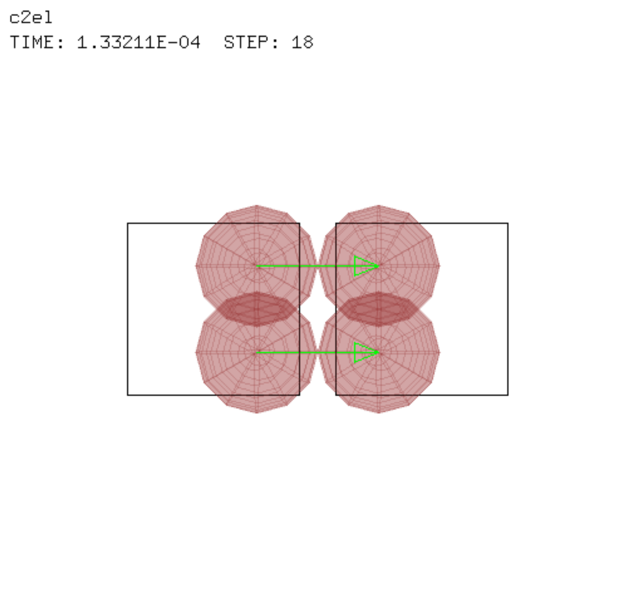
The EPX input file reads:

```
C2EL
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -10.0
     LECT el2 TERM
  VITE 2 0.0
     LECT el2 TERM
  DEPL 1 0.022
     LECT el2 TERM
  DEPL 2 -0.0
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 18
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  0.00000E+00 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE  0.00000E+00 TOLE 1.E-3
FIN
```

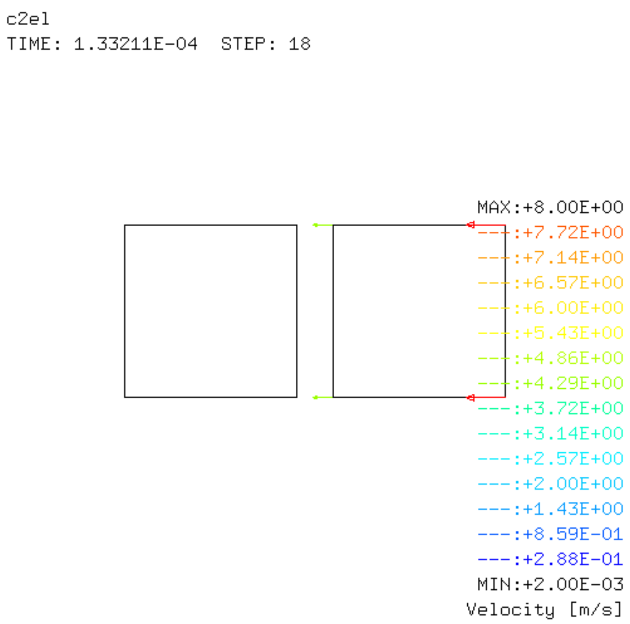
The first contact occurs at step 18, as shown in Figure 3. The solution seems correct. No tangential contact forces arise despite the presence of friction, since the impact is perfectly normal.



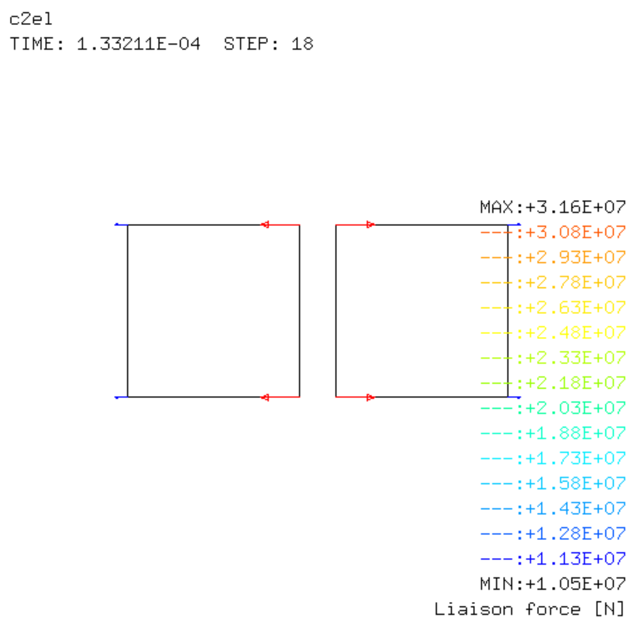
(a) Contact joints



(b) Contact normals



(c) Velocities



(d) Contact forces

Figure 3: First-contact conditions in test C2EL.

In the `.pin` file produced we find:

```
  STEP                 T   N_RAW  N_NCOL  N_RCEL  N_REBO
          0 0.00000E+00      0       0       0       0
          1 7.40063E-06      0       0       0       0
. . . (skip)
         17 1.25811E-04      0       0       0       0
         18 1.33211E-04      2       2       2       2
```

i.e., there are two raw contacts detected at step 18, and both of them are retained.

In the `.lks` file produced we find:

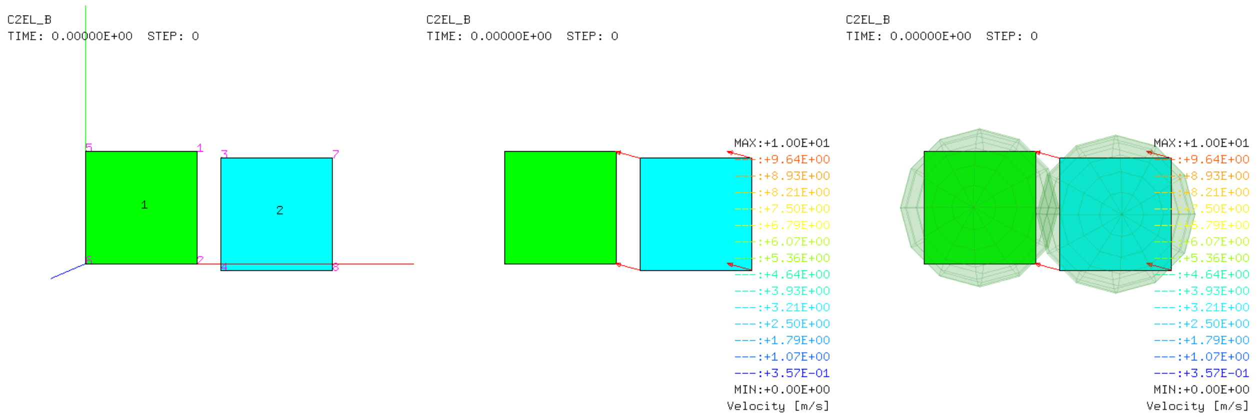
```
  STEP                 T   N_GPS    N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
          0 0.00000E+00      0       0       0       0
```

```
           1 7.40063E-06       0       0       0       0
 . . . (skip)
          17 1.25811E-04       0       0       0       0
          18 1.33211E-04       1       2       0       2  PINB=       2
```

i.e., there is a single group of links (due to `SPLT NONE`) containing two links at step 18, both of type `PINB`. It seems therefore that each contact generates only one link, along the normal, while no link along the tangent is generated, despite the presence of friction.

3.2 Case C2EL_B

This test is similar to C2EL (using `SPLT NONE`) but the initial position and initial velocities of the projectile are different and set in such a way that an oblique impact occurs, see Figure 4.



(a) Mesh (b) Velocities (c) Parent pinballs

Figure 4: Initial geometry and initial conditions of test C2EL_B.

The EPX input file reads (only the differences with respect to case C2EL are highlighted):

```
C2EL_B
 . . .
INIT
  VITE 1 -9.659258262890683
     LECT el2 TERM
  VITE 2 2.5881904510252074
     LECT el2 TERM
  DEPL 1 0.021250368178359503
     LECT el2 TERM
  DEPL 2 -0.005694018992255456
     LECT el2 TERM
 . . .
FIN
```

From this, one sees that the initial inclination of the velocities (and of the positions) is about 15°.

The first contact occurs at step 11, as shown in Figure 5. The solution seems correct. Some tangential contact forces seem to arise, as shown in Figure 5(d), due to the presence of friction, since the impact is not perfectly normal. However, deeper inspection will be needed to determine whether this is actually the case, since the contact normals are not perfectly horizontal, see Figure 5(b).

In the `.pin` file produced we find:

```
  STEP                T   N_RAW  N_NCOL  N_RCEL  N_REBO
           0 0.00000E+00      0       0       0       0
           1 7.40063E-06      0       0       0       0
 . . . (skip)
          10 7.40063E-05      0       0       0       0
          11 8.14070E-05      2       2       2       2
```

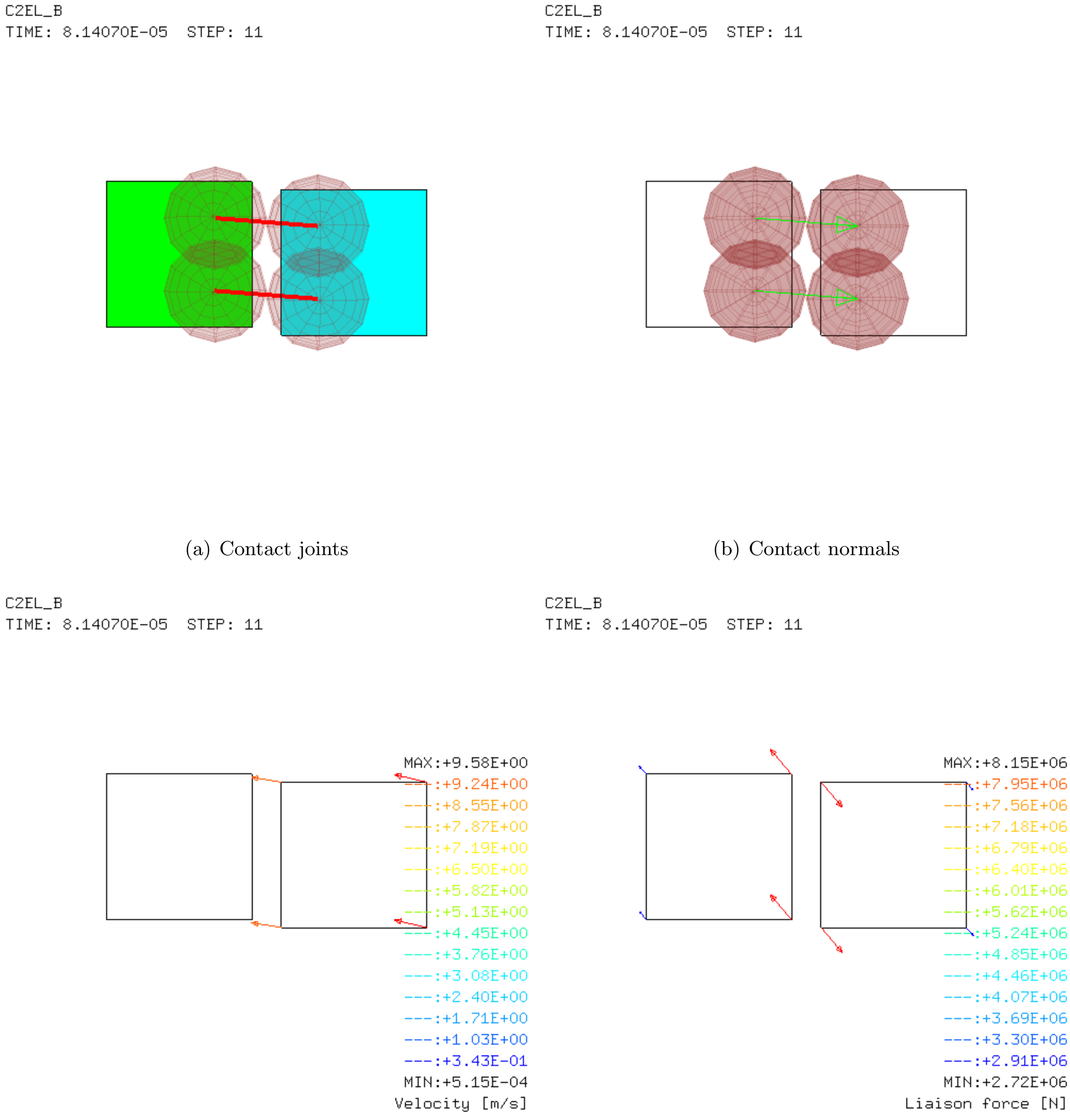
i.e., there are two raw contacts detected at step 11, and both of them are retained.

In the `.lks` file produced we find:

```
  STEP                T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
           0 0.00000E+00      0       0       0       0
           1 7.40063E-06      0       0       0       0
 . . . (skip)
          10 7.40063E-05      0       0       0       0
          11 8.14070E-05      1       4       0       4  PINB=         4
```

i.e., there there is a single group of links (due to `SPLT NONE`) containing four links at step 11, all of type `PINB`. It seems therefore that each contact generates two links, one along the normal and one (since we are in 2D here) along the tangent, due to the presence of friction.

By inspecting the listing, we see that at the moment first contact occurs (step 11) the contact normal is $\hat{\boldsymbol{n}}_{AB} = (0.997, -0.078)$. This gives the following tangent of the angle of inclination α of the normal with respect to the horizontal: $\tan\alpha = n_y/n_x = -0.078/0.997 = -0.07823$, i.e. $\alpha \approx -4.47°$. At the same instant, the inclination according to the positions can be obtained from the current coordinates of nodes 2 and 4: $P_2 = (0.1, 0.0)$, $P_4 = (1.20464 \times 10^{-1}, -5.48332 \times 10^{-3})$. This results in an inclination $\tan\beta = \Delta y/\Delta x = -5.48332 \times 10^{-3}/1.20464 \times 10^{-1} = -0.04552$, i.e. $\beta \approx -2.61°$.



(a) Contact joints

(b) Contact normals

(c) Velocities

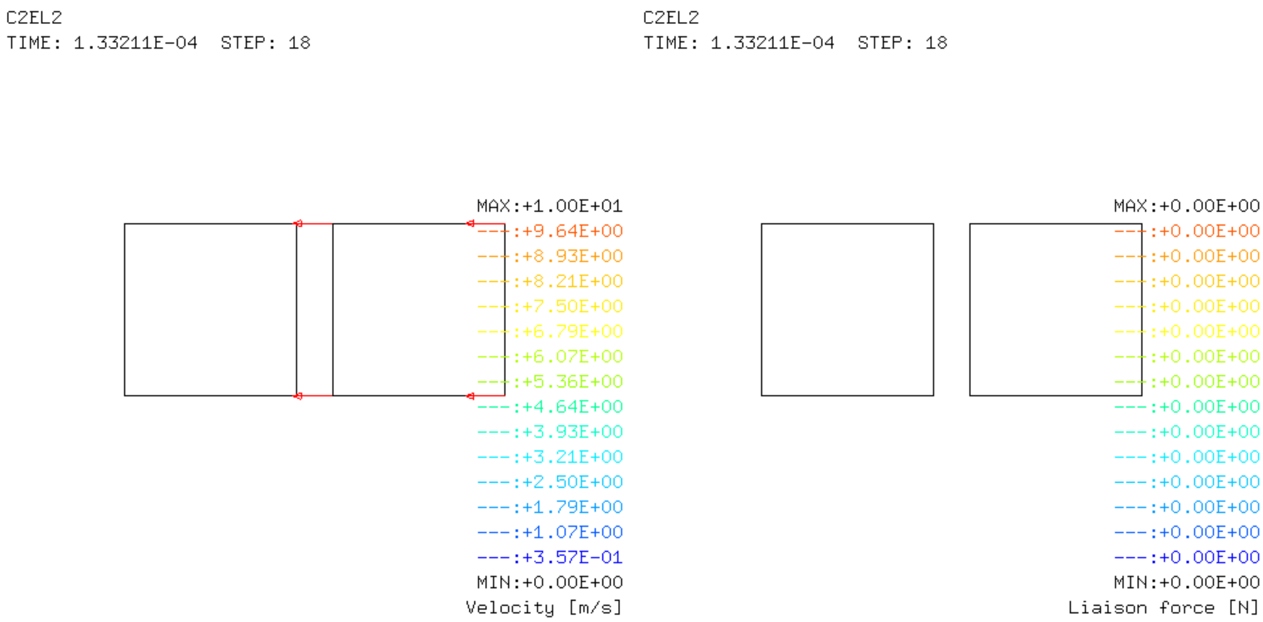
(d) Contact forces

Figure 5: First-contact conditions in test C2EL_B.

3.3 Case C2EL2

This test is similar to C2EL (normal impact), but we remove the `SPLT NONE` keywords. Therefore, as per default, the links system will be split into mutually independent sub-systems according to a per-dof rule.

The solution looks clearly wrong, in fact no contact forces (not even normal forces) arise at the first contact, see Figure 2(b). All other contact conditions are identical to those of test C2EL, cfr. Figure 3.



(a) Velocities (b) Contact forces

Figure 6: First-contact conditions in test C2EL2.

In the `.pin` file produced we find:

```
 STEP             T   N_RAW  N_NCOL  N_RCEL  N_REBO
        0 0.00000E+00      0       0       0       0
        1 7.40063E-06      0       0       0       0
. . . (skip)
       17 1.25811E-04      0       0       0       0
       18 1.33211E-04      2       2       2       2
```

i.e., there are two raw contacts detected at step 18, and both of them are retained (exactly like in case C2EL).

In the `.lks` file produced we find:

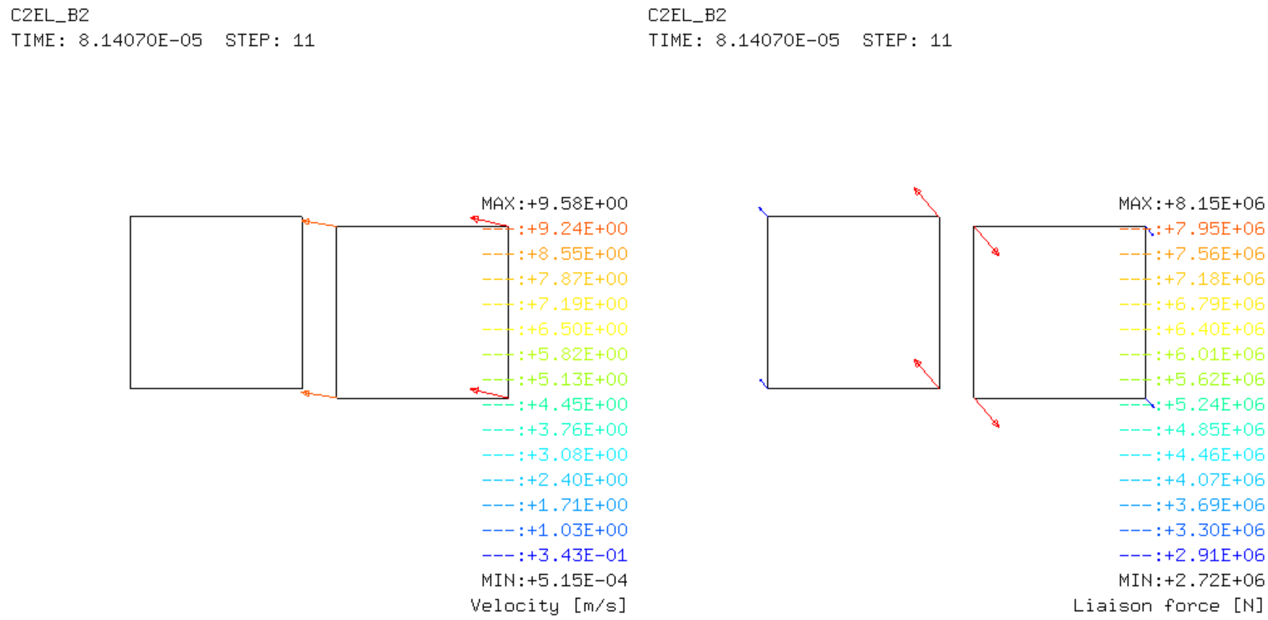
```
 STEP             T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
        0 0.00000E+00      0       0       0       0
        1 7.40063E-06      0       0       0       0
. . . (skip)
       17 1.25811E-04      0       0       0       0
       18 1.33211E-04      1       4       0       4  PINB=       4
```

i.e., there is a single group of links like in case C2EL (despite the absence of `SPLT NONE`), but this containing four (not just two) links at step 18, al of type `PINB`. It seems therefore that in this case each contact generates two links, one along the normal and the other along the tangent, while in case C2EL only one link per contact was generated.

3.4 Case C2EL_B2

This test is similar to C2EL_B (oblique impact), but we remove the SPLT NONE keywords. Therefore, as per default, the links system will be split into mutually independent sub-systems according to a per-dof rule.

The solution (see Figure 7) is identical to that of case C2EL_B, which had been obtained with SPLT NONE, cfr. Figure 5.



(a) Velocities (b) Contact forces

Figure 7: First-contact conditions in test C2EL_B2.

In the .pin file produced we find:

```
  STEP               T   N_RAW  N_NCOL  N_RCEL  N_REBO
         0 0.00000E+00       0       0       0       0
         1 7.40063E-06       0       0       0       0
. . . (skip)
        10 7.40063E-05       0       0       0       0
        11 8.14070E-05       2       2       2       2
```

i.e., there are two raw contacts detected at step 11, and both of them are retained.

In the .lks file produced we find:

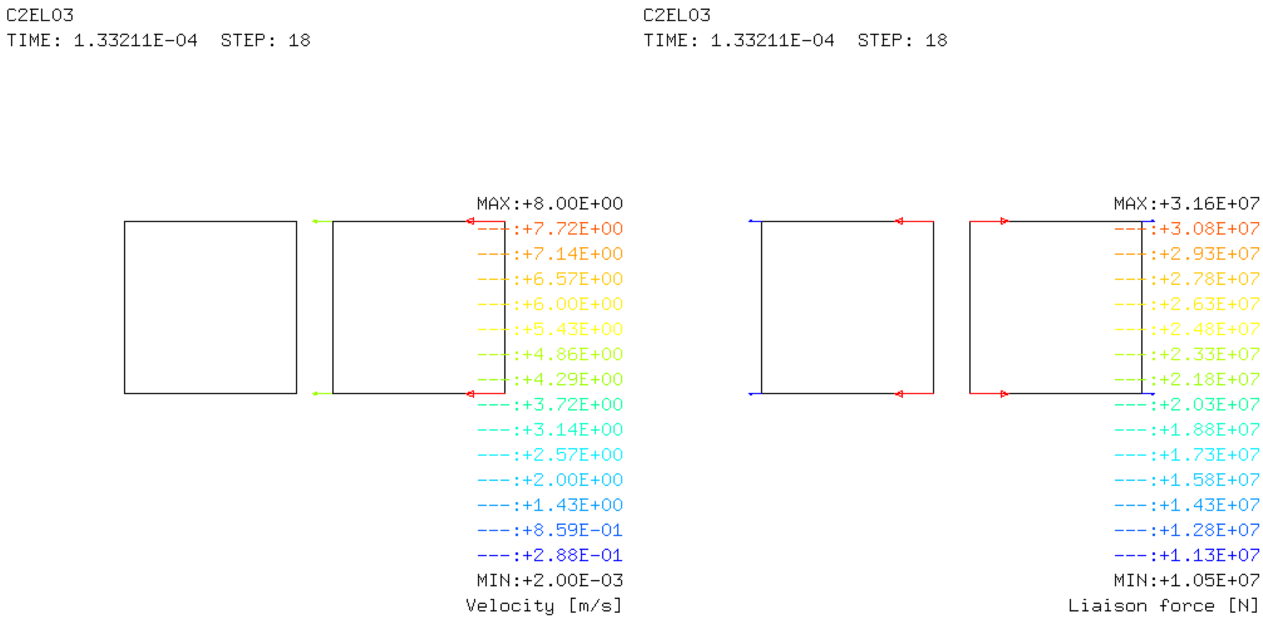
```
  STEP               T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
         0 0.00000E+00       0       0       0       0
         1 7.40063E-06       0       0       0       0
. . . (skip)
        10 7.40063E-05       0       0       0       0
        11 8.14070E-05       1       4       0       4  PINB=        4
```

i.e., there there is a single group of links (due to SPLT NONE) containing four links at step 11, all of type PINB, like in case C2EL_B. It seems therefore that each contact generates two links, one along the normal and one (since we are in 2D here) along the tangent, due to the presence of friction, like in case C2EL_B.

3.5 Case C2EL03

This test is similar to C2EL (normal impact using `SPLT NONE`) but we add the `NORE` keyword in the `LINK COUP` directive. This should tell EPX to refrain from renumbering the links.

The solution (see Figure 8) looks correct, as it is identical to that of case C2EL, which had been obtained by renumbering the links (as per default), cfr. Figure 3. In this case a perfectly normal impact occurs, so there are no tangential contact forces.



(a) Velocities (b) Contact forces

Figure 8: First-contact conditions in test C2EL03.

In the `.pin` file produced we find:

```
 STEP                T    N_RAW  N_NCOL  N_RCEL  N_REBO
         0 0.00000E+00        0       0       0       0
         1 7.40063E-06        0       0       0       0
. . . (skip)
        17 1.25811E-04        0       0       0       0
        18 1.33211E-04        2       2       2       2
```

i.e., there are two raw contacts detected at step 18, and both of them are retained.

In the `.lks` file produced we find:

```
 STEP                T    N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
         0 0.00000E+00        0       0       0       0
         1 7.40063E-06        0       0       0       0
. . . (skip)
        17 1.25811E-04        0       0       0       0
        18 1.33211E-04        1       2       0       2  PINB=        2
```

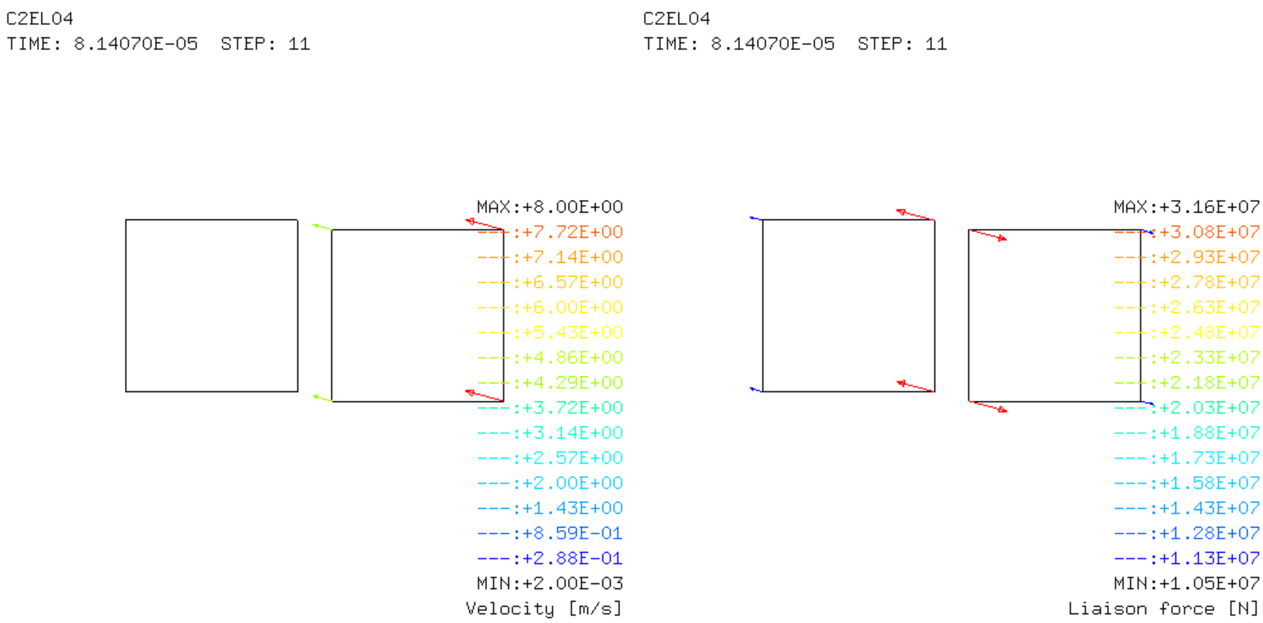
i.e., there there is a single group of links (due to `SPLT NONE`) containing two links at step 18, both of type `PINB`, like in case C2EL. It seems therefore that each contact generates only one link, along the normal, while no link along the tangent is generated, despite the presence of friction.

3.6 Case C2EL04

This test is similar to C2EL_B (oblique impact using `SPLT NONE`) but we add the `NORE` keyword in the `LINK COUP` directive.

The solution (see Figure 9) is different from that of case C2EL_B, cfr. Figure 5, both as concerns the velocities and the contact forces. This seems to indicate that the solution obtained in case C2EL_B (with renumbering) was wrong. Indeed, the contact forces now seem directed roughly along the contact normal, which was not the case in solution C2EL_B.

Only deeper inspection will reveal if there is a (slight) tangential component in the reactions (which would be plausible due to friction). In any case, the tangential components of the reactions in case C2EL_B look excessive, see Figure 5.



(a) Velocities (b) Contact forces

Figure 9: First-contact conditions in test C2EL04.

In the `.pin` file produced we find:

```
 STEP              T   N_RAW  N_NCOL  N_RCEL  N_REBO
        0 0.00000E+00      0       0       0       0
        1 7.40063E-06      0       0       0       0
. . . (skip)
       10 7.40063E-05      0       0       0       0
       11 8.14070E-05      2       2       2       2
```

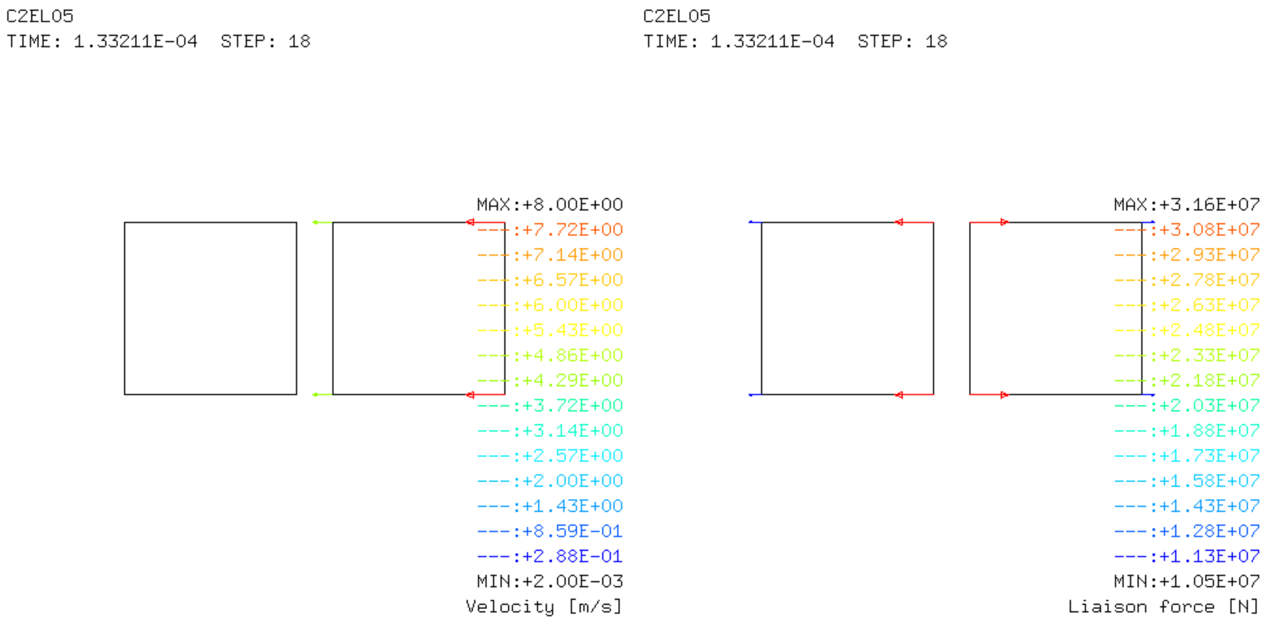
i.e., there are two raw contacts detected at step 11, and both of them are retained.
In the `.lks` file produced we find:

```
 STEP              T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
        0 0.00000E+00      0       0       0       0
        1 7.40063E-06      0       0       0       0
. . . (skip)
       10 7.40063E-05      0       0       0       0
       11 8.14070E-05      1       4       0       4  PINB=       4
```

i.e., there there is a single group of links (due to `SPLT NONE`) containing four links at step 11, all of type `PINB`, like in case C2EL_B. It seems therefore that each contact generates two links, one along the normal and one (since we are in 2D here) along the tangent, due to the presence of friction.

3.7 Case C2EL05

This test is similar to C2EL03 but has no friction. The solution (see Figure 10) looks correct, as it is identical to that of case C2EL03 which included friction but generated no tangential contact forces, since the contact is perfectly normal.



(a) Velocities (b) Contact forces

Figure 10: First-contact conditions in test C2EL05.

In the `.pin` file produced we find:

```
  STEP                 T   N_RAW  N_NCOL  N_RCEL  N_REBO
          0 0.00000E+00      0       0       0       0
          1 7.40063E-06      0       0       0       0
. . . (skip)
         17 1.25811E-04      0       0       0       0
         18 1.33211E-04      2       2       2       2
```

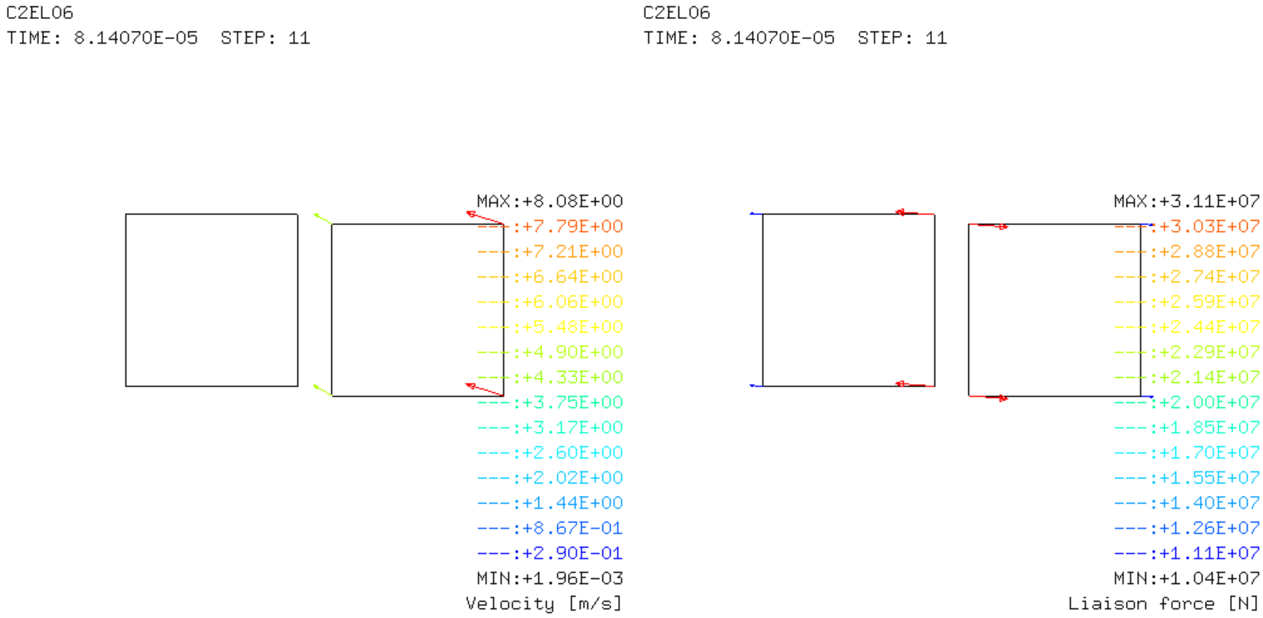
i.e., there are two raw contacts detected at step 18, and both of them are retained.
In the `.lks` file produced we find:

```
  STEP                 T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
          0 0.00000E+00      0       0       0       0
          1 7.40063E-06      0       0       0       0
. . . (skip)
         17 1.25811E-04      0       0       0       0
         18 1.33211E-04      1       2       0       2  PINB=        2
```

i.e., there there is a single group of links (due to `SPLT NONE`) containing two links at step 18, both of type `PINB`, like in case C2EL03. Each contact generates only one link, along the normal, and no link along the tangent is generated, because there is no friction.

3.8 Case C2EL06

This test is similar to C2EL04 but has no friction. The solution (see Figure 11) looks correct, and differs slightly from that of case C2EL04 which included friction. In particular, note that the contact forces are less inclined than in case C2EL04 (actually, they should be directed exactly along the contact normal, see verification below). This visually confirms that the contact forces in case C2EL04 did contain tangential components generated by the friction present in that case.



(a) Velocities (b) Contact forces

Figure 11: First-contact conditions in test C2EL06.

In the `.pin` file produced we find:

```
STEP             T   N_RAW  N_NCOL  N_RCEL  N_REBO
        0 0.00000E+00     0      0      0      0
        1 7.40063E-06     0      0      0      0
. . . (skip)
       10 7.40063E-05     0      0      0      0
       11 8.14070E-05     2      2      2      2
```

i.e., there are two raw contacts detected at step 11, and both of them are retained.
In the `.lks` file produced we find:

```
STEP             T   N_GPS   N_LKS  N_PLKS N_NPLKS  N_OF_LINKS_BY_TYPE
        0 0.00000E+00     0      0      0      0
        1 7.40063E-06     0      0      0      0
. . . (skip)
       10 7.40063E-05     0      0      0      0
       11 8.14070E-05     1      2      0      2  PINB=        2
```

i.e., there there is a single group of links (due to `SPLT NONE`) containing two links at step 11, both of type `PINB`, like in case C2EL04. Unlike case C2EL04 that had friction, each contact generates only one link, along the normal, and no link along the tangent is generated, because there is no friction.

We want to verify that the contact forces (reactions) are directed along the contact normal, i.e. that there are no tangential components. From the listing we see that, for both contacts, the (unit) contact normals are $\hat{\boldsymbol{n}}_{AB} = (0.997, -0.078)$. This gives the following tangent of the angle of inclination α of the normal with respect to the horizontal: $\tan\alpha = n_y/n_x = -0.078/0.997 = -0.07823$, i.e. $\alpha \approx -4.47°$. Clearly, the same contact normals were also obtained in case C2EL04 with friction.

Also from the listing, the contact (link) force in node 4 is $\boldsymbol{F}_{c4} = (3.09597 \times 10^7, -2.40920 \times 10^6)$. We obtain the following angle of inclination β of the force with respect to the horizontal: $\tan\beta = F_y/F_x = -2.40920 \times 10^6/3.09597 \times 10^7 = -0.07782$, i.e. $\beta \approx -4.45°$, in good agreement with the value of α obtained above. This confirms that there are no tangential force components in this case.

Instead, in case C2EL04 (with friction) we obtained $\boldsymbol{F}_{c4} = (3.05110 \times 10^7, -8.17540 \times 10^6)$, corresponding to an angle of inclination β of the force with respect to the horizontal: $\tan\beta = F_y/F_x = -8.17540 \times 10^6/3.05110 \times 10^7 = -0.2678$, i.e. $\beta \approx -15.0°$. Therefore, in case C2EL04 the friction caused non-negligible tangential components of the contact forces.

3.9 Malfunctioning pattern

From the tests performed so far, the following pattern of malfunctioning emerges:

- The renumbering of links (`RENU`, which is the default) causes wrong solutions. The only exception is case C2EL, in which there happen to be only two links due to the fact that the impact is perfectly normal.
- All solutions without renumbering (`NORE`) look correct.
- It is not clear why in case C2EL2 (normal impact), which is identical to C2EL apart from the splitting (`DOF` instead of `NONE`), there are four links instead of two, since the links end up in a single group like in case C2EL. However, this causes the solution to be wrong.
- The presence or not of `SPLT NONE` has no clear effect in these examples. The renumbering issue should be addressed first, before drawing any conclusions.

4 Current state of the model and implemented corrections

We start by inspecting the available documentation, i.e. the available reports concerning the friction algorithm, in particular as applied to the pinball contact method, are inspected first.

4.1 Report [29]

The report [29] entitled "Implementation of friction in the pinball-based contact-impact model of EUROPLEXUS" first describes the Coulomb friction model and its algorithmic implementation in the GLIS (sliding surfaces) contact model, more precisely in the coupled, LM-based version of GLIS.

The friction model described there is the original version from CEA, i.e. before the correction implemented by EDF and CEA in 2018 [39]. Such correction aimed at avoiding some pathological situations concerning the definition of the tangent direction (in 3D). Instead of using a single tangent direction in 3D, determined by using the relative velocity (or the relative internal force), a local reference frame is built by always using using two tangent directions. This introduces an extra link (three links instead of two, in 3D) in contacts with friction, but avoids the above mentioned pathological cases.

Next, the report [29] describes a simplified version of the friction model and its application to the penalty-based version of the method (`PINB PENA`). The algorithm is relatively different from the one for the coupled, LM-based case, because each constraint (i.e. the normal one and the tangential one) is treated independently in the penalty formulation (apart of course from checking the Coulomb condition, which involves both constraints). No description is given of the friction model implemented (and currently available) in the coupled, LM-based version of the pinball method. Upon inspection of the code, at the moment of this writing the LM pinball friction model still uses the original uncorrected single-tangent formulation (in 3D).

All numerical examples in the report [29] use the decoupled, penalty-based version of the pinball method. The coupled, L-M based version of the pinball friction algorithm (which is actually implemented in EPX) is not addressed in that report, which is probably incomplete with respect to its original intentions. It is therefore felt that the title of that report is misleading, and should be modified (in case of a reprint) to "Implementation of friction in the penalty-based pinball contact-impact model of EUROPLEXUS".

4.2 Report [37]

A second (yet unpublished) report exists on the subject, but is still in a draft state, see [37]. The title is "Treatment of friction in the PINB and GPIN contact-impact models of EUROPLEXUS" and it was evidently meant to document the treatment of friction in all the main contact models available in EPX, namely the PINB (basic pinballs), the GPIN (generalized pinballs, still under development), and the GLIS (3D sliding surfaces) method. Note in fact that GLIS is addressed in the report, although it is not mentioned in its title. Therefore, the title should probably be modified into "Treatment of friction in the GLIS, PINB and GPIN contact-impact models of EUROPLEXUS". For completeness, we remark that the SLID contact model is still in an early stage of development and therefore it is better described in a dedicated report [35] rather than being also included in [37].

In its original intention, both the LM-based and the penalty-based version of the contact algorithms with friction had to be documented in the report, for each one of the contact models:

- The report starts by describing the LM-based friction algorithm for GLIS. Both the original version (single tangent) and the corrected version (two tangents), which is the one currently implemented in the code, are described.
- The penalty-based friction algorithm for GLIS is not (yet) described in [37]. It is not known whether, and to which extent, this friction model is already implemented in the code, despite being mentioned in the Users' Manual [1]. In any case, there seem to exist no theoretical report on the penalty (or uncoupled) version of GLIS (with or without friction) and there are no practical examples of its usage. The subject remains open and should be investigated with CEA.

- Coming to the PINB method, the LM-based friction model is not yet described. This part of the report will have to be completed in a future contribution.
- The penalty based friction model for PINB is then described. The description given is the same already present in the report [29].
- Finally, the GPIN model is addressed. The LM-based friction model for GPIN is described in detail, by highlighting the conceptual differences between GLIS and GPIN as concernds in particular the rebound algorithms, the role of slave and master nodes and the 2D case (which is not available in GLIS). The GPIN friction model uses two tangents (like the current GLIS), i.e. it already includes the correction introduced in reference [39].
- The penalty-based GPIN friction algorithm is not described because no penalty-based GPIN contact model has been formulated and implemented yet.

4.3 Current status of friction in the PINB model and needed improvements

From the information in the above mentioned reports, as well as from code inspection, it may be concluded that the LM-based version of the PINB model with friction is still in the status corresponding to the original GLIS model, i.e. it uses the single tangent formulation in 3D (as well as in 2D). The detailed model and its implementation in the code have not yet been documented so far and, indeed, the model has been rarely used so far (probably because of malfunctionings), in favor of the penalty-based version of the PINB model, in applications with friction.

The LM-based version of the PINB model with friction has to be upgraded similarly to GLIS (and GPIN) by introducing the correction of reference [39], i.e. using two tangents instead of one, in the 3D case, and has to be fully documented. A distinction between the 2D case and the 3D case will be required in the implementation. This subject can be considered a further development rather than a simple correction of the friction algorithm for the PINB contact model, and is treated in Section 6.

But, before performing the mentioned development, the malfunctionings described in Section 3 have to be (urgently) corrected. The corrections are described in detail in the next Section.

4.4 Preliminary considerations for the corrections

Before presenting the corrections, we briefly recall the general organization and some relevant specific aspects of the LM-based treatment of links in EPX, wich corresponds to the `LINK COUP` directive.

4.4.1 Constraints splitting

By default, the `SPLT DOF` splitting strategy is used. This means that the monolithic system of constraints is split into a number of independent sub-systems, which are then solved independently (in the hope, not always fulfilled, of achieving a faster solution of the implicit linear constraints problem).

The `DOF` case means that the splitting is done on the base of degrees of freedom. In other words, the resulting sub-systems are disjoint as concerns the dofs involved. Each constrained dof will appear in one and only one sub-system. This is the most extreme (i.e. the finest-grain) type of splitting. At the other extreme, we have `SPLT NONE`, meaning that no splitting is performed, so that the monolithic version of the system is retained and solved as a single linear problem.

Yet another possibility is `SPLT NODE` where the splitting is node-based instead of dof-based, leading to an intermediate situation between `DOF` and `NONE` (intermediate-grain splitting).

4.4.2 Requirements for constraints manipulation

Normally, constraints can be manipulated independently from one another, apart from the obvious necessity of ending up in the same linear sub-system, if they involve the same dofs or the same nodes). However, in the case of contact with friction, the two (or three) constraints that are imposed for a certain contact must not only be kept together (one after the other) but they must also respect a pre-determined order, e.g., the tangent constraint(s) before the normal constraint, for the algorithm to work properly.

This is because, when treating the friction at a certain contact, the code will need to access all the corresponding constraints (tangent and normal) at the same time, hence the necessity of precisely addressing each component. Therefore, in case of splitting, it is essential that all constraints relative to a certain contact:

- end up in the same sub-system, and
- remain in the order in which they were originally imposed.

Of course, the latter requirement might be threatened by the use of a renumbering algorithm which re-orders the links to achieve smaller bandwidth of the linear system and thus increase the efficiency of its solution.

4.4.3 Solution details

Once the system of constraint has been split into disjoint sub-systems, each sub-system is solved separately from the others. However, the solution method adopted depends upon the number of constraints in the sub-system, with the dispatching taking place in subroutine FE_TRUE_LIAIS (for links in a linked list) or in subroutine FE_LINKS (for links in a dynamic array) of module M_LINKS:

- If there is just one constraint, the solution is done in subroutine SOLVE_SINGLE_LIAISON of module M_LINKS. The solution method adopted is special (analytical or direct), since there is only one constraint and not an actual system.
- If there are more than two constraints in the system, the solution is done in subroutine SOLVE_-GROUP of module M_LINKS. The general implicit solution method of the linear system (the method of Choleski, by default) is adopted in this case.
- If there are two constraints in the system, the solution is done either in subroutine SOLVE_TWO_-LIAISONS or in subroutine SOLVE_GROUP of module M_LINKS, depending upon the setting of the LINKS_SOL2 flag in M_LIAISONS_DATA. The first approach aims at increasing performance and treats the two constraints analytically without resorting to the full-fledged general solution method (valid for any number of constraints) adopted in the second approach.

According to the Users' manual [1]: *The optional keyword* `SOL2` *can be used after* `LINK COUP` *to choose closed-form solution for groups of links containing just two links (in addition to groups containing just one link). By default, all groups of links containing more than one links are solved by the general numerical method (Choleski's method by default). In some cases, the closed-form solution may be more efficient.*

It seems therefore that, since the `SOL2` keyword was never specified, the general (SOLVE_GROUP) routine was always used in all of the examples presented so far in the present work. We also note, incidentally, that at the moment of this writing there are no EPX non-regression benchmarks using the `SOL2` keyword which, therefore, seems to remain untested.

4.4.4 Coverage of the solution routines

We observe that, in the case of a set of contact constraints with friction, by taking for granted that such constraints end up in the same sub-system (see discussion below), there should always be at least two (possibly at least three) constraints in the concerned sub-system.

In the case of LM-based GLIS with friction, which only addresses the 3D case and uses two tangents in its current version, there will always be three constraints for each contact. So the solution will always be obtained in SOLVE_GROUP.

In the case of LM-based PINB with friction, the current implementation (to be upgraded in the near future as described above) uses just one tangent, and this both in 2D and in 3D. Therefore, in principle (e.g. for an isolated contact) one might get a two-constraint sub-system, which would then be treated in SOLVE_TWO_LIAISONS if `SOL2` is set. In the more general case of a non-isolated pinball contact the sub-system will probably end up containing more than two constraints, so the solution will be performed in SOLVE_GROUP.

4.5 Corrections implemented in the present work

We now present in detail all the corrections implemented in the code as part of the present work.

4.5.1 Corrections in subroutine SOLVE_TWO_LIAISONS

From code inspection it has been found out that the subroutine SOLVE_TWO_LIAISONS is not currently equipped to deal with contact friction, whatever be the contact model used.

In order to contain the complexity of the code it is therefore decided, instead of further complicating the routine, to raise an error if either of the two constraints received by the routine is a tangent constraint of some contact model. This should stop the code with an informative message in the (unlikely) event that a user sets the `SOL2` optional keyword and a two-constraint sub-system concerning a contact is generated, a possibility that would be hard to exclude *a priori*, e.g. with the PINB model. In that case, removing the `SOL2` keyword from the input will bypass SOLVE_TWO_LIAISONS in favor of SOLVE_GROUP and the problem should be avoided.

The following code (in red) is added to the routine:

```
      SUBROUTINE SOLVE_TWO_LIAISONS . . .
 . . .
      INTEGER :: . . . . , II
 . . .
      NLIE = 2
      L1 => FIRST (LL)
      L2 => LAST (LL)
      ITYP1 = L1%LTYPE
      ITYP2 = L2%LTYPE
!fc
!fc 20/05/2023
!fc This routine is not equipped for the treatment of friction
!fc in contacts treated by GLIS, IMPA, PINB, GPIN or SLID.
!fc Therefore, we issue an error message if either of the two constraints
!fc L1 or L2 is a tangent constraint of such contact models.
      SELECT CASE (ITYP1) ! Type of first constraint (L1)
      CASE (19:20) ! GLIS, IMPA
* GLIS is only used in 3D and there should be 3 constraints per contact
* with friction, and we should never end up in this routine,
* but we check nevertheless for safety ...
        II = L1%LDATA(2) ! 1, 2 or 3 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (31) ! PINB
        II = L1%LDATA(1) ! 1 = tangent, 0 = normal w/o fric, -1 = normal w. fr.
        IF (II /= 0) GO TO 998 ! Raise error and stop
      CASE (76) ! GPIN
        II = L1%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (99) ! SLID
        II = L1%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      END SELECT
      SELECT CASE (ITYP2) ! Type of second constraint (L2)
      CASE (19:20) ! GLIS, IMPA
* GLIS is only used in 3D and there should be 3 constraints per contact
* with friction, and we should never end up in this routine,
* but we check nevertheless for safety ...
        II = L2%LDATA(2) ! 1, 2 or 3 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (31) ! PINB
        II = L2%LDATA(1) ! 1 = tangent, 0 = normal w/o fric, -1 = normal w. fr.
        IF (II /= 0) GO TO 998 ! Raise error and stop
      CASE (76) ! GPIN
        II = L2%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (99) ! SLID
        II = L2%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      END SELECT
!fc 20/05/203
!fc
      LD1 = L1%NDLIE
      LD2 = L2%NDLIE
 . . .
  998 WRITE (BLABLA, 1005)
      CALL ERRMSS ('SOLVE_TWO_LIAISONS', BLABLA)
 1005 FORMAT('CANT TREAT CONTACT WITH FRICTION. REMOVE "SOL2"')
      CALL EPX_STOP('SOLVE_TWO_LIAISONS')
 . . .
      END SUBROUTINE SOLVE_TWO_LIAISONS
```

The added check concerns all the contact models available in EPX that might use some form of friction, namely GLIS, IMPA, PINB, GPIN and SLID.

4.5.2 Corrections in subroutine SOLVE_SINGLE_LIAISON

In theory one should never end up calling SOLVE_SINGLE_LIAISON by passing a tangent constraint of a contact model, since such constraint should always be treated together with the corresponding normal constraint of the same contact.

However, for safety reasons we add to subroutine SOLVE_SINGLE_LIAISON a check similar to what done above for SOLVE_TWO_LIAISONS:

```
      SUBROUTINE SOLVE_TWO_LIAISONS . . .
 . . .
      INTEGER :: . . . . , II
 . . .
      LTYP = L%LTYPE
!fc
!fc 20/05/2023
!fc This routine is not equipped for the treatment of friction
!fc in contacts treated by GLIS, IMPA, PINB, GPIN or SLID.
!fc Therefore, we issue an error message if the constraint
!fc L is a tangent constraint of such contact models.
      SELECT CASE (LTYP) ! Type of the constraint (L)
      CASE (19:20) ! GLIS, IMPA
* GLIS is only used in 3D and there should be 3 constraints per contact
* with friction, and we should never end up in this routine,
* but we check nevertheless for safety ...
        II = L%LDATA(2) ! 1, 2 or 3 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (31) ! PINB
        II = L%LDATA(1) ! 1 = tangent, 0 = normal w/o fric, -1 = normal w. fr.
        IF (II /= 0) GO TO 998 ! Raise error and stop
      CASE (76) ! GPIN
        II = L%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      CASE (99) ! SLID
        II = L%LDATA(2) ! > 0 for a tangent constraint (see SOLVE_GROUP)
        IF (II > 0) GO TO 998  ! Raise error and stop
      END SELECT
!fc 20/05/203
!fc
      LD1 = L1%NDLIE
      LD2 = L2%NDLIE
 . . .
  998 WRITE (BLABLA, 1005)
      CALL ERRMSS ('SOLVE_SINGLE_LIAISON', BLABLA)
 1005 FORMAT('CANT TREAT CONTACT WITH FRICTION')
      CALL EPX_STOP('SOLVE_SINGLE_LIAISON')
 . . .
      END SUBROUTINE SOLVE_SINGLE_LIAISON
```

4.5.3 Corrections in subroutine SOLVE_GROUP

The corrections presented so far in subroutines SOLVE_SINGLE_LIAISON and SOLVE_TWO_LIAISONS are precautionary ones, but will have no influence on the malfunctionings described in Section 3. In fact, according to Table 1 all the failing tests generated a single group containing four constraints, which therefore is treated in SOLVE_GROUP.

By inspecting SOLVE_GROUP, the reason for the malfunctionings was detected in the part concerning the treatment of friction in the PINB model, and was related (as it could have been suspected) to the renumbering of the links. In fact, always from Table 1, it may be noted that the tests using `NORE` to disable the renumbering worked correctly. The correction implemented is shown in red below.

```
      SUBROUTINE SOLVE_GROUP . . .
 . . .
      INTEGER :: IR, IR1, . . .
 . . .
      IF (DO_PINB_FROT) THEN
*
* prise en compte du frottement dans les contacts par pinball
*
* attention: en cas de modif dans cet if, modifier aussi la recherche de
* do_pinb_frot plus haut (optimisation cpu)
*
        DO ILIES = 1, NLIE
          L => LL_ARRAY(ILIES)%LIAISON
          ITYP = L%LTYPE
          SELECT CASE (ITYP)
          CASE (31) ! PINB
            II = L%LDATA(1) !  1 if this is a tangent link (friction-related).
                            !    (the companion normal link is the next one).
                            !  0 if this is a normal link without friction
                            !    (there is no companion link)
                            ! -1 if this is a friction-related normal link.
                            !    (companion tangent link is the previous one)
*
            IF (II == -1) THEN   ! Link ilies is normal link with friction
              IF (ILIES <= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                     'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
              ENDIF
              ILIES1 = ILIES - 1 ! Companion tangent link
              L2 => LL_ARRAY(ILIES1)%LIAISON
              ITYP1 = L2%LTYPE
              IF (ITYP1 /= 31) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                     'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
              ENDIF
              IF (L2%LDATA(1) /= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                     'NO TANGENT PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - NO TANGENT PINB LINK')
              ENDIF
*
              MU = L2%RDATA(4)
!fc 19/05/2023
!fc Attention! In order to access SMLAG, one must take into account LRENUM!
              IR = LS%LRENUM(ILIES)
              IR1 = LS%LRENUM(ILIES1)
!fc           FTNOR = ABS (LS%SMLAG(ILIES))
!fc           FTTAN = ABS (LS%SMLAG(ILIES1))
              FTNOR = ABS (LS%SMLAG(IR))
              FTTAN = ABS (LS%SMLAG(IR1))
!fc
              REDU = MU*FTNOR / FTTAN
              IF (REDU < 1.DO) THEN
!fc             LS%SMLAG(ILIES1) = REDU*LS%SMLAG(ILIES1)
                LS%SMLAG(IR1) = REDU*LS%SMLAG(IR1)
!fc 19/05/2023
              ENDIF
*
            ENDIF
*
          CASE DEFAULT
          END SELECT
        END DO
      ENDIF
 . . .
      END SUBROUTINE SOLVE_GROUP
```

The error concerned the fact that the organization of the SMLAG array (representing the normalised RHS of the links) is different (through the renumbering array LRENUM) from that of the SM array (which represents the RHS of the links). Both such arrays, as well as the LRENUM array, are contained in the module M_LINK_SOLVER_DATA and the description of such arrays in the comments did not point out this fundamental difference.

Only when renumbering is disabled (`NORE` optional keyword) do the two organizations coincide. This explains why tests with `NORE` worked correctly in Section 3.

4.5.4 Corrections in module M_LINK_SOLVER_DATA

In the description of the SMLAG and SM arrays contained in this module, a comment is added warning about the fact that the organization of these two arrays is different (through the use of LRENUM):

```
      MODULE M_LINK_SOLVER_DATA
 . . .
      TYPE LINK_SOLVER
        INTEGER :: NLINKS              ! NUMBER OF LINKS IN THE SYSTEM
        INTEGER(8), POINTER :: ILIAIS(:)  ! LOWER PROFILE OF LINKS MATRIX
        INTEGER, POINTER :: LRENUM(:)  ! RENUMBERING OF LINKS MATRIX
        INTEGER, POINTER :: NBPLI(:,:) ! N OF DOFS ; ADDRESS IN LPAIR
        REAL(8), POINTER :: ANORM(:)   ! NORM OF THE LINKS
        REAL(8), POINTER :: SM(:)      ! RIGHT-HAND SIDE OF THE LINKS
        REAL(8), POINTER :: SMLAG(:)   ! NORMALISED RHS OF THE LINKS
!fc Attention!
!fc The SM and SMLAG arrays are organized differently!!!
!fc One is in the natural order, while the other uses the RENUmbered
!fc order of the constraints (if RENU is active).
!fc Use LRENUM to access the correct item from these vectors
!fc as appropriate (see examples in the code).
!fc Attention!
        INTEGER, POINTER :: LPAIR(:)   ! LINKED DOFS
        REAL(8), POINTER :: COEFL(:)   ! COEFFICIENTS
        REAL(8), POINTER :: CGAMMA(:)  ! COEFFICIENTS * GAMMA
        REAL(8), POINTER :: BL(:)      ! B MATRIX (JUST LOWER PROFILE)
        LOGICAL :: NEEDS_RENUMBERING,  ! IS RENUMBERING NEEDED?
     &             DO_PREMIER_CALCUL,
     &             DO_ECR_ET_SIG
!--      Structure LMATRIX  pour TSOLVER=0, 1 ou 3
        TYPE (LMATRIX), POINTER :: MATRIX
!
        REAL(8)  :: DTSUM ! SAUVEGARDE DU DTSUM AU CAS ON NE RECALCULE
                          ! PAS BL
        REAL(8), POINTER :: ANORM_SAV(:) ! SAUVEGARDE AUSSI DU LA NORME
                                         ! DES LIAISONS
      END TYPE LINK_SOLVER
 . . .
      END MODULE M_LINK_SOLVER_DATA
```

5 Test results after the corrections

After the corrections described in Section 4.5, all the tests of Section 3 are re-run in order to verify the results. All tests succeed and pass the qualifications, as summarized in Table 2.

Test	Mesh	Impact	Fric.	Split	Renu.	Contact @ step	Groups	Links	Solution
C2EL	2 Q42L	Normal	YES	`NONE`	(`RENU`)	18	1	2	Correct
C2EL_B	2 Q42L	Oblique	YES	`NONE`	(`RENU`)	11	1	4	Correct
C2EL2	2 Q42L	Normal	YES	(`DOF`)	(`RENU`)	18	1	4	Correct
C2EL_B2	2 Q42L	Oblique	YES	(`DOF`)	(`RENU`)	11	1	4	Correct
C2EL03	2 Q42L	Normal	YES	`NONE`	`NORE`	18	1	2	Correct
C2EL04	2 Q42L	Oblique	YES	`NONE`	`NORE`	11	1	4	Correct
C2EL05	2 Q42L	Normal	NO	`NONE`	`NORE`	18	1	2	Correct
C2EL06	2 Q42L	Oblique	NO	`NONE`	`NORE`	11	1	2	Correct

Table 2: Simulations performed with the corrected version of EPX.

All four tests involving the normal impact now give identical results, as it could be expected, since the friction does not play any role in normal impact.

Of the four tests involving oblique impacts, the first three (which include the friction) give identical results, while the fourth one, which does not include the friction, gives a slightly different result. This difference was expected, since the friction plays a role in oblique impact.

6 Developments and upgrade of the model

This Section describes the further developments that have been applied to the PINB contact model, after the corrections outlined in the previous Sections 4.5 and 5, in order to upgrade the model to the same level as the other contact models in EPX as far as the treatment of friction is concerned, namely using the two-tangent formulation.

From the corrections described in Section 4.5 it has emerged that a crucial point of the links implementation is the renumbering of the links prior to solution of the system. Therefore, we start by analyzing the renumbering scheme.

6.1 Renumbering of the links

In the following discussion, we limit ourselves to the LM-based links model of constraints (`LINK COUP` directive). The older `LIAI` model is not considered.

Links renumbering has the scope of reducing the bandwidth of the links matrix and hopefully speeding up its solution. Therefore, renumbering should only be taken into account in the final part of the links treatment of each time step, the one which deals with the solution of the linear system. All the manipulation of the links done prior to the solution phase do not need to take the renumbering into account, and are performed on the links in their natural order.

Renumbering is only applied when the Choleski solver, which is the default one in EPX, is adopted, and only provided the user has not explicitly disabled it by specifying the `NORE` optional keyword.

By inspecting the code, it is found that links renumbering is computed in subroutine RENUMBER_0 from module M_LINK_SOLVER, which is called from subroutine SOLVE_GROUP of module M_LINKS. The subroutine RENUMBER_0 calls RENUMBER_1 and this in turn calls RENUMBER_2.

The scope is to fill in the LRENUM array of the LS link solver (LS%LRENUM). This array is allocated (if necessary) in subroutine SOLVE_GROUP of module M_LINKS and has a length of 2*NLIE, where NLIE is the number of links in the group under consideration. It is then immediately initialized in such a way that it contains the integers in natural order:

- LRENUM(I) = I, for I = 1, NLIE
- LRENUM(NLIE+I) = I, for I = 1, NLIE

6.1.1 Example of renumbering

By taking as an example the case C2EL2 of Section 3.3, there are four links in the (only) group (NLIE = 4), therefore LRENUM is allocated with a length of 8 and initialized as follows:

- LRENUM(1:8) = 1, 2, 3, 4, 1, 2, 3, 4.

Then, the subroutine RENUMBER_0 is called, which in turn calls RENUMBER_1 by passing to it the second "slot" of the LRENUM array (LRENUM(NLIE+1)). Upon return from RENUMBER_1, we find:

- LRENUM(1:8) = 1, 2, 3, 4, 4, 3, 2, 1.

i.e., the second slot of the array now contains the renumbered links order. Immediately after that, before returning control, RENUMBER_0 redefines the first slot of LRENUM by the following code:

```
DO I = 1, NLIE
  K = LRENUM(I+NLIE)
  LRENUM(K) = I
END DO
```

Therefore, upon returning to SOLVE_GROUP, the LRENUM array ends up containing:

- LRENUM(1:8) = 4, 3, 2, 1, 4, 3, 2, 1.

6.1.2 Replacing the second slot pf LRENUM by a local buffer

Code inspection reveals that the second slot of LRENUM is not used any longer. Therefore, this seems to be just a buffer (of length NLIE) which is used by RENUMBER_1 in order to compute the renumbering. In order to verify this hypothesis, we decide to tentatively dimension the length of

LRENUM to just NLIE (instead of 2*NLIE) and to allocate a local buffer in RENUMBER_0 to be passed to RENUMBER_1.

Note that this modification concerns not only module M_LINKS (subroutine SOLVE_GROUP) but also the module M_INTERFACE, as regards the domain decomposition case. For that case, the subroutine RNUM0 is modified similarly to RENUMBER_0 by replacing the second slot of LRENUM by a local buffer.

Then, the whole set of benchmarks using LINK is run in order to check that there are no failing tests and that the results are identical. Since this is indeed the case, the modification is evolved.

6.1.3 Meaning of LRENUM

The *natural order* of the links is simply the progression of natural numbers: 1, 2, and so on until NLIE, which is the number of links in the group being currently treated. We may indicate this as NAT(I) = I, with I = 1, ..., NLIE.

The *renumbered order* of the links can be any permutation of the natural order that is found convenient for the solution of the system (according to some undocumented algorithm), that we will denote as REN(I), I = 1, ..., NLIE.

For example, in the case C2EL2 of Section 3.3 there are four links, so that NAT = 1, 2, 3, 4. In the same test the renumbering algorithm finds that, to speed up the solution of the system, the links should be treated (during the solution) in a different order, which in that particular case is simply the inverse order: REN = 4, 3, 2, 1.

If no renumbering is applied, then of course it is simply REN(I) = NAT(I), I = 1, ..., NLIE. Therefore, REN(I) can safely be used in the programming where appropriate, without checking whether there actually is or not renumbering, provided REN(I) has been initialized as NAT(I).

The motivation behind the building and use of the LRENUM array (REN in the above notation) is the fact that some arrays used in the links model have their contents arranged according to the natural order of the links, i.e. the order in which the links appear in the list LL_ARRAY, while others (namely some of those related to the solution of the linear problem) are arranged according to the renumbered order. For example, the SM array contains the RHS of the links in the natural order, while SMLAG contains the normalised RHS of the links in the renumbered order.

6.1.4 Description of LRENUM

The meaning of LRENUM is described in various comments disseminated around the code, but the descriptions appear somewhat confusing (and even imprecise). For example, in the subroutine FRCONT which is part of the old LIAI model of constraints, we find:

```
*     renumerotation des liaisons :
*      ilies : ordre des renumerotation : smlag, sm(2,:)
*      ir    : ordre naturel des liaisons : nplie,nbpli,sm(1,*)
*                                         sauf  sm(2,:)
```

The choice of the names is a bit unfortunate, since one would perhaps expect IR to indicate the renumbered index and ILIES the natural one, while it is exactly the opposite.

As concerns the list of arrays following each organization, it should be noted that the SM array has only one slot (of length NLIE) in the links model. Therefore, `sm(1,*)` should be read `sm`, and `sm(2,:)` does not exist. Furthermore, yet another array which follows the renumberered order like SMLAG but has not been mentioned in the comments above is the ANORM vector, which contains the norms of the (RHS of the) links.

Another description is found in subroutine RENUMBER_0 of module M_LINK_SOLVER:

```
*     tableau <lrenum> : ir    : ordre naturel des liaisons
*                        ilies : ordre apres renumerotation
*
*     nous avons :     ir    = lrenum(ilies)
*                      ilies = lrenum(ir+maxlie)
```

In the second part of the comment, the line IR = LRENUM(ILIES) is correct and would correspond to IR = REN(I) in the notation introduced in Section 6.1.3.

However, the following comment line ILIES = LRENUM(IR+MAXLIE) is wrong, for various reasons. The first reason, which is not so important, is that NLIE (the actual number of links) should be used instead of MAXLIE (maximum number of liaisons), which is a relic from the LIAI model of constraints.

The second and more fundamental reason is that the second slot of the LRENUM array, even before being eliminated as described in Section 6.1.2, did *not* contain the inverse of the renumbered order, contrary to what the comment seems to imply. This can be verified by looking at the example of Section 6.1.1 from which it appears that the second slot of LRENUM actually contained a copy of the first slot, i.e. of the (direct) renumbered order REN.

Having access to the inverse of the renumbered order, i.e. being able to directly compute NAT(I) from REN(I) in the above notation, might perhaps be useful in some manipulations. Indeed, in a relatively recent re-programmation of the LIAI model (performed after the introduction of the LINK model), cfr. module M_OLIAISONS_RESOU, we find *two* renumbering arrays, LRNUM1 and LRNUM2. It is believed that LRNUM1 corresponds to LRENUM of the links model (direct renumbering) while LRNUM2 corresponds to its inverse (inverse renumbering). The LRNUM2 array is built when calling the RNUM0 routine, but its contents has not been verified in a real example.

6.1.5 Use of LRENUM

Typically, LRENUM is used when performing loops over all the links. For example, when solving a group of links in SOLVE_GROUP of module M_LINKS we find various types of loops over the links.

Direct loop

The first type of loop is performed in the *natural order* of the links, i.e. in the order in which they are stored in the array LL_ARRAY, which presumably coincides with the order in which they have been imposed (i.e. added to the set of links, before possibly splitting this set into a number of disjoint groups):

```
*
* build up lpair, coefl, nbpli, sm
*
 . . . (skip)
        DO ILIES = 1, NLIE
          L => LL_ARRAY(ILIES)%LIAISON
          NDL = L%NDLIE
          LS%NBPLI(1,ILIES) = NDL
          LS%NBPLI(2,ILIES+1) = LS%NBPLI(2,ILIES) + NDL
          IF (.NOT. ALLOCATED(COMPACT_REMOTE_DOFS)) THEN ! STANDARD CASE
             DO I = 1, NDL
               K = K + 1
               LS%LPAIR(K) =  L%DLIE(I)
               LS%COEFL(K) =  L%COEF(I)
             END DO
          ELSE ! CAS PARTICULIER: GPCG & ADAP
             DO I = 1, NDL
               K = K + 1
               IF (L%DLIE(I) <= NLIBT) THEN
                  LS%LPAIR(K) = L%DLIE(I) ! LOCAL DOF
               ELSE
                  II=L%DLIE(I) - DOMAIN(1)%NLIBI ! ACTUAL REMOTE DOF NUMBER
                  LS%LPAIR(K) = NLIB_ACTIVE + COMPACT_REMOTE_DOFS(2,II) ! COMPACTED NUMBER
               ENDIF
               LS%COEFL(K) =  L%COEF(I)
             END DO
          ENDIF
          LS%SM(ILIES) = L%BTERM
        END DO
```

In this type of loop, the loop index ILIES represents the *natural index* of each link, which goes from 1 to NLIE, the number of links (in this group). As already noted, this is in contrast with the comments discussed in the previous Section, where it is said that ILIES should be used to represent the renumbered order. This might cause some confusion to the code reader or the programmer.

The current link pointer L in the loop is set to the ILIES-th link, i.e. the links are traversed (and used) in their natural order. In fact, all the arrays used or modified within the loop, namely LPAIR, COEFL and SM in the above example, are organized in the natural order of the links. This is in agreement with the comments in FRCONT mentioned above. However, according to those comments, the loop index to be used in the above sample code should be IR and not ILIES.

Indirect loop

The second type of loop is, for example:

```
*
* premier calcul des forces normales aux noeuds esclaves
 . . . (skip)
*
        DO ILIES = 1, NLIE
          IR = LS%LRENUM(ILIES)
          L => LL_ARRAY(IR)%LIAISON
          ITYP = L%LTYPE
          SELECT CASE (ITYP)
          CASE (19:20) ! GLIS, IMPA
            II = L%LDATA(1) ! 1er ddl du noeud esclave (voir CALC_BCOEF3D_LINK)
            IAUX = L%NDLIE / IDIM ! Ceci parait le n. de noeuds lies (L%NPLIE)
            ISLAVE = 0
            DO I=1,IAUX
* ISLAVE ici est le rang du noeud esclave dans L%PLIE (liste des noeuds lies)
               IF (L%DLIE(IDIM*(I-1)+1) .EQ. II) THEN
                 ISLAVE = I
                 EXIT
               ENDIF
            ENDDO
            IF (ISLAVE == 0) THEN ! On n'a pas trouve islave ...
              CALL ERRMSS ('SOLVE_GROUP','ISLAVE WAS NOT FOUND')
              CALL EPX_STOP ('SOLVE_GROUP - ISLAVE WAS NOT FOUND')
            ENDIF
*
            VNORM(1) = L%RDATA(1)
            VNORM(2) = L%RDATA(2)
            VNORM(3) = L%RDATA(3)
            CALL LINK_FRNORM(ISLAVE, LS%NBPLI, LS%CGAMMA, LS%SMLAG,
     &                      LS%ANORM, FNORM, ILIES, IR, ITYP,
     &                      VNORM) ! Returns FNORM
            L%RDATA(5) = FNORM ! On sauve la force de liaison calculee
 . . . (skip)
          END SELECT
        END DO
```

Again, ILIES is used as the loop index (natural order), in contrast with what stated in the comments. However, in this loop, unlike the previous one, the links should be treated in the renumbered order. Therefore, the renumbered index IR (which should rather be named ILIES according to the comments) is computed via `IR = LS%LRENUM(ILIES)` and then the current link pointer L in the loop is set to the IR-th link, i.e. the links are traversed (and used) in their renumbered order. In this particular case, no global array arranged according to IR is used directly in the loop, but note that both ILIES and IR are passed to the subroutine LINK_FRNORM, which will use one or the other depending on the array it manipulates.

Again, in order to conform to the comments, the names ILIES and IR should be swapped in the above loop.

Mixed-type loop

Finally, an example of a "mixed-type" of loop is:

```
      DO ILIES = 1, NLIE
        IR = LS%LRENUM(ILIES)
        LS%SMLAG(ILIES) = LS%SM(IR)
* recherche de do_premier_calcul et de do_ecr_et_sig
        L => LL_ARRAY(ILIES)%LIAISON
 . . . (skip)
      END DO
```

This loop traverses the links list in the natural order like in the first type of loop, since L points to the ILIES-th link (`L => LL_ARRAY(ILIES)%LIAISON`). However, the renumbered index IR is also built from ILIES (through LRENUM) within the loop, and is then used to construct the SMLAG array from the SM array.

Two alternative forms of the same loop

We present here two alternative, but equally valid, forms of writing the same loop. The loop considered is the one presented (and corrected) in Section 4.5.3, i.e. the code which accounts for friction in the pinball contacts in subroutine SOLVE_GROUP.

The two versions are shown side-by-side below to facilitate comparison. In the left column we have the version described in Section 4.5.3. This version does not conform to the style of the loops presented in the previous examples. In fact, we see the use of LS%SMLAG(IR) instead of LS%SMLAG(ILIES), which is the notation used in the rest of the subroutine SOLVE_GROUP. This is atypical, and in fact it had led to the error which was eventually corrected in Section 4.5.3 (see commented out lines).

In the right column, the same loop is re-written conforming to the style used in the rest of the subroutine SOLVE_GROUP. Both versions work correctly, but the second one should be preferred for uniformity with the rest of the subroutine.

Note that the left-column version contains an extra correction with respect to the one described in Section 4.5.3, and the same correction is also applied to the right-column version. The correction concerns the case when FTTAN (tangential contact force) vanishes in a case where friction has been activated. This may occur for example if the impact (with friction) is perfectly normal.

In that case, computing the reduction coefficient REDU would cause a division by zero, resulting into a (positive) infinite value for REDU. Then, the following test on REDU less than 1.0 would return .FALSE. and the tangential component of SMLAG would not be modified, thus leading to the correct numerical result (so the problem is relatively harmless and had not been noticed so far).

However, one can note that in the case of vanishing FTTAN there is no need to correct the tangential component of SMLAG in the first place, so the calculation of REDU and the test on it may simply be skipped, thus also avoiding the division by zero.

The right-column version of the loop has been retained and evolved in the code. Note that this version requires the inverse renumbering when computing ILIES1 from IR1. The problem is provisionally solved here by resorting to the ILYEST subroutine. However, this routine performs a straight search and will be inefficient if the links group contains many links (large NLIE).

To improve the efficiency, an inverse renumbering array (LRNUM2) should be added to the code and should be built from LRNUM (e.g. by using the ICLASS routine available in EPX) as soon as this has been computed. However, such optimization is not applied since, in any case, the LM-based pinballs friction model is going to be completely re-implemented next by using the two-tangent version, similarly to the other contact models of EPX, as shown in Section 6.2.

```
      IF (DO_PINB_FROT) THEN
*
* prise en compte du frottement dans les contacts par pinball
*
* attention: en cas de modif dans cet if, modifier aussi la recherche de
* do_pinb_frot plus haut (optimisation cpu)
*
        DO ILIES = 1, NLIE
          L => LL_ARRAY(ILIES)%LIAISON
          ITYP = L%LTYPE
          SELECT CASE (ITYP)
          CASE (31) ! PINB
            II = L%LDATA(1) !  1 if this is a tangent link (friction-related)
                            !    (the companion normal link is the next one).
                            !  0 if this is a normal link without friction
                            !    (there is no companion link)
                            ! -1 if this is a friction-related normal link.
                            !    (companion tangent link is the previous one)
*
            IF (II == -1) THEN   ! Link ilies is normal link with friction
              IF (ILIES <= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
              ENDIF
              ILIES1 = ILIES - 1 ! Companion tangent link
              L2 => LL_ARRAY(ILIES1)%LIAISON
              ITYP1 = L2%LTYPE
              IF (ITYP1 /= 31) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
```

```
              ENDIF
              IF (L2%LDATA(1) /= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'NO TANGENT PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - NO TANGENT PINB LINK')
              ENDIF
*
              MU = L2%RDATA(4)
!fc 19/05/2023
!fc Attention! In order to access SMLAG, one must take into account LRENUM!
              IR = LS%LRENUM(ILIES)
              IR1 = LS%LRENUM(ILIES1)
!fc           FTNOR = ABS (LS%SMLAG(ILIES))
!fc           FTTAN = ABS (LS%SMLAG(ILIES1))
              FTNOR = ABS (LS%SMLAG(IR))
              FTTAN = ABS (LS%SMLAG(IR1))
!fc
! protect against case that FTTAN=0 despite the fact that there is friction
! (such as in perfectly normal impact with friction)
! to avoid division by zero.
! if FTTAN=0, there is no need to correct the tangent component anyway,
! so no need to compute REDU
              IF (FTTAN > 0.D0) THEN
                REDU = MU*FTNOR / FTTAN
                IF (REDU < 1.D0) THEN
!fc                LS%SMLAG(ILIES1) = REDU*LS%SMLAG(ILIES1)
                  LS%SMLAG(IR1) = REDU*LS%SMLAG(IR1)
!fc 19/05/2023
                ENDIF
              ENDIF
*
            ENDIF
```

```
*
          CASE DEFAULT
          END SELECT
        END DO
      ENDIF




      IF (DO_PINB_FROT) THEN
*
* prise en compte du frottement dans les contacts par pinball
*
* attention: en cas de modif dans cet if, modifier aussi la recherche de
* do_pinb_frot plus haut (optimisation cpu)
*
!fc 25/05/2023 the following loop is re-written in the standard way,
!fc i.e. using ILIES as the loop index, which corresponds to the
!fc renumbered index (index after renumbering), and IR=LRENUM(ILIES) as
!fc the natural index.
!fc 25/05/2023
        DO ILIES = 1, NLIE ! Loop index = index after renumbering
          IR = LS%LRENUM(ILIES) ! Natural index of the link
          L => LL_ARRAY(IR)%LIAISON
          ITYP = L%LTYPE
          SELECT CASE (ITYP)
          CASE (31) ! PINB
            II = L%LDATA(1) !  1 if this is a tangent link (friction-related)
                            !    (the companion normal link is the next one).
                            !  0 if this is a normal link without friction
                            !    (there is no companion link)
                            ! -1 if this is a friction-related normal link.
                            !    (companion tangent link is the previous one).
*
            IF (II == -1) THEN   ! Link ilies is normal link with friction
              IF (IR <= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
              ENDIF
              IR1 = IR - 1 ! Companion tangent link (natural index)
              L2 => LL_ARRAY(IR1)%LIAISON
              ITYP1 = L2%LTYPE
              IF (ITYP1 /= 31) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'WRONG PREVIOUS PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - WRONG PREVIOUS PINB LINK')
              ENDIF
              IF (L2%LDATA(1) /= 1) THEN
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'NO TANGENT PINB LINK')
                CALL EPX_STOP('SOLVE_GROUP - NO TANGENT PINB LINK')
              ENDIF
*
              MU = L2%RDATA(4)
! ilies1 is such that IR1 = LS%LRENUM(ILIES1) (inverse of the renumbering)
! provisionally use ILYEST (inefficient ...)
              CALL ILYEST (IR1 , NLIE, LS%LRENUM, ILIES1)
              IF (ILIES1 == 0) THEN ! ILIES1 is the index after renumbering
                                    ! of the tangent link
                CALL ERRMSS ('SOLVE_GROUP',
     >                       'CANT FIND ILIES1')
                CALL EPX_STOP('SOLVE_GROUP - CANT FIND ILIES1')
              ENDIF
              FTNOR = ABS (LS%SMLAG(ILIES))  ! SMLAG is organized according
              FTTAN = ABS (LS%SMLAG(ILIES1)) ! to the renumbered order
! protect against case that FTTAN=0 despite the fact that there is friction
! (such as in perfectly normal impact with friction)
! to avoid division by zero.
! if FTTAN=0, there is no need to correct the tangent component anyway,
! so no need to compute REDU
              IF (FTTAN > 0.D0) THEN
                REDU = MU*FTNOR / FTTAN
                IF (REDU < 1.D0) THEN
                  LS%SMLAG(ILIES1) = REDU*LS%SMLAG(ILIES1)
                ENDIF
              ENDIF
*
            ENDIF
*
          CASE DEFAULT
          END SELECT
        END DO
      ENDIF
```

6.2 Upgrade of the friction algorithm for PINB

We now describe the formulation and implementation of the two-tangent version of the friction algorithm for the PINB contact model, which replaces the single-tangent version used before the present work. The two-tangent version is inspired by the two-tangent algorithms used by the other contact models, namely GLIS/IMPA, GPIN and SLID.

6.2.1 General considerations

The modifications applied to the PINB method implementation do not affect only the case with friction, but also the case without friction, due to the complexity of the contact model. They are mostly carried on in the subroutine SOLVE_GROUP of module M_LINKS. The main change is due to the fact that contact forces are now provisionally computed in a new subroutine LINK_FRNORM_PINB and then finally evaluated in another new subroutine LINK_FRCONT_PINB, while previously the pre-calculations for PINB were done directly in SOLVE_GROUP (in a loop executed under the DO_PINB_FROT condition, i.e. in the presence of friction) and the final calculation in FLINKS, like for non-contact links.

The two new subroutines are inspired from the similarly-named ones of the other contact models (LINK_FRNORM and LINK_FRCONT for GLIS/IMPA, LINK_FRNORM_GPIN and LINK_FRCONT_GPIN for GPIN, SLID_FRNORM and SLID_FRCONT for SLID). The two new subroutines are described in detail below.

Some modifications are applied also to the subroutine SOLVE_SINGLE_LIAISON, but these are only minor, due to the fact that in cases with friction there are always at least two links (one along the normal and one along the tangent) concerning the same dofs, so that they should form a group of (at least two) links and one should never enter in this subroutine.

The subroutine SOLVE_TWO_LIAISONS is not modified, other than by adding some protections. This subroutine is only used when the `SOL2` optional keyword is specified, and a group of only two links has to be solved. The routine is not equipped with the treatment of contact friction (to keep it relatively simple), which is only implemented in the more general routine SOLVE_GROUP. Therefore, if by chance either of the two links is a tangent contact link, an error message is issued and the simulation is stopped. In such a case, removing the SOL2 keyword from the input should fix the problem.

6.2.2 Rebound-related options

Since the modifications have to cope also with the contact rebound model of PINB (as it is the case also for GPIN), we shortly recall some key information relative to such model.

The following excerpt from M_PINBALL and M_GPINBALL summarizes the rebound-related options for the PINB (left column) and GPIN (right column) contact models. Note that the default value for PINBALL_REBO is 0, while for GPINBALL_REBO it is 1, but in both cases the standard a-priori rebound detection algorithm is meant to be used.

```
INTEGER :: PINBALL_REBO  ! 0 : (DEFAULT) USE A PRIORI PINBALL
                         !     REBOUND DETECTION ALGORITHM
                         !     (OPTI PINS REB1 = DEFAULT)
                         ! 1 : USE A POSTERIORI PINBALL
                         !     REBOUND DETECTION ALGORITHM
                         !     (OPTI PINS REB2)
                         ! 2 : USE NO PINBALL REBOUND DETECTION
                         !     ALGORITHM (ONLY FOR TESTING!)
                         !     (OPTI PINS NORB)
```

```
INTEGER :: GPINBALL_REBO  ! 0 : USE SIMPLIFIED A PRIORI
                          !     REBOUND DETECTION ALGORITHM
                          !     COMPUTE PDOT USING ONLY V
                          !     (OPTI GPNS REBO)
                          ! 1 : (DEFAULT) USE PINB-LIKE A PRIORI
                          !     GPINBALL REBOUND DETECTION ALGORITHM
                          !     COMPUTE PDOT USING V AND A
                          !     (OPTI GPNS REB1 = DEFAULT)
                          ! 2 : USE A POSTERIORI GPINBALL
                          !     REBOUND DETECTION ALGORITHM
                          !     (OPTI GPNS REB2)
                          ! 3 : USE NO GPINBALL REBOUND DETECTION
                          !     ALGORITHM (ONLY FOR TESTING!)
                          !     (OPTI GPNS NORB)
INTEGER :: GPINBALL_REBC  ! IT ONLY AFFECTS GPINBALL_REBO == 0 OR 1:
                          ! 0 : (DEFAULT) USE GENUINE A PRIORI REBOUND
                          !     ALGORITHM
                          ! 1 : USE COMBINED PRE-POST REBOUND ALGORITHM
```

6.2.3 The two-tangent version of the friction algorithm for PINB

The upgraded, two-tangent version of the friction algorithm for the PINB contact model, which is listed hereafter for completeness, is basically identical to that used by the other contact models, in particular that of GLIS described in Section 3.1.2 of reference [37]. However, it also presents some particularities, which will be highlighted during the description.

The first particularity concerns the rebound. In GLIS, the rebound is always checked *a posteriori*. In the PINB model, rebound is checked *a priori* by default. In that case, if rebound is preliminarly detected, the contact is discarded before being treated (no associated links are imposed). Therefore, only the case of *a posteriori* rebound is treated in the contact algorithm described below. Another particularity is that the PINB model is available both in 3D and in 2D, while GLIS is only available in 3D.

By considering first the updated PINB model in the general 3D case, a local orthonormal reference frame is built for every (non a-priori rebounding) contact, with one axis $\hat{\boldsymbol{n}}$ directed along the contact normal, like in the original model. Then, instead of just one tangent axis $\hat{\boldsymbol{\tau}}$, *two* tangent axes $\hat{\boldsymbol{\tau}}_1$ and $\hat{\boldsymbol{\tau}}_2$ are constructed, which define the *tangent plane* τ (perpendicular to $\hat{\boldsymbol{n}}$). The choice of these two particular tangent axes is irrelevant, as long as they form an orthonormal basis together with $\hat{\boldsymbol{n}}$. The solution will not depend upon the particular axes chosen. Then in the case of friction, instead of considering just one tangential link, we consider *two* tangential links, along $\hat{\boldsymbol{\tau}}_1$ and $\hat{\boldsymbol{\tau}}_2$.

Another less important modification with respect to the original version of the algorithm is that the two tangential links are written (i.e., added to the set of links) *before* and not after the corresponding normal link in the list of imposed links. Thus the constraints become:

$$\boldsymbol{v}_s^{n+3/2} \cdot \hat{\boldsymbol{\tau}}_1{}^{n+1} = \boldsymbol{v}_d^{n+3/2} \cdot \hat{\boldsymbol{\tau}}_1{}^{n+1} = \left(\sum_{i=1}^{M} c_i \boldsymbol{v}_{m_i}^{n+3/2} \right) \cdot \hat{\boldsymbol{\tau}}_1{}^{n+1} \qquad \text{First tangential} \tag{11}$$

$$\boldsymbol{v}_s^{n+3/2} \cdot \hat{\boldsymbol{\tau}}_2{}^{n+1} = \boldsymbol{v}_d^{n+3/2} \cdot \hat{\boldsymbol{\tau}}_2{}^{n+1} = \left(\sum_{i=1}^{M} c_i \boldsymbol{v}_{m_i}^{n+3/2} \right) \cdot \hat{\boldsymbol{\tau}}_2{}^{n+1} \qquad \text{Second tangential} \tag{12}$$

$$\boldsymbol{v}_s^{n+3/2} \cdot \hat{\boldsymbol{n}}^{n+1} = \boldsymbol{v}_d^{n+3/2} \cdot \hat{\boldsymbol{n}}^{n+1} = \left(\sum_{i=1}^{M} c_i \boldsymbol{v}_{m_i}^{n+3/2} \right) \cdot \hat{\boldsymbol{n}}^{n+1} \qquad \text{Normal} \tag{13}$$

This results in the following modified algorithm (in 3D). In the case with friction, for each detected contact (penetration) we have *three* constraints: two (provisional) constraints J_1 and J_2 along the tangents $\hat{\boldsymbol{\tau}}_1$ and $\hat{\boldsymbol{\tau}}_2$, associated with Lagrange multipliers λ_{J_1} and λ_{J_2}, respectively, followed by a constraint I along the normal $\hat{\boldsymbol{n}}$, of Lagrange multiplier λ_I. In the implementation, it is $I = J_1 + 2 = J_2 + 1$, that is, in the list of links the first tangent constraint is directly followed by the second tangent constraint, which in turn is followed directly by the corresponding normal constraint.

The system of links is solved by the Lagrange multipliers method, thus obtaining the three Lagrange multipliers λ_{J_1}, λ_{J_2} and λ_I. The reactions are *provisionally* computed by:

$$\boldsymbol{R} = \boldsymbol{C}^T \boldsymbol{\lambda} \tag{14}$$

where $\boldsymbol{C}$ is the matrix of constraint coefficients. At this point, in the other contact models (GLIS/IMPA, GPIN and SLID), the reaction $\boldsymbol{R}_s = (R_{sn}, R_{s\tau_1}, R_{s\tau_2})$ at the slave node (ISLAVE) is evaluated directly. However, in the case of the PINB model, there is no slave node and, in general, the reactions act over all nodes of the elements containing the two pinballs A and B involved in the contact.

Therefore, we arbitrarily select the first contacting pinball (denoted the A pinball) as the "slave" pinball, only as far as the present rebound and friction calculations are concerned, and compute $\boldsymbol{R}_s = \boldsymbol{R}_A$ by assembling the reactions at all the n_A nodes i_A of the element containing this pinball, obtaining $\boldsymbol{R}_s = \boldsymbol{R}_A = \sum_{i_A=1}^{n_A} \boldsymbol{R}_{i_A}$ (see details in Section 6.2.4). Next:

1. If rebound should be checked *a posteriori* (`REB2` option), we check the normal reaction R_{sn} for rebound. If $R_{sn} < 0$, then there would be traction along the normal, i.e. a posteriori rebound is occurring. We set $\lambda_I = 0$ and $\lambda_{J_1} = \lambda_{J_2} = 0$ (if present), thus eliminating *all three* constraints from the system, and we re-compute the reactions by (14). The reactions concerning the contact under consideration will vanish out because the corresponding multipliers are zero.

2. Else, $R_{sn} \geq 0$ and contact (not a posteriori rebound) is actually occurring:

   - If no friction is involved in this contact (so that neither J_1 nor J_2 constraint is present), then the computed contact reactions (along the normal) for all nodes of both pinballs are already correct and the calculation is complete.
   - Otherwise, there is friction. The reactions along the normal are already the final ones, but those in the tangent plane need to be corrected if they do not satisfy Coulomb's condition. Let R_{sn} denote the (final) normal reaction and $R_{s\tau_1}$, $R_{s\tau_2}$ the (provisional) tangential reactions at the "slave" pinball $s \equiv A$ resulting, respectively, from the constraints I, J_1 and J_2. Compute the magnitude of the tangential reaction:

     $$R_{s\tau} = \sqrt{R_{s\tau_1}^2 + R_{s\tau_2}^2} \tag{15}$$

     Then:

     – If $R_{s\tau} \leq \mu R_{sn}$, then the Coulomb condition is satisfied. All the computed reactions are already correct and the calculation is complete.

     – Else $R_{s\tau} > \mu R_{sn}$ and the Coulomb condition is not satisfied, i.e. the provisional tangential reaction force causes the total provisional reaction force to lie outside Coulomb's cone. We evaluate a *reduction coefficient* α:

     $$\alpha = \frac{\mu R_{sn}}{R_{s\tau}} < 1 \tag{16}$$

     and then we scale the two tangential reactions accordingly. To this end, we scale down the Lagrange multipliers of the tangential constraints:

     $$\lambda_{J_1} \Leftarrow \alpha \lambda_{J_1} \qquad \lambda_{J_2} \Leftarrow \alpha \lambda_{J_2} \tag{17}$$

     which results in the following effect. For the nodes of the "slave" pinball we obtain, formally:

     $$\boldsymbol{R}_{s_i\tau} \Leftarrow \alpha \boldsymbol{R}_{s_i\tau} \qquad i = 1, \cdots, n_A \tag{18}$$

and, similarly, also the tangential reactions on the nodes m_i of the "master" pinball B are scaled by the same factor α:

$$\boldsymbol{R}_{m_i\tau} \Leftarrow \alpha \boldsymbol{R}_{m_i\tau} \qquad i = 1, \cdots, n_B \tag{19}$$

In the 2D case with friction, for each detected contact we have *two* constraints: one (provisional) constraint J along the tangent $\hat{\boldsymbol{\tau}}$, associated with Lagrange multiplier λ_J, followed by a constraint I along the normal $\hat{\boldsymbol{n}}$, of Lagrange multiplier λ_I. In the implementation, it is $I = J + 1$, that is, in the list of links the tangent constraint is directly followed by the corresponding normal constraint.

The rest of the algorithm is perfectly analogous to the 3D case, but with only one tangent direction instead of two. Thus, the tangent reaction $R_{s\tau}$ is computed directly and not through Eq. (15).

6.2.4 Computing the total reaction on the "slave" pinball

As anticipated in the previous Section, the total reaction on the "slave" (A) pinball $\boldsymbol{R}_s = \boldsymbol{R}_A$ must be computed by assembling the reactions at all the n_A nodes i_A of the element containing this pinball:

$$\boldsymbol{R}_s = \boldsymbol{R}_A = \sum_{i_A=1}^{n_A} \boldsymbol{R}_{i_A} \tag{20}$$

This task is performed in the new subroutine LINK_FRNORM_PINB of module M_LINK_PINB and is not straightforward. In the other contact models (GLIS/IMPA, GPIN and SLID) there is a (single) slave node (ISLAVE) in the very definition of the model. Therefore, it is sufficient to pre-calculate the reaction at this node.

In the PINB case, the reactions at all "slave" nodes (the nodes of the element containing the A pinball) must be assembled. However, such nodes are not directly identifiable from the subroutine LINK_FRNORM_PINB, which has only access to the COEFL array. This array is dimensioned over the total number of dofs involved by the PINB link, i.e. it contains dofs of both the A pinball and the B pinball.

When the link is first written in subroutine BL_LINL_PINBALLS of module M_PINBALLS, the link has the form:

$$\boldsymbol{v}_A \cdot \hat{\boldsymbol{n}}_{AB} - \boldsymbol{v}_B \cdot \hat{\boldsymbol{n}}_{AB} = \boldsymbol{0} \tag{21}$$

where $\hat{\boldsymbol{n}}_{AB}$ is the unit contact normal oriented from pinball A towards pinball B. By expanding the pinball velocities in terms of the corresponding nodal velocities using the shape functions, this results in:

$$\left(\sum_{i_A=1}^{n_A} N_{iA} \boldsymbol{v}_{iA} \right) \cdot \hat{\boldsymbol{n}}_{AB} - \left(\sum_{i_B=1}^{n_B} N_{iB} \boldsymbol{v}_{iB} \right) \cdot \hat{\boldsymbol{n}}_{AB} = \boldsymbol{0} \tag{22}$$

that is, the link contains first the dofs related to the A pinball, then those related to the B pinball.

However, this "natural" order is typically destroyed by the so-called *reordering* of the link, which is performed in subroutine REORDER_LIAISON of module M_LINKS_ADD, which is called from ADD_TRUE_LIAISON in the same module, that is called from ADD_PINB_LINK within the routine BL_LINK_PINBALLS that builds up the pinball links in module M_PINBALLS.

The links reordering (which is different from and should not be confused with the links *renumbering* described in Section 6.1), consists simply in arranging the DLIE (linked dofs) and the COEF (corresponding link coefficients) arrays (but *not* the PLIE array of the linked nodes) of a link in *growing* order. Apparently, this particular ordering is needed by the links renumbering algorithm. The reordering process acts within a single link, while the renumbering process alters the order in which the list of links is traversed (without re-arranging the list itself).

Therefore, in a data structure of type LIAISON (a link), after the reordering process:

- The PLIE array (linked nodes) remains in the natural order (as when the link is first written).
- The DLIE array is in growing order.
- The COEF array is in growing order (matching DLIE).

- The RANG array allows to pass from the growing to the natural order. RANG(I), I=1,NDLIE (with NDLIE the total number of dofs in this link) is the rank that the I-th dof, as listed in the growing order list DLIE, would have in a naturally ordered list of dofs (say DNAT, which is not available in the data structure) according to the natural list of nodes PLIE.

Thus, the RANG array is, besides the knowledge of the number of nodes of the A pinball n_A, the key to evaluating $\boldsymbol{R}_s$ according to (20). The RANG is added, together with NDLIE, to the exchange list of the LINK_FRNORM_PINB routine, while n_A is added to the LDATA array of the link and is passed to the LINK_FRNORM_PINB routine as NBNA (first argument) in place of ISLAVE. The loop to compute the (signed norm of the) total slave reaction FLC in LINK_FRNORM_PINB reads:

```
      SUBROUTINE LINK_FRNORM_PINB (NBNA, NBPLI, COEFL, SMLAG, ANORM,
     >                             FLC, ILIES, IR, ITYP, VNORM,
     >                             NDLIE, RANG)
*
 . . . (skip)
*
      AUX = SMLAG(ILIES)*ANORM(ILIES)
*
 . . . (skip)
*
      IF (IDIM == 2) THEN
        AUX1 = 0.D0 ! X-comp. of total contact force on "slave" (A) pinball
        AUX2 = 0.D0 ! Y-comp.
        DO K = 1, NBNA ! Loop over all nodes of "slave" (A) pinball
          I1C = IDIM*(K-1) + 1 ! Ordre croissant
          I2C = IDIM*(K-1) + 2
          I1N = RANG(I1C)      ! Ordre naturel
          I2N = RANG(I2C)
          IAD1 = IAD + I1N
          IAD2 = IAD + I2N
          AUX1 = AUX1 + COEFL(IAD1)*AUX
          AUX2 = AUX2 + COEFL(IAD2)*AUX
        END DO
        FLC = AUX1*VNORM(1) + AUX2*VNORM(2)
 . . . (skip)
      ELSE
        AUX1 = 0.D0 ! X-comp. of total contact force on "slave" (A) pinball
        AUX2 = 0.D0 ! Y-comp.
        AUX3 = 0.D0 ! Z-comp.
        DO K = 1, NBNA ! Loop over all nodes of "slave" (A) pinball
          I1C = IDIM*(K-1) + 1 ! Ordre croissant
          I2C = IDIM*(K-1) + 2
          I3C = IDIM*(K-1) + 3
          I1N = RANG(I1C)      ! Ordre naturel
          I2N = RANG(I2C)
          I3N = RANG(I3C)
          IAD1 = IAD + I1N
          IAD2 = IAD + I2N
          IAD3 = IAD + I3N
          AUX1 = AUX1 + COEFL(IAD1)*AUX
          AUX2 = AUX2 + COEFL(IAD2)*AUX
          AUX3 = AUX3 + COEFL(IAD3)*AUX
        END DO
        FLC  = AUX1*VNORM(1) + AUX2*VNORM(2) + AUX3*VNORM(3)
 . . . (skip)
      ENDIF
*
* il faut changer le signe de FLC comme dans le cas IMPA
      FLC = - FLC
*
      END SUBROUTINE LINK_FRNORM_PINB
```

7 Numerical examples to verify the upgrade

The upgrade of the friction algorithm in the PINB model presented in Section 6 has involved re-writing substantial parts of the PINB model implementation. As already noted, due to the complexity of the data structure and algorithms involved, the modifications may affect also the cases of contact without friction. Therefore, it is necessary to thoroughly check the numerical results.

This is done by first considering a set of small and simple contact examples. Then, as usual, the complete battery of non-regression tests is passed in order to verify that the new results do not differ (unless justified by the different and hopefully more accurate treatment of friction) from those obtained with the previous implementation.

7.1 Basic tests

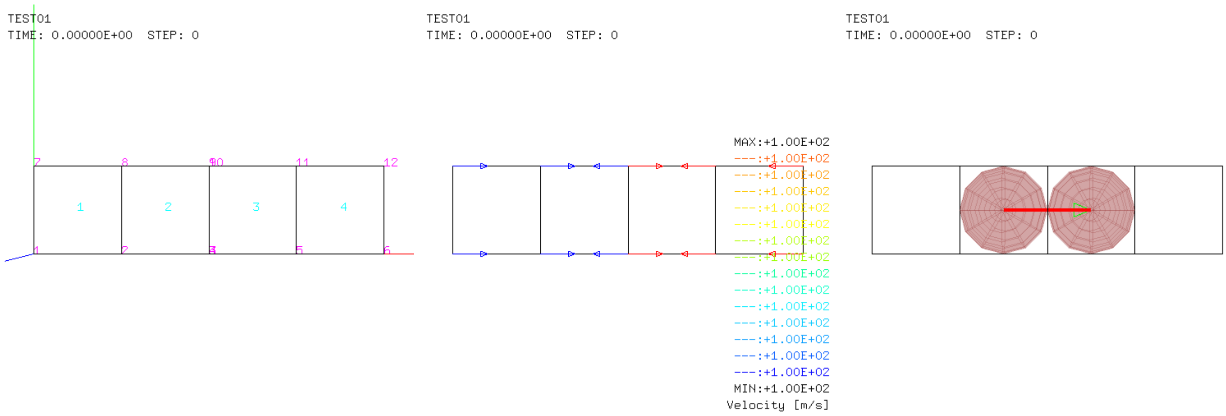
We start by a series of small basic tests. They are listed in Table 3.

Test	Mesh	Contact	Impact	Fric.	Split	Renu.	Rebo.	Groups	Links
TEST01	4 Q41L	`PINB`	Normal	NO	`NONE`	(`RENU`)	(`REB1`)	1	1
TEST02	4 Q41L	`PINB`	Normal	YES	`NONE`	(`RENU`)	(`REB1`)	1	2 or 1
TEST03	1 CUB8 1 PMAT	`GLIS`	Normal	NO	`NONE`	(`RENU`)	(a post.)	1	(1)
TEST04	3 CUB8 1 PMAT	`GLIS`	Normal	NO	`NONE`	(`RENU`)	(a post.)	—	—
TEST05	18 CUB8 1 PMAT	`GLIS`	Normal	NO	`NONE`	(`RENU`)	(a post.)	1	1
TEST06	18 CUB8 1 PMAT	`GLIS`	Normal	YES	`NONE`	(`RENU`)	(a post.)	1	3
TEST07	1 CUB8 1 PMAT	`IMPA`	Normal	NO	`NONE`	(`RENU`)	(a post.)	1	4
TEST08	1 CUB8 1 PMAT	`IMPA`	Normal	YES	`NONE`	(`RENU`)	(a post.)	1	12

Table 3: Basic tests to verify the upgraded PINB model.

7.1.1 Case TEST01

This test considers the perfectly normal impact between two bars in 2D, each one modeled by just two quadrangular elements Q41L, as shown in Figure 12. Zero-level (parent) pinballs are inserted in the two elements susceptible to come into contact. The setting is such that contact is detected already at step 0. No friction is prescribed. As per default, renumbering of the links is activated and the a priori rebound algorithm is used. The `SPLT NONE` option is activated but this, as well as the renumbering, has no effect because there is just one link in this particular case.



(a) Mesh (b) Velocities (c) Contacting pinballs

Figure 12: Definition of the test TEST01.

The contact-related part of the EPX input file reads:

```
LINK COUP SPLT NONE
     PINB BODY DIAM 1.0 LECT 2 TERM
          BODY DIAM 1.0 LECT 3 TERM
```

In this case there is just one link and the solution is performed by SOLVE_SINGLE_LIAISON. The results obtained with the upgraded model are identical (listing) to those obtained with the previous model.

7.1.2 Case TEST02

This test is identical to TEST01 but we add friction. Therefore, upon contact two links are generated (tangential, then normal) instead of just one (with the upgraded model). The solution is performed in SOLVE_GROUP, since the `SOL2` option is not set. Friction should have no effect on the solution here, since the impact is perfectly normal.

Note that the previous model generated only one link in this test (at the 0 step). This is because the old model removed all terms with zero coefficients when writing the links (in order to reduce the number of terms in the links system). Since the impact is perfectly normal here, all coefficients of the tangent link vanish and therefore the entire link was skipped. Instead, the upgraded model keeps all link terms, including those with zero coefficients (in analogy with the other contact models), for simplicity, see the upgraded version of BL_LINK_PINBALLS.

The results obtained with the upgraded model are identical (listing) to those obtained with the previous model. The only differences concern some non-significant values $\mathcal{O}(10^{-11})$ and the value of WIMP, which was zero with the old version and is now non-zero. But the latter is just a post-treatment quantity and therefore has no influence on the actual results.

7.1.3 Case TEST03

This test is conceptually similar to TEST01 but uses the GLIS contact model instead of PINB in order to allow for a (qualitative) comparison between the two models. Since GLIS is only available in 3D, we use a CUB8 element to represent the projectile (left bar). The target (right bar) is replaced by a PMAT which should impact the projectile at the center of its right face, see Figure 13.

The contact-related part of the EPX input file reads:

```
LINK COUP SPLT NONE
     GLIS 1 MAIT LECT proj TERM
            PESC LECT targ TERM
```

Unfortunately, as it is often the case with GLIS in cases with very coarse discretizations (just one element here), the contact is not well detected. One penetration is detected in this case, as expected, both at step 0 and at the succeeding steps, as the material point enters more and more into the cube. However, at step 0 no contact forces are developed, see Figure 13(c). At step 1, some very weak reactions are developed, but they have a weird orientation, like if the penetration would concern the back face of the cube rather than its right one, see Figure 13(d).

The solution is therefore completely wrong and it makes no sense to compare the results obtained by the two versions of the executable.

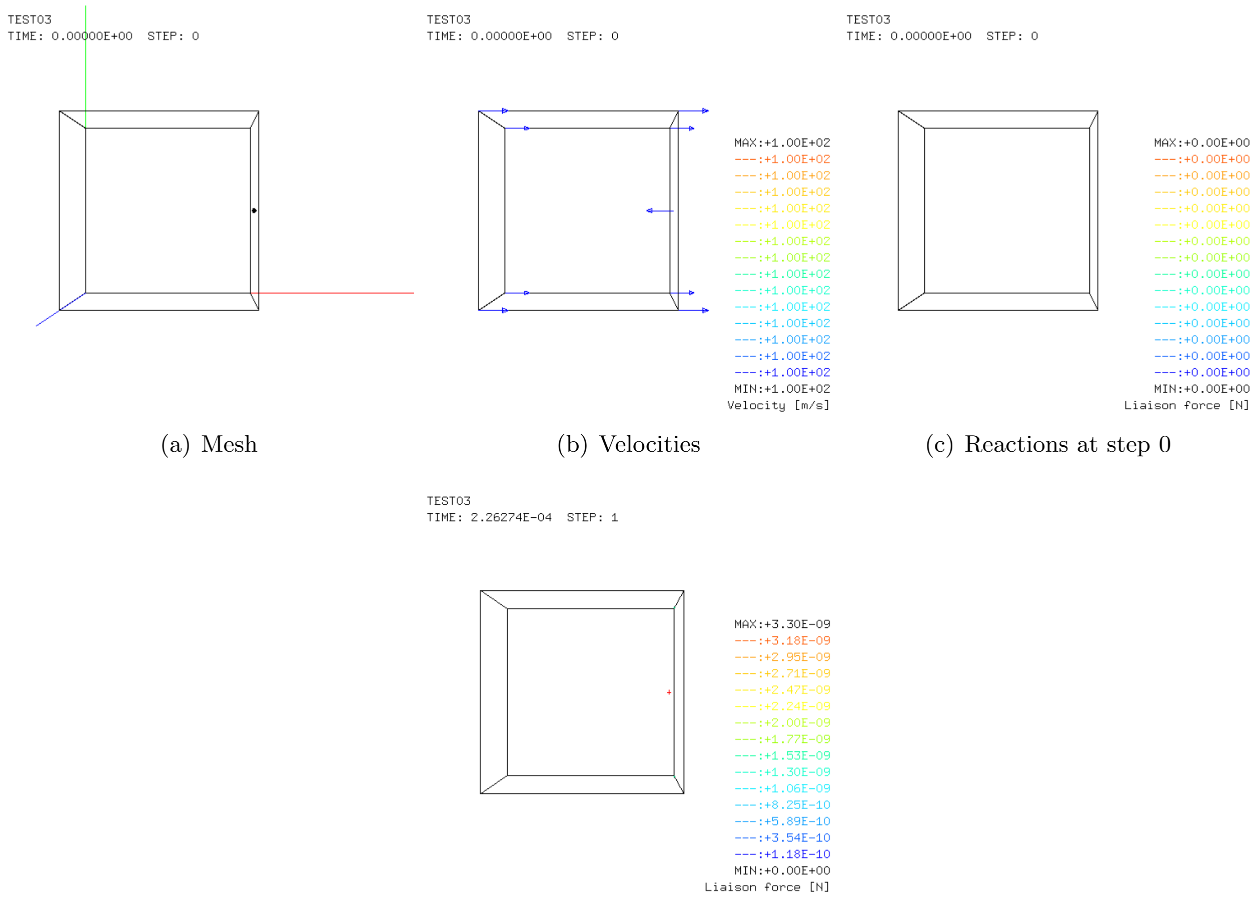
7.1.4 Case TEST04

This test is similar to TEST03 but we use three cubes in a row in the projectile in the hope of ameliorating the contact detection by GLIS, see Figure 14. Unfortunately, this is not the case. No penetration is ever detected, the contact forces are always zero and the material point traverses the solid mesh undisturbed.

The solution is therefore completely wrong and it makes no sense to compare the results obtained by the two versions of the executable.

7.1.5 Case TEST05

This test is similar to TEST04 but we use an array of $2 \times 3 \times 3$ cubes in the projectile in the hope of ameliorating the contact detection by GLIS, see Figure 15.



(a) Mesh

(b) Velocities

(c) Reactions at step 0

(d) Reactions at step 1

Figure 13: Definition and results of the test TEST03.

Indeed, in this case, the contact is correctly detected, already starting at step 0, see the (significant) contact forces in Figure 15(c). Contact continues until step 2, and from step 3 on the rebound occurs and the contact is broken.

The solution looks correct, and is identical (listing) with the two versions of the executable. This test is therefore suitable for a comparison (necessarily qualitative, due to the different geometry) with solutions using PINB obtained previously.

7.1.6 Case TEST06

This test is similar to TEST05 but we add friction. In this case, there are three links (two tangential and one normal) forming one group of links. Therefore, the solution is carried on in SOLVE_GROUP, whereas in case TEST05 there was just one link, so the solution was performed in SOLVE_SINGLE_LIAISON (by calling SOLVE_SINGLE_GENERIC).

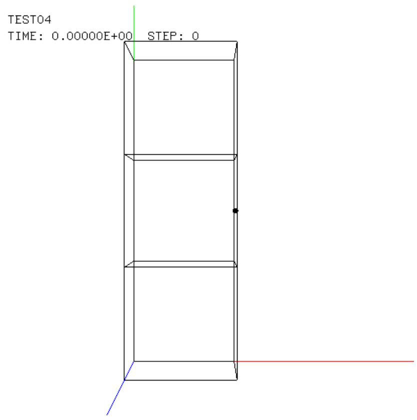
The solution looks correct, and is identical (listing) with the two versions of the executable. This test is therefore suitable for a comparison (necessarily qualitative, due to the different geometry) with solutions using PINB (with friction) obtained previously.

In particular, this solution was used to verify the implementation of the calculation of the total reaction on the "slave" pinball described in Section 6.2.4.

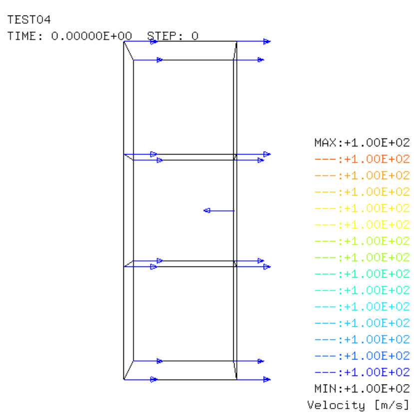
7.1.7 Case TEST07

This test is similar to TEST03 but it uses the IMPA contact model instead of GLIS. The relevant part of the EPX input file reads:

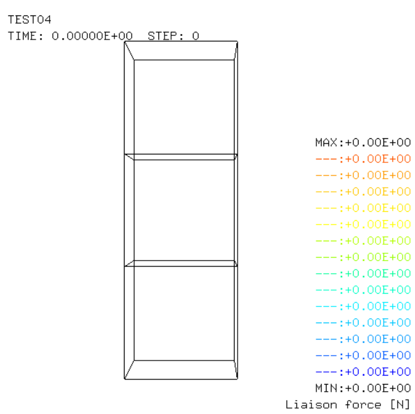
```
TEST07
 . . .
```



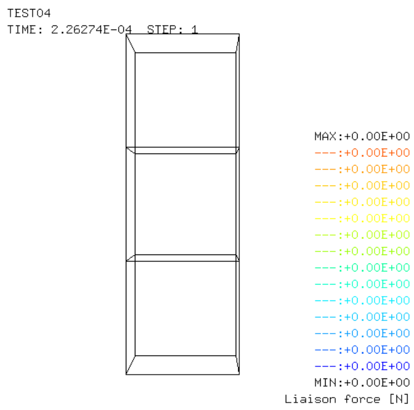
(a) Mesh



(b) Velocities

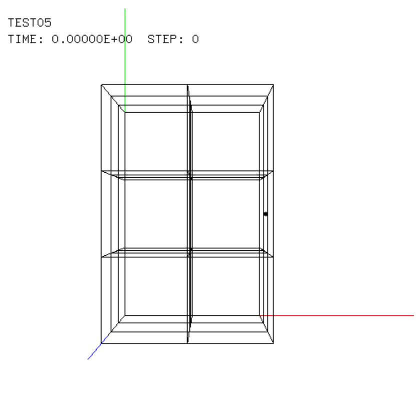


(c) Reactions at step 0

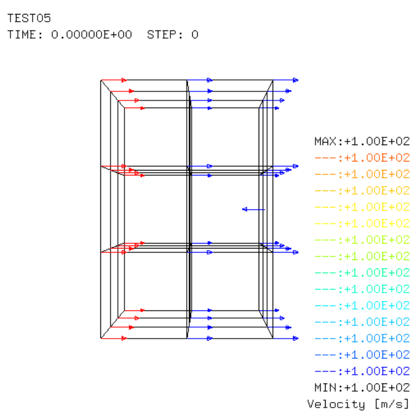


(d) Reactions at step 1

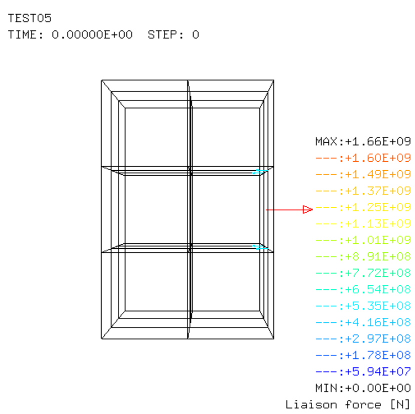
Figure 14: Definition and results of the test TEST04.



(a) Mesh



(b) Velocities



(c) Reactions at step 0

Figure 15: Definition and results of the test TEST05.

```
COMP . . .
     NGRO 1 'cibl' LECT 2 3 6 7 TERM
 . . .
LINK COUP SPLT NONE
     IMPA DDL 1 COTE -1
          PROJ LECT targ TERM
          CIBL LECT cibl TERM
```

Unlike GLIS, IMPA is able to correctly detect the contact. The solution looks correct and is identical (listing) with the two versions of the executable. Note that IMPA writes four links, probably one (normal) link for each node of the impacted face. Therefore, the solution occurs in SOLVE_GROUP.

The `LAGC` optional keyword has not been specified in the input data set after `LAGR TRID`. However, it is believed that this keyword affects only the old `LIAI` links model, and not the `LINK COUP` model used here. By adding the `LAGC` keyword, the test gives exactly the same results (listing) in the present case, and this seems to confirm the above assumption.

7.1.8 Case TEST08

This test is similar to TEST07 but we add friction. In this case IMPA writes twelve links, most likely three links (two tangent and one normal) for each node of the impacted face.

The solution is identical (listing) with the two versions of the executable. However, the solution differs from that of TEST07 which had no friction, despite the fact that the impact is perfectly normal. This may perhaps reveal an issue in the IMPA model, but even if this were the case, the result has not been modified by the present developments.

7.1.9 Non-regression benchmarks

All non-regression benchmarks that use the keywords GLIS, IMPA, PINB, GPIN or SLID are run with the standard and upgraded version, and the results are compared. At the moment of this writing, these consist of 231 sequential tests and 42 MPI tests.

All tests pass the qualification with the upgraded version, including the bench BM_STR_PINB_FR-OT that has been introduced to summarize the tests presented in Section 5, which use the PINB model with friction.

Only minor differences (invisible at the naked eye) are detected (by `diff`) in the PostScript curves of 4 tests (three sequential and one MPI), and in a handful of .BMP and .AVI files. Therefore, it is concluded that the upgraded version is safe and can be evolved.

7.2 Sliding cube in 2D

We now consider a slightly more realistic test than the basic ones of the previous Section. A metallic cube is pushed against a metallic plane by a constant pressure applied on its top face. Then, after the elastic oscillations have been damped out, a second pressure is suddenly applied to the lateral face, in the direction tangent to the plane, causing accelerated sliding of the cube along the plane.

We first solve the problem in 2D, where the cube is a unit square and the plane is a 7×1 m rectangle. The initial problem geometry is shown in Figure 16. The cube is shown in cyan, the plane in green. The base (bottom face) of the plane is blocked.

Contact between the cube and the plane is realized by inserting pseudo-nodal pinballs (appearing as small red dots in the Figure) in the bottom face of the cube. Hierarchic pinballs (`MLEV 5`) are inserted in the top row of elements of the plane.

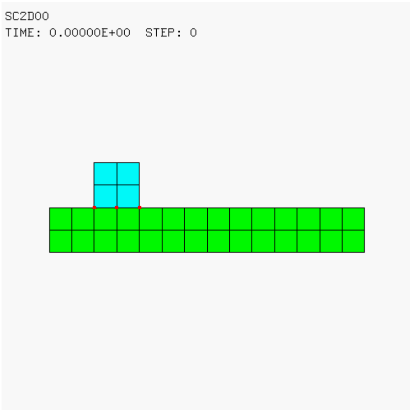


Figure 16: Definition of the sliding cube test in 2D.

The simulations performed are summarized in Table 4.

Test	Mesh	Description	$t_{\rm fin}$ [ms]	Steps	CPU [s]
SC2DQS	32 Q42L 3 PMAT 4 CL2D	Contact by `PINB`, no horizontal pressure	85	5 636	1.9
SC2D00	32 Q42L 3 PMAT 4 CL2D	Only horizontal pressure	85	5 636	1.6
SC2D01	32 Q42L 3 PMAT 4 CL2D	Both pressures, no friction	85	5 636	2.1
SC2D02	32 Q42L 3 PMAT 4 CL2D	Both pressures, with friction	85	5 636	2.0
SC2D03	32 Q42L 3 PMAT 4 CL2D	Idem 02 but old exe (1-tangent friction)	85	6 099	1.8
SC2D04	32 Q42L 3 PMAT 4 CL2D	Idem 01 (no friction) but penalty	85	5 637	1.5
SC2D05	32 Q42L 3 PMAT 4 CL2D	Idem 02 (friction) but penalty	85	5 637	1.5
SC2D07	8 Q42L 2 PMAT 1 CL2D	Punching problem, no friction, no damping	85	2 818	1.3
SC2D08	8 Q42L 2 PMAT 1 CL2D	Idem 07 but with friction and damping	85	2 818	1.4
SC2D09	8 Q42L 2 PMAT 1 CL2D	Idem 08 but `NORE`	85	2 818	1.3
SC2D10	8 Q42L 2 PMAT 1 CL2D	Idem 08 but no friction	85	2 818	1.4
SC2D11	32 Q42L 4 CL2D	Idem 01 (no friction) but GPIN	85	5 636	1.7
SC2D12	32 Q42L 4 CL2D	Idem 02 (with friction) but GPIN	85	5 636	1.8
SC2D13	32 Q42L 4 CL2D	Idem 01 (no friction) but SLID	85	5 636	1.7
SC2D14	32 Q42L 4 CL2D	Idem 02 (with friction) but SLID	85	5 636	1.6

Table 4: Sliding cube tests in 2D.

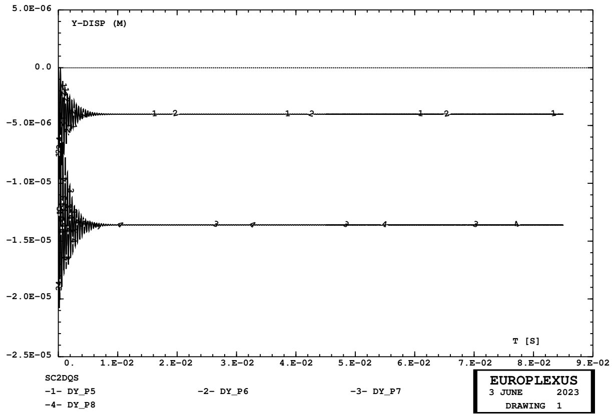
7.2.1 Case SC2DQS

This is a preliminary version of the 2D test, where only the vertical pressure is applied. The scope is to calibrate the `QUAS STAT` damping in order to damp out the elastic oscillations produced by the sudden (step-like) application of the (constant) vertical pressure from the initial time.

No friction is specified in the contact directive. Indeed, in this type of setting (only vertical pressure) the influence of friction should be irrelevant, except for the small horizontal motions caused by the Poisson effect.

Figure 17 shows the vertical displacements of the four corner points of the square. The displacement of the two bottom points is caused by the elastic deformation (compression) of the plane. The displacement of the two top points is larger, because it includes also the deformation of the cube.

It can be seen that, thanks to the quasi-static damping applied, the elastic oscillations almost completely disappear from $t_{\mathrm{QS}} = 10$ ms. Therefore, in the subsequent tests, the horizontal pressure will be applied starting from $t = t_{\mathrm{QS}} = 10$ ms.



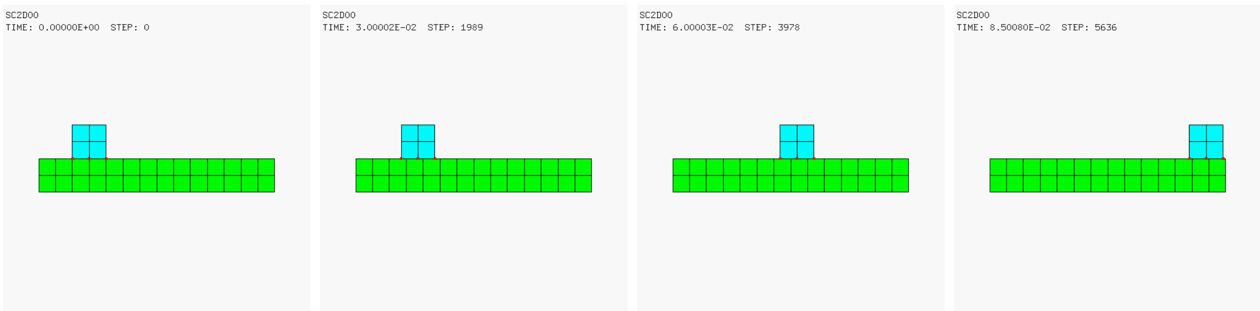
(a) Y-displacements

Figure 17: Results of test SC2DQS.

7.2.2 Case SC2D00

This test is similar to SC2DQS but we apply only the horizontal pressure and, therefore, no contact is specified between the cube and the plane. The scope is to obtain the free horizontal motion of the cube, which is uniformly accelerated from $t_{\mathrm{QS}} = 10$ ms, and will be used as a reference in the subsequent simulations.

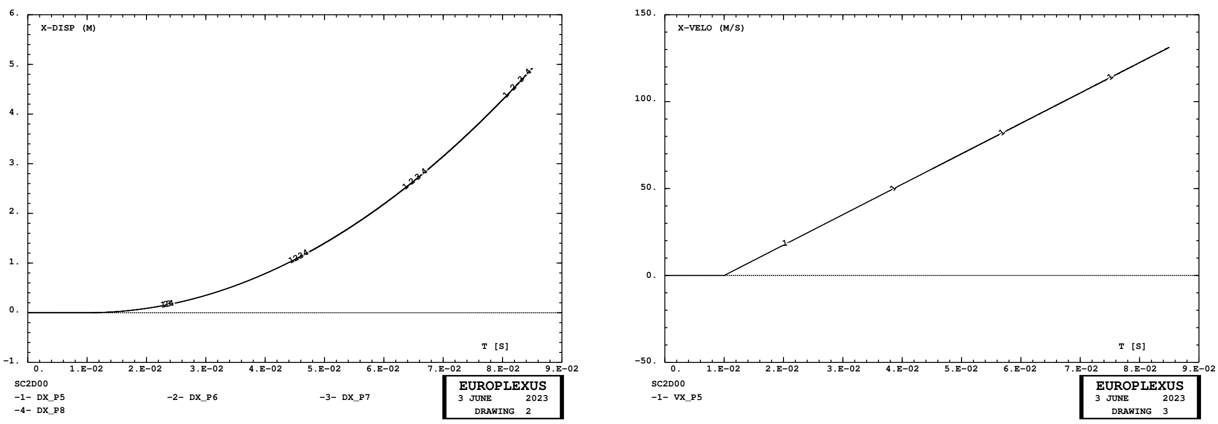
Figure 18 shows the accelerated cube motion by a sequence of deformed mesh images at some characteristic times. Although the plane (in green) is present, it has no effect on the results. The quasi-static damping is prescribed (until $t_{\mathrm{QS}} = 10$ ms) in this simulation, like in the previous one. However, it will have no effect on the results since the horizontal pressure is applied only later.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 18: Motion of the cube in test SC2D00.

Figure 19 shows the X-displacement and the X-velocity histories. The velocity is zero until $t_{\mathrm{QS}} = 10$ ms, then it ramps up linearly under constant acceleration, due to the constant applied pressure. These curves can be used as references for comparison of the subsequent simulations, where contact (first without and then with friction) will be added.



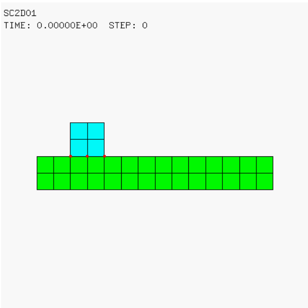
(a) X-displacements (b) X-velocities

Figure 19: Further results of test SC2D00.

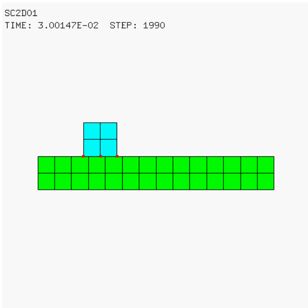
7.2.3 Case SC2D01

This test is similar to SC2D00 but we add also the vertical pressure and the contact (without friction).

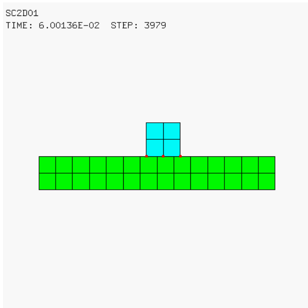
Figure 20 shows the accelerated cube motion by a sequence of deformed mesh images at some characteristic times. The quasi-static damping is prescribed (until $t_{QS} = 10$ ms) in this simulation, like in the previous ones.



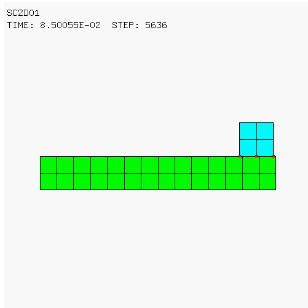
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



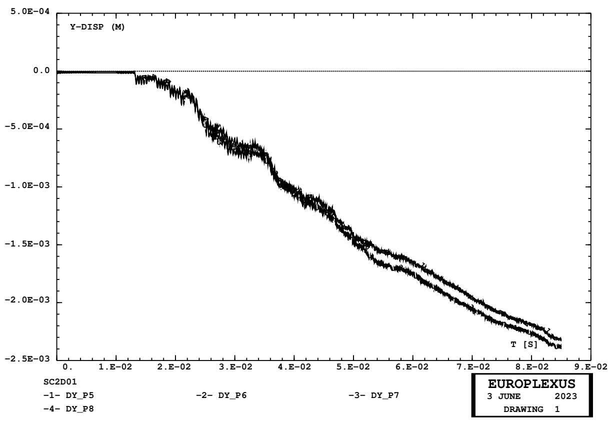
(d) $t = 85$ ms

Figure 20: Motion of the cube in test SC2D01.

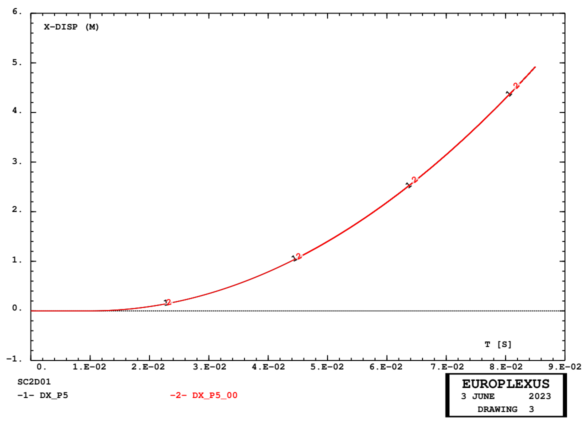
Figure 21 shows the Y-displacement and the X-displacement histories, the latter compared with solution SC2D00. The vertical displacement remains very small, but is not zero. This is in part due to the deformability of the plane. However, one may see that the cube slowly penetrates into the plane as it slides along it.

This effect is undesirable, and could eventually destroy the solution for very large horizontal displacements of the cube. It is probably due to the fact that the LM method does not provide a restoring (penalty-like) force proportional to the penetration. To eliminate it, one could perhaps introduce an additional penalty term, to complement the LM-based reaction. However, it is not known whether such a practice has ever been attempted in other codes, and the subject remains open for further investigation.

The horizontal displacement (black curve in Figure 21(b)) has a parabolic shape and is in very good agreement with the solution SC2D00 (red curve), as expected.



(a) Y-displacements



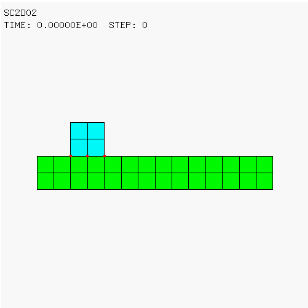
(b) X-displacements

Figure 21: Further results of test SC2D01.

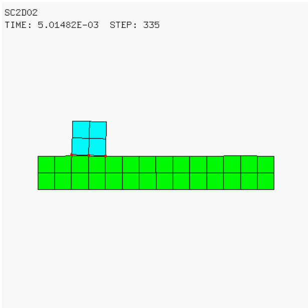
7.2.4 Case SC2D02

This test is similar to SC2D01 but we add friction. Figure 22 shows the accelerated cube motion by a sequence of deformed mesh images at some characteristic times. Although difficult to appreciate in these images, during the first 10 ms (when only the vertical pressure is active) and shortly after, the cube undergoes a strange vertical motion/rotation (see also Figure 23(a)). This effect might be due to the friction model operating in the static regime, and will have to be investigated.

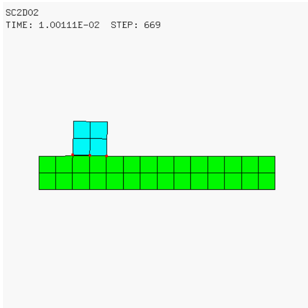
When the horizontal pressure is activated, and after the cube has gained a certain horizontal velocity, the solution becomes more credible. As shown in Figure 23(b), due to friction, at the final time the cube (black curve) has displaced less than in the previous, friction-less solution (red curve), as expected.



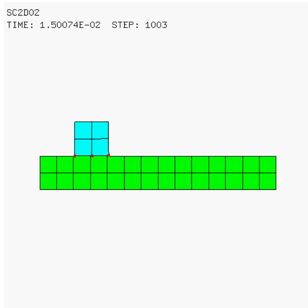
(a) $t = 0$



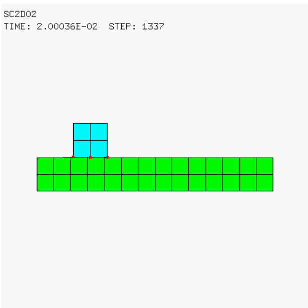
(b) $t = 5$ ms



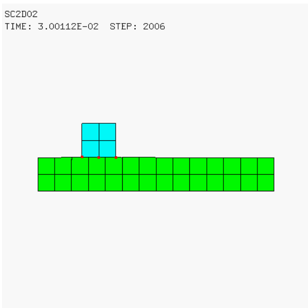
(c) $t = 10$ ms



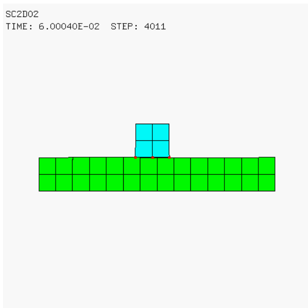
(d) $t = 15$ ms



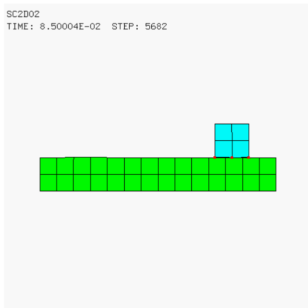
(e) $t = 20$ ms



(f) $t = 30$ ms

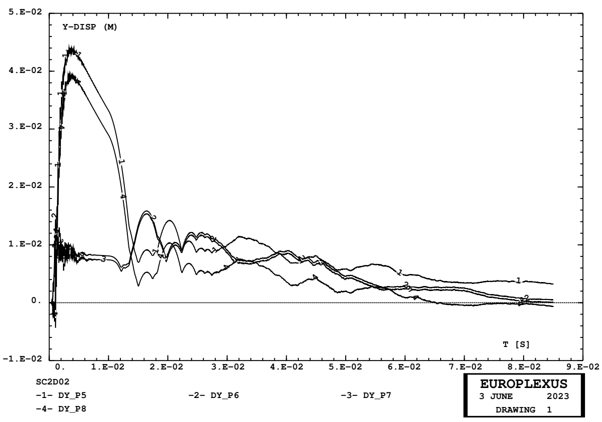


(g) $t = 60$ ms

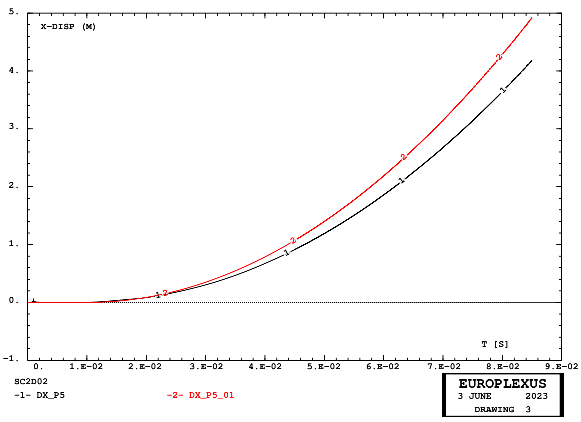


(h) $t = 85$ ms

Figure 22: Motion of the cube in test SC2D02.



(a) Y-displacements



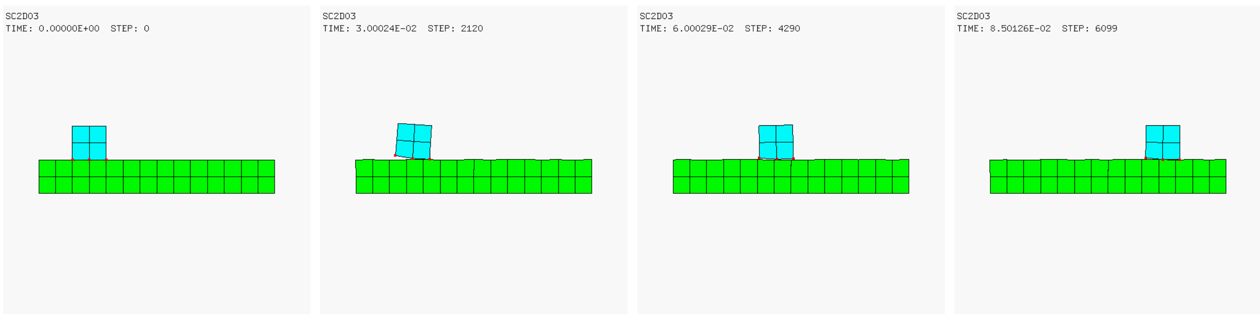
(b) X-displacements

Figure 23: Further results of test SC2D02.

7.2.5 Case SC2D03

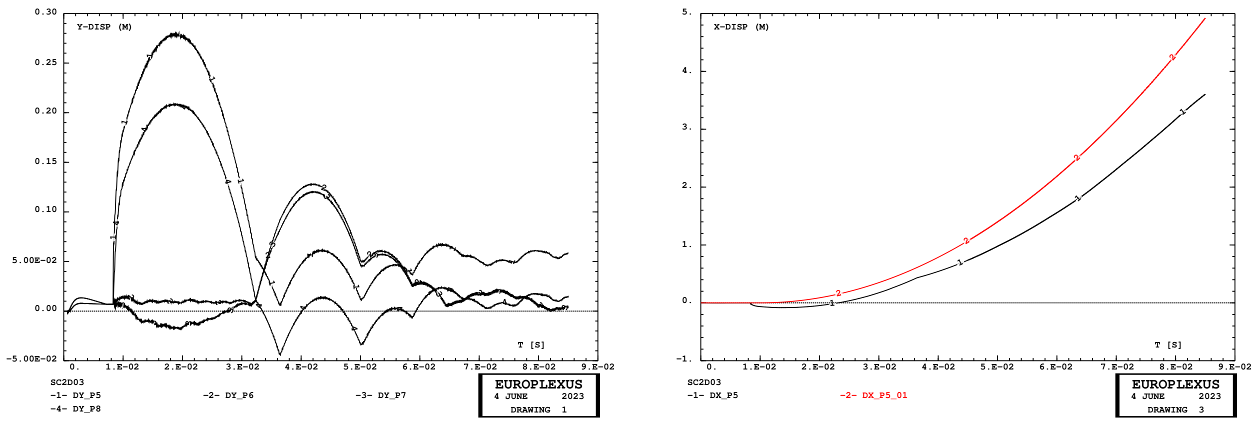
This test is identical to SC2D02 but it is run with the old EPX executable, which used the 1-tangent friction model instead of the 2-tangent model. However, being in 2D, there is just one tangent in both models anyway.

The solution looks quite weird, see Figures 24 and 25.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 24: Motion of the cube in test SC2D03.



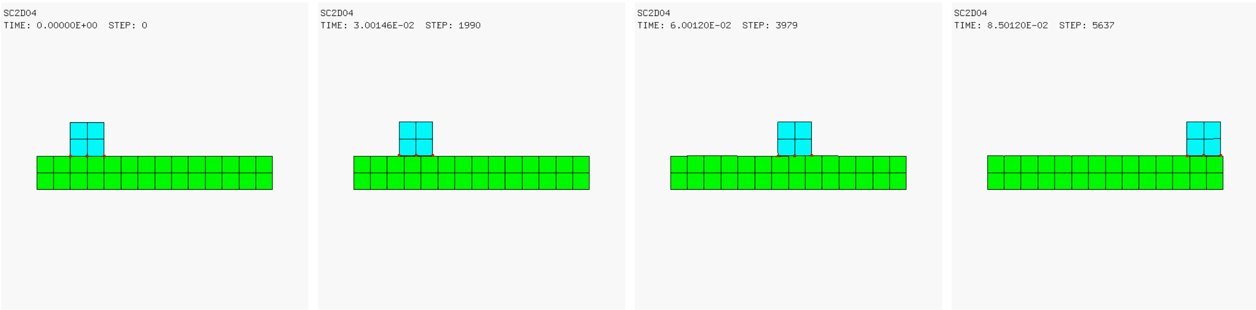
(a) Y-displacements (b) X-displacements

Figure 25: Further results of test SC2D03.

7.2.6 Case SC2D04

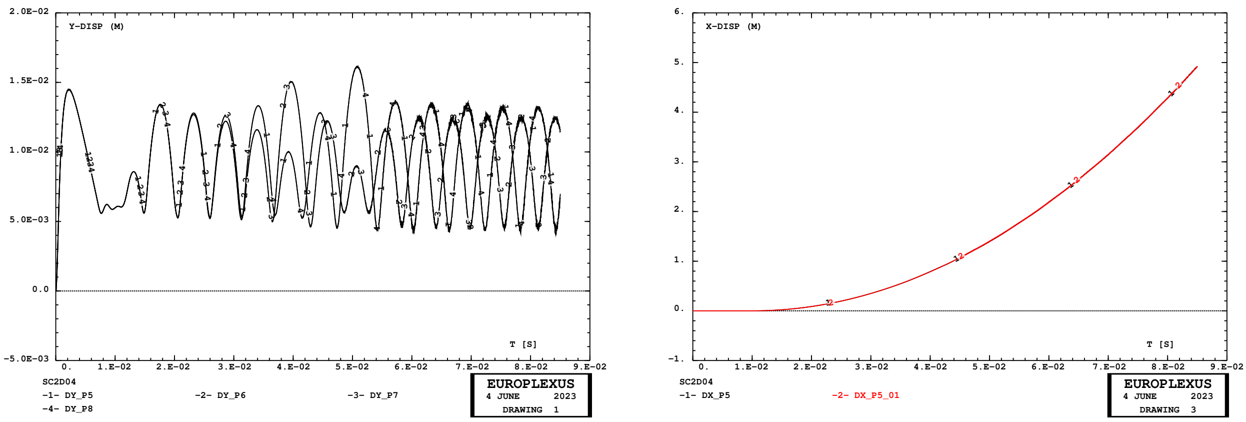
This test is identical to SC2D01 (no friction) but it uses the penalty formulation of the pinball contact model (`LINK DECO PINB PENA`).

The solution looks correct, see Figures 26 and 27.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 26: Motion of the cube in test SC2D04.



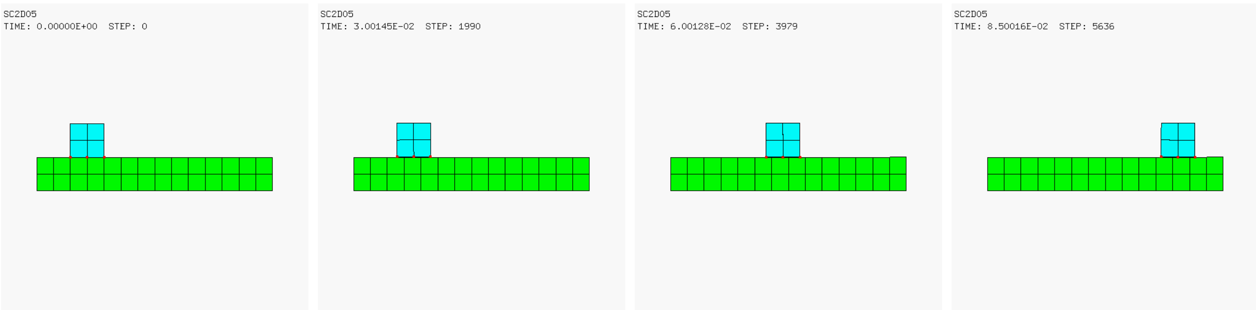
(a) Y-displacements (b) X-displacements

Figure 27: Further results of test SC2D04.

7.2.7 Case SC2D05

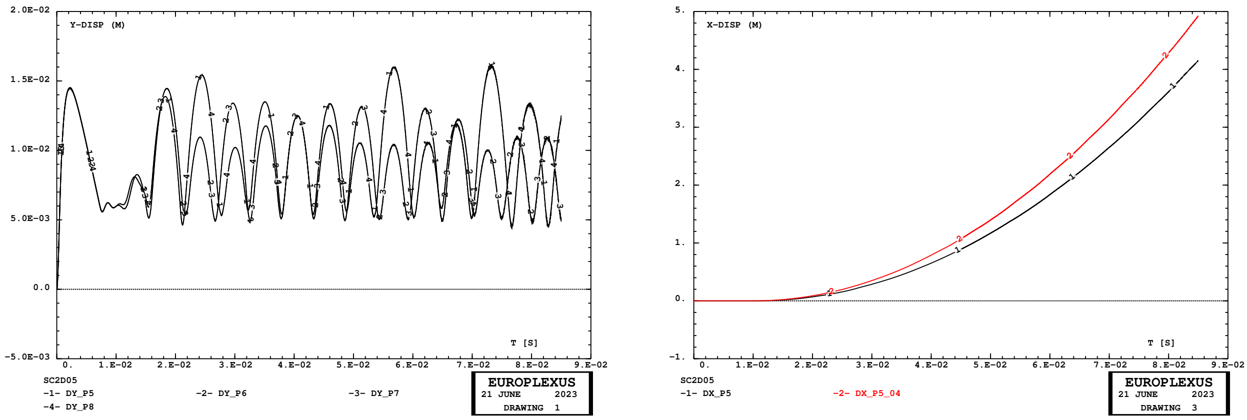
This test is identical to SC2D02 (with friction) but it uses the penalty formulation of the pinball contact model (`LINK DECO PINB PENA`).

The solution is smooth, see Figures 28 and 29. As expected, due to friction, the horizontal displacement of the cube is less than in case SC2D04, which had no friction.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 28: Motion of the cube in test SC2D05.



(a) Y-displacements (b) X-displacements

Figure 29: Further results of test SC2D05.

7.2.8 Case SC2D07

This test is a simplified version of the previous ones, designed to check some particular aspects. A penetrator punches vertically into a target, under the effect of a vertical pressure on its upper face, without tangential sliding, see Figure 30. So the only tangential motions are those caused by the Poisson effect in the material.

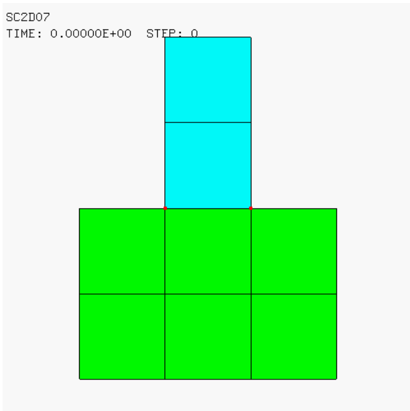
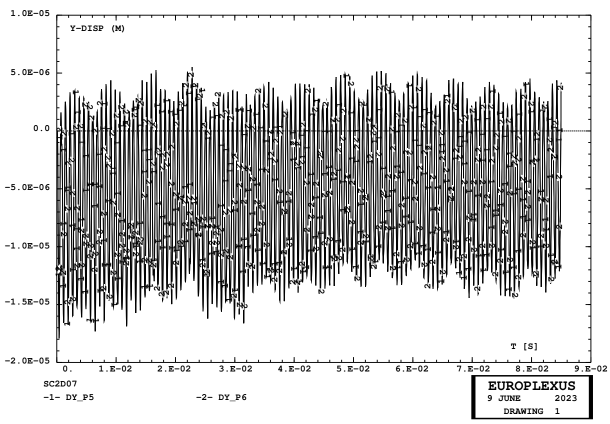


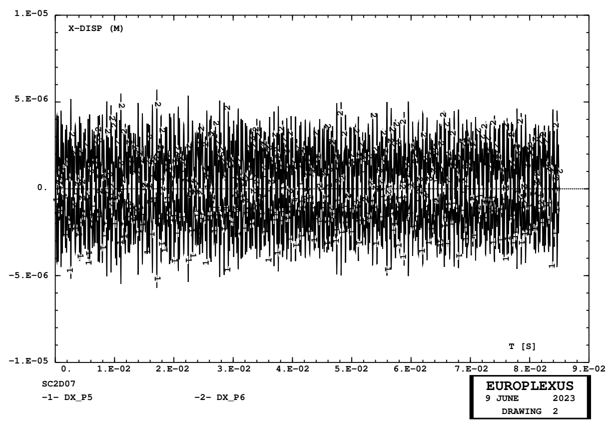
Figure 30: Definition of the punching test in 2D.

No friction is activated in this first simulation. Scope of the simulation is to check stability and symmetry of the solution, and to verify that the resultants of the contact forces (reactions) in the two contacting bodies are equal and opposite.

Al the above requirements are verified. The solution is noisy, see Figure 31, but this was expected since the material is elastic and there is no quasi-static damping.



(a) Y-displacements



(b) X-displacements

Figure 31: Further results of test SC2D07.

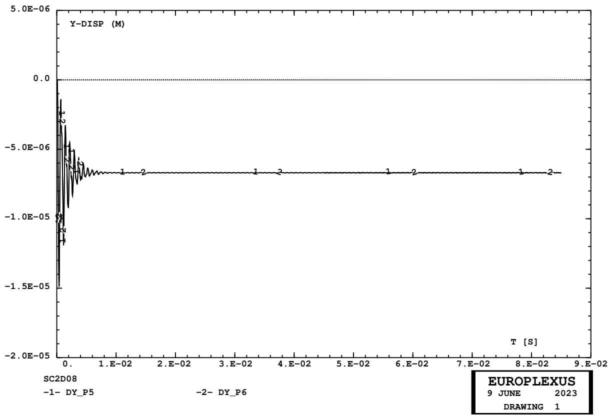
To check the resultants, we define two regions (`REGI` directive) consisting of all nodes associated with pinballs in the target and, respectively, in the penetrator. The damping is disabled because, if present, it would introduce some dissipative forces and the symmetry of the resultants would be violated. This was indeed observed in the previous tests, as long as the quasi-static damping remained active. The values of the resultants are checked from the listing.

7.2.9 Case SC2D08

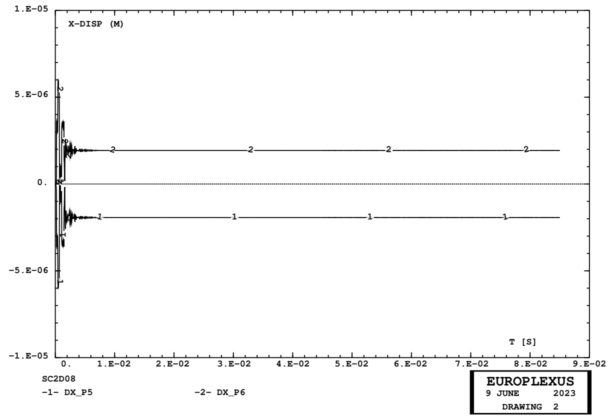
This test is similar to SC2D07 but we add friction and activate the quasi-static damping until $t_{QS} = 10$ ms.

The solution is stable. The contact force resultants are unsymmetric as long as quasi-static damping is applied, but become perfectly symmetric as soon as the damping is removed.

The solution in terms of displacements is shown in Figure 32 and looks correct. It is smooth thanks to the quasi-static damping.



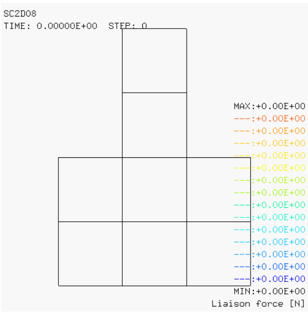
(a) Y-displacements



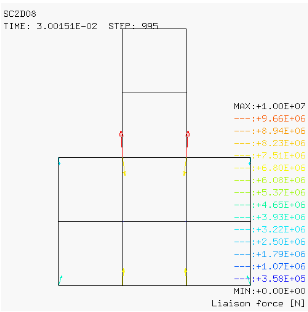
(b) X-displacements

Figure 32: Results of test SC2D08.

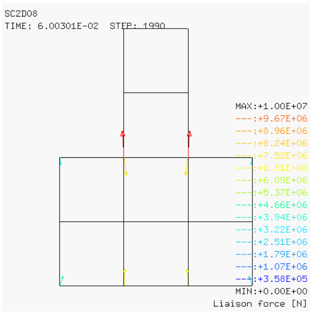
From the animation, by visualizing the contact forces, we see that what we obtain, after an initial transient, is the expected static solution, see Figure 33. The reactions are slightly inclined, as expected, due to the friction, but they fall within the Coulomb cone for the chosen value of μ.



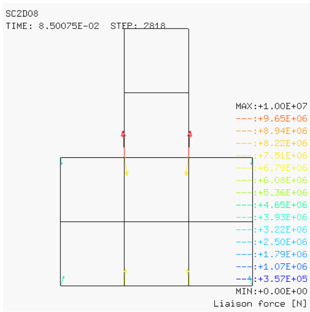
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



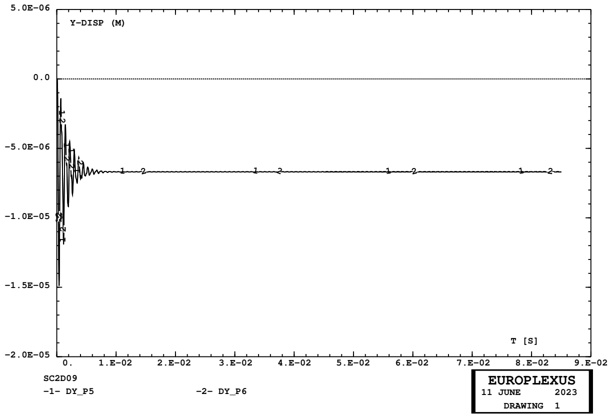
(d) $t = 85$ ms

Figure 33: Deformed configuration and contact forces in test SC2D08.

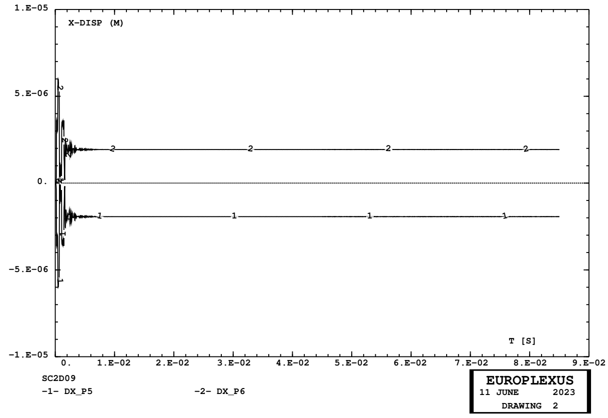
7.2.10 Case SC2D09

This is a repetition of case SC2D08 by adding the `NORE` keyword, which disables the renumbering of the links.

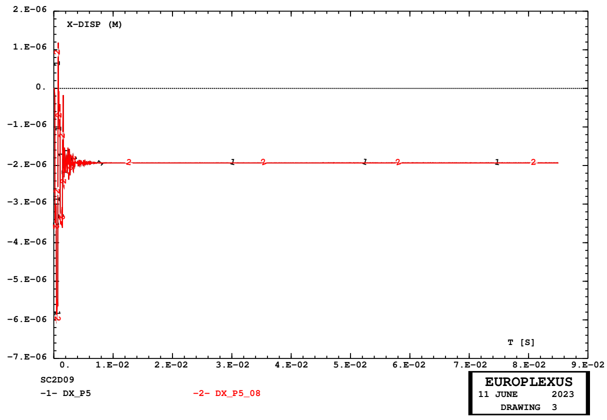
The solution is successful and visually identical to that of case SC2D08, see Figures 34 and 35.



(a) Y-displacements

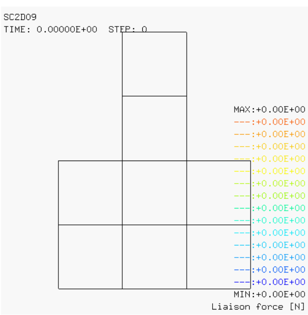


(b) X-displacements

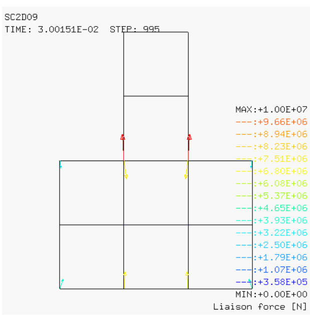


(c) SC2D09 vs. SC2D08

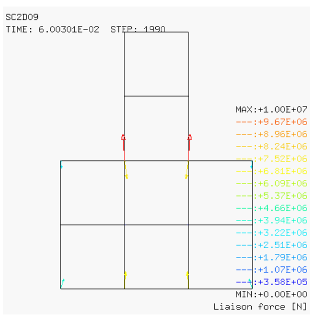
Figure 34: Results of test SC2D09.



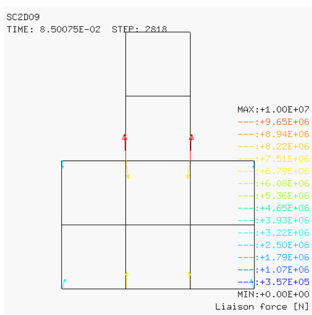
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



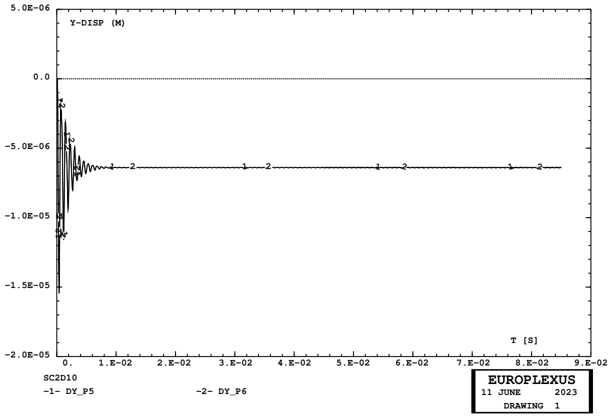
(d) $t = 85$ ms

Figure 35: Deformed configuration and contact forces in test SC2D09.

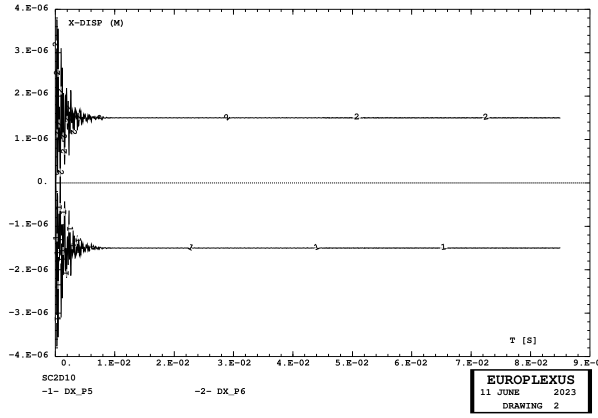
7.2.11 Case SC2D10

This test is identical to SC2D08 but we remove the friction.

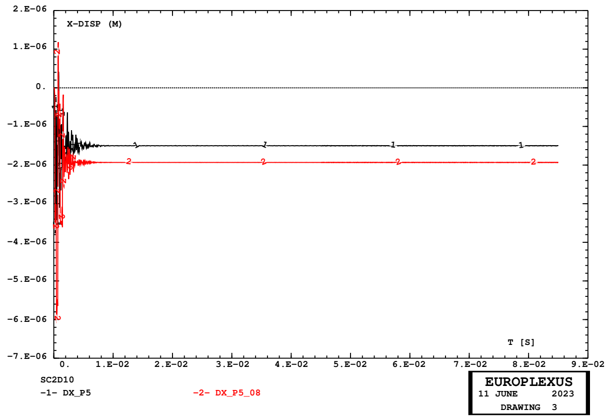
The solution is successful, see Figures 36 and 37. the reaction forces are perfectly vertical since there is no friction.



(a) Y-displacements

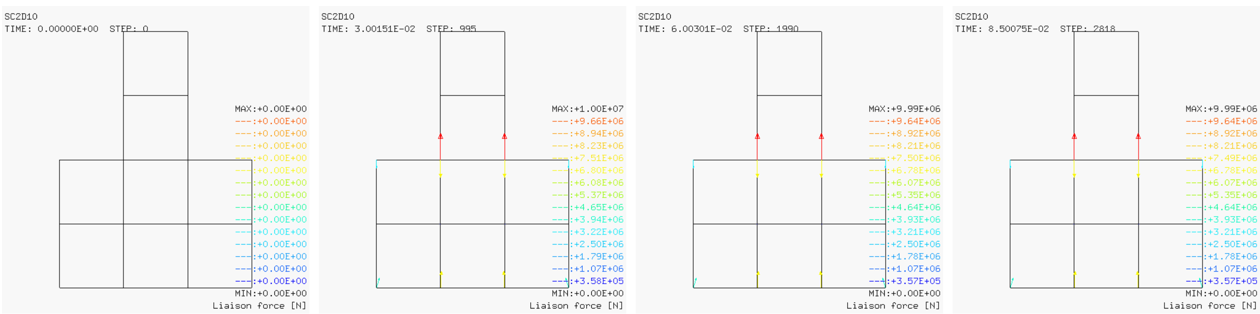


(b) X-displacements



(c) SC2D10 vs. SC2D08

Figure 36: Results of test SC2D10.



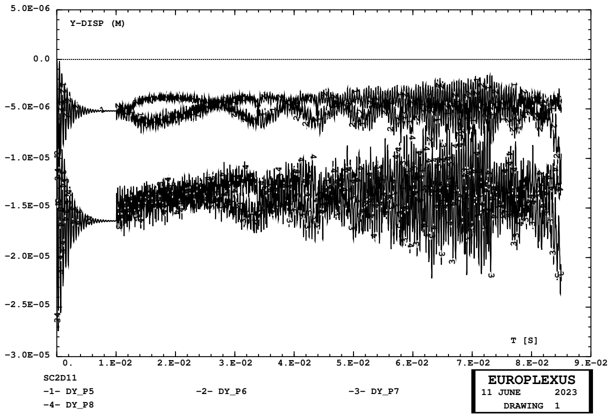
(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 37: Deformed configuration and contact forces in test SC2D10.

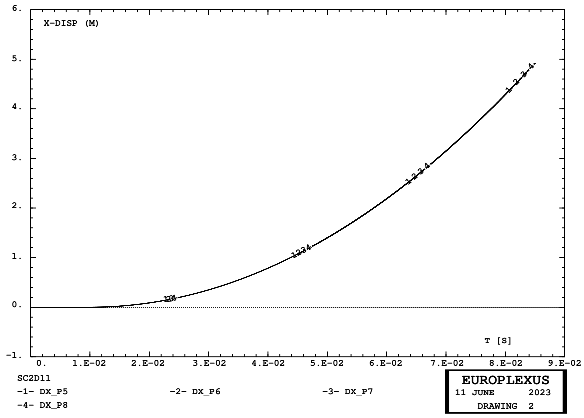
7.2.12 Case SC2D11

This test is similar to SC2D01 (no friction) but uses the GPIN contact model instead of PINB.

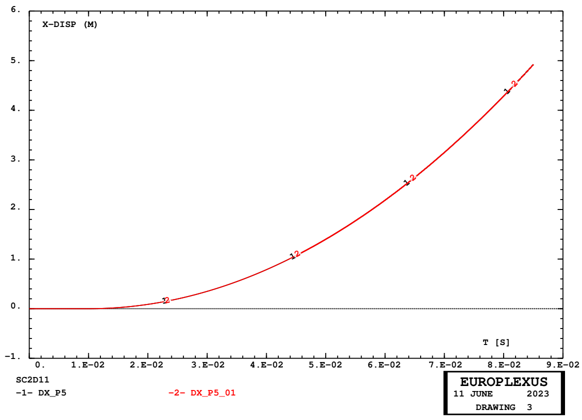
The solution is successful and visually identical to that of case SC2D01, see Figures 38 and 39.



(a) Y-displacements

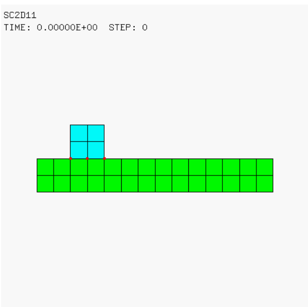


(b) X-displacements

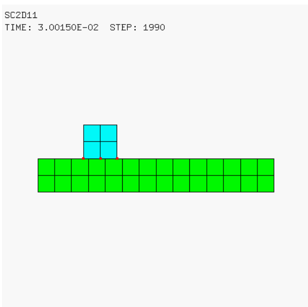


(c) SC2D11 vs. SC2D01

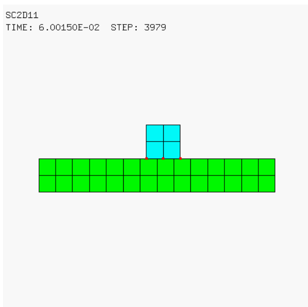
Figure 38: Results of test SC2D11.



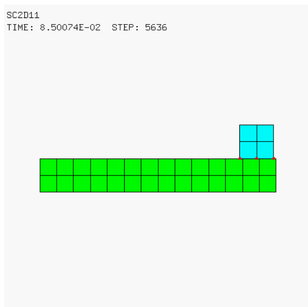
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



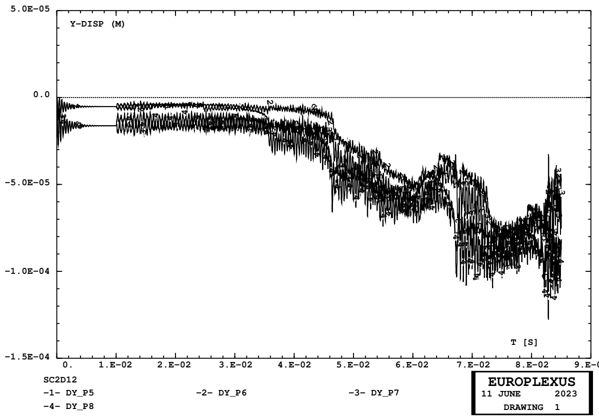
(d) $t = 85$ ms

Figure 39: Deformed configuration and contact forces in test SC2D11.

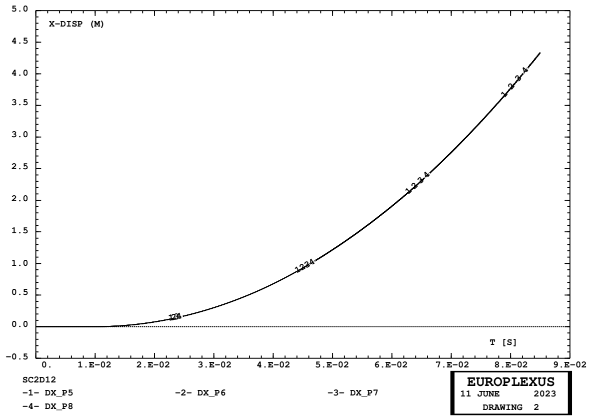
7.2.13 Case SC2D12

This test is similar to SC2D02 (with friction) but uses the GPIN contact model instead of PINB.

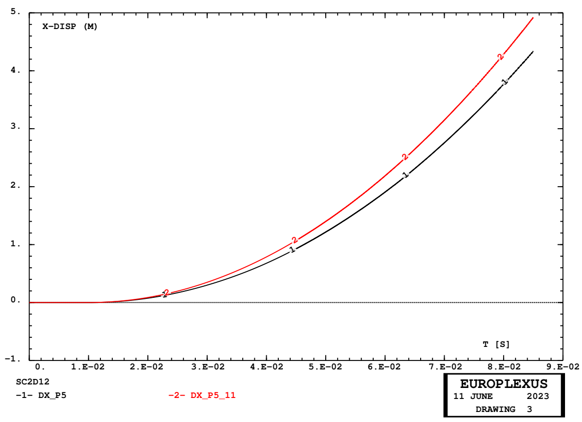
The solution is successful, see Figures 40 and 41. The horizontal displacement is less than that of case SC2D11, as expected, but larger than that of case SC2D02 (PINB).



(a) Y-displacements

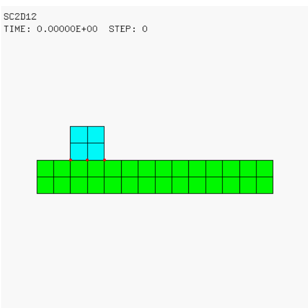


(b) X-displacements

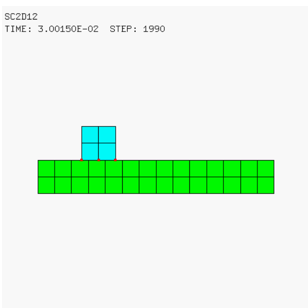


(c) SC2D11 vs. SC2D01

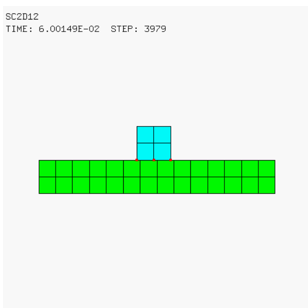
Figure 40: Results of test SC2D12.



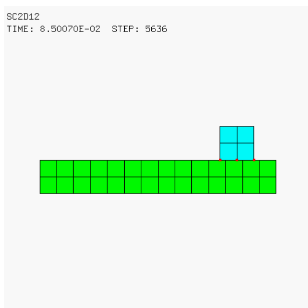
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

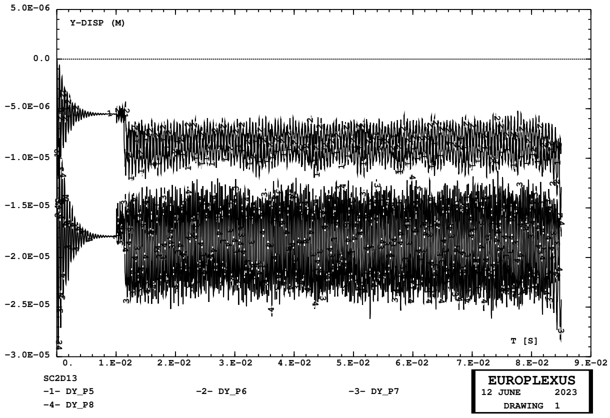


(d) $t = 85$ ms

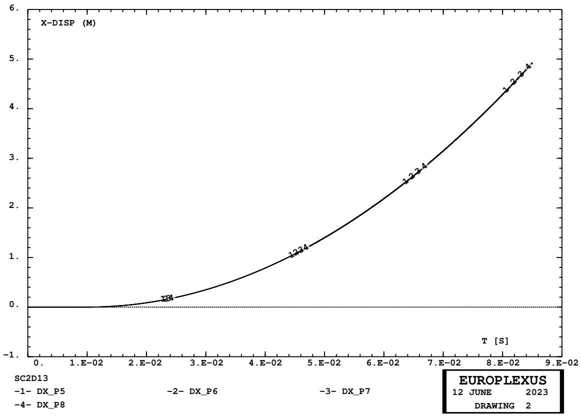
Figure 41: Deformed configuration and contact forces in test SC2D12.

7.2.14 Case SC2D13

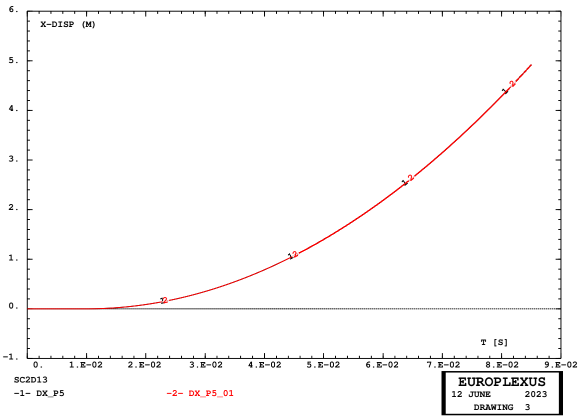
This test is similar to SC2D01 (no friction) but uses the SLID contact model instead of PINB. The solution is successful and visually identical to that of case SC2D01 as concerns the main (horizontal) displacement, see Figures 42 and 43. The vertical displacement is a bit noisy.



(a) Y-displacements

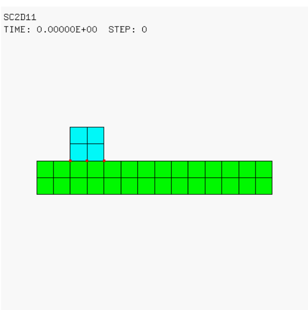


(b) X-displacements

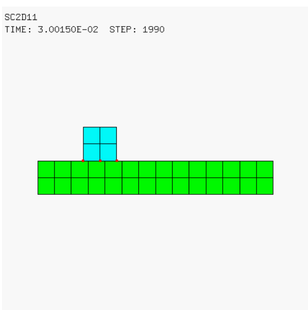


(c) SC2D13 vs. SC2D01

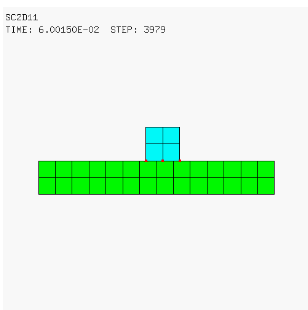
Figure 42: Results of test SC2D13.



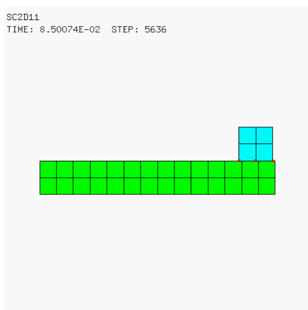
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

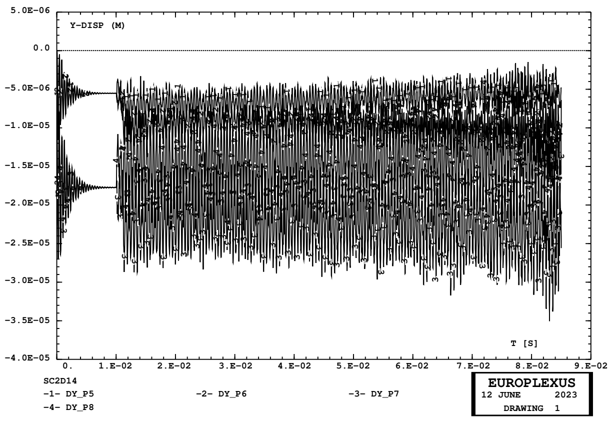


(d) $t = 85$ ms

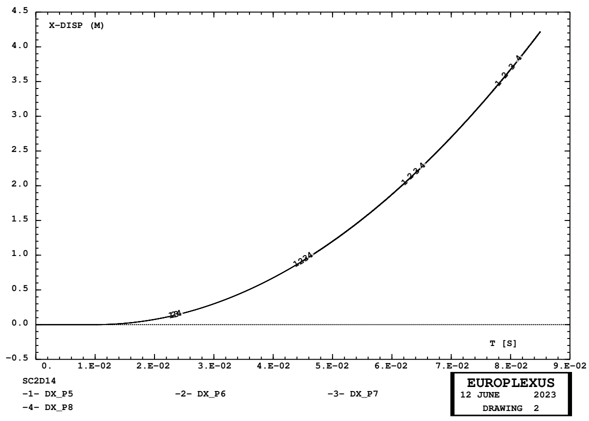
Figure 43: Deformed configuration and contact forces in test SC2D13.

7.2.15 Case SC2D14

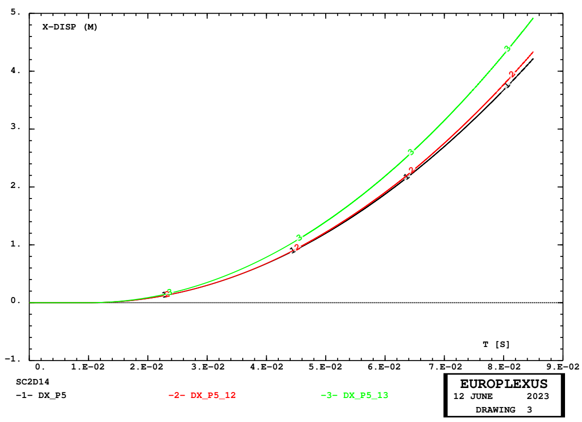
This test is similar to SC2D02 (with friction) but uses the SLID contact model instead of PINB. The solution is successful, see Figures 44 and 45. The horizontal displacement is less than that of case SC2D13, as expected, but not identical to that of case SC2D12 (GPIN). The vertical displacement is very noisy.



(a) Y-displacements

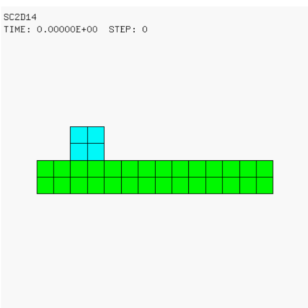


(b) X-displacements

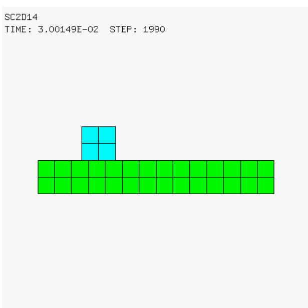


(c) SC2D14 vs. 12&13

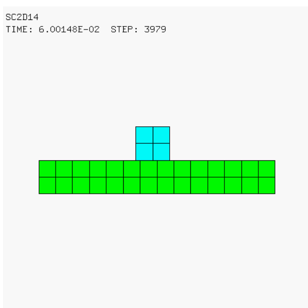
Figure 44: Results of test SC2D14.



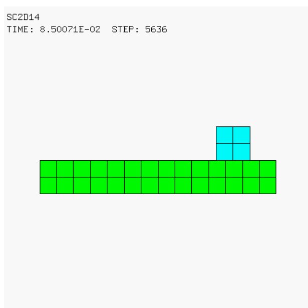
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



(d) $t = 85$ ms

Figure 45: Deformed configuration and contact forces in test SC2D14.

7.3 Sliding cube in 3D

We pass now to the 3D version of the test. The slider is a unit cube, while the plane is represented by a $7 \times 2 \times 1$ m parallelepiped. The initial problem geometry is shown in Figure 46.

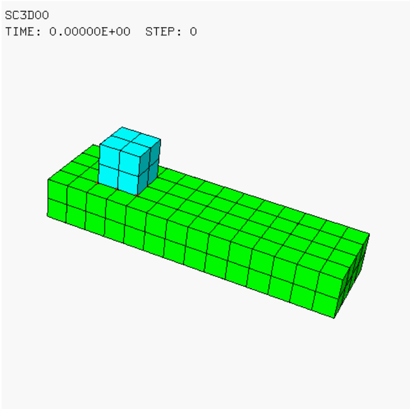


Figure 46: Definition of the sliding cube test in 3D.

The simulations performed are summarized in Table 5.

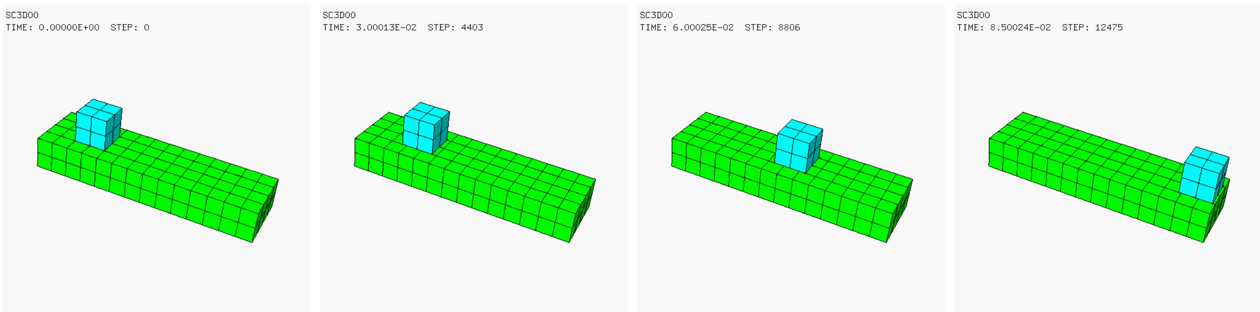
Test	Mesh	Description	t_{fin} [ms]	Steps	CPU [s]
SC3D00	120 C82L 9 PMAT 8 CL3D	Only horizontal pressure	85	12 475	7.9
SC3D01	120 C82L 9 PMAT 8 CL3D	Both pressures, no friction	85	12 482	14.4
SC3D02	120 C82L 9 PMAT 8 CL3D	Both pressures, with friction	85	12 575	11.6
SC3D03	120 CUB8 9 PMAT 8 CL3D	Idem 02 but CUB8 and VMIS ISOT	85	5 408	5.4
SC3D04	120 CUB8 9 PMAT 8 CL3D	Idem 03 but old exe (1-tangent friction)	85	5 378	6.3
SC3D05	120 C82L 9 PMAT 8 CL3D	Idem 01 (no friction) but penalty	85	12 476	10.8
SC3D06	120 C82L 9 PMAT 8 CL3D	Idem 02 (friction) but penalty	85	12 476	11.1
SC3D07	120 C82L 9 PMAT 8 CL3D	Idem 01 (no friction) but GLIS	85	12 475	10.2
SC3D08	120 C82L 9 PMAT 8 CL3D	Idem 02 (friction) but GLIS	85	12 475	10.2
SC3D09	10 C82L 4 PMAT 2 CL3D	Simplified version of 02	60	4 406	2.9
SC3D10	10 C82L 4 PMAT 2 CL3D	Idem 09 (friction) but GLIS	46	3 376	1.6
SC3D12	10 C82L 4 PMAT 1 CL3D	Punching problem, no friction, damping	85	6 238	3.8
SC3D13	10 C82L 4 PMAT 1 CL3D	Idem 12 but with friction	85	6 239	3.1
SC3D14	10 C82L 4 PMAT 1 CL3D	Idem 12 but GLIS	85	6 238	2.1
SC3D15	10 C82L 4 PMAT 1 CL3D	Idem 14 (GLIS) but with friction	85	6 238	2.3
SC3D16	10 C82L 8 PMAT 1 CL3D	Idem 12 but only pseudo-nodal pinballs	85	6 238	2.1
SC3D17	10 C82L 8 PMAT 1 CL3D	Idem 16 but with friction	85	6 238	2.2
SC3D18	10 C82L 1 CL3D	Idem 14 (no friction) but GPIN	85	6 238	2.3
SC3D19	10 C82L 1 CL3D	Idem 15 (with friction) but GPIN	85	6 238	2.3
SC3D20	120 C82L 8 CL3D	Idem 07 (no friction) but GPIN	85	12 475	11.6
SC3D21	120 C82L 8 CL3D	Idem 08 (friction) but GPIN	85	12 475	12.5
SC3D22	120 C82L 8 CL3D	Idem 14 (no friction) but SLID	85	6 238	2.3
SC3D23	120 C82L 8 CL3D	Idem 15 (friction) but SLID	85	6 238	2.3
SC3D24	120 C82L 8 CL3D	Idem 20 (no friction) but SLID	85	12 475	9.1
SC3D25	120 C82L 8 CL3D	Idem 21 (friction) but SLID	85	12 475	9.9

Table 5: Sliding cube tests in 3D.

7.3.1 Case SC3D00

This is the 3D version of test SC3D00. Only the horizontal pressure is applied, starting at $t_{\text{QS}} = 10$ ms. Until that time, the same quasi-static damping as in the 2D cases is applied (although this is irrelevant in the present simulation).

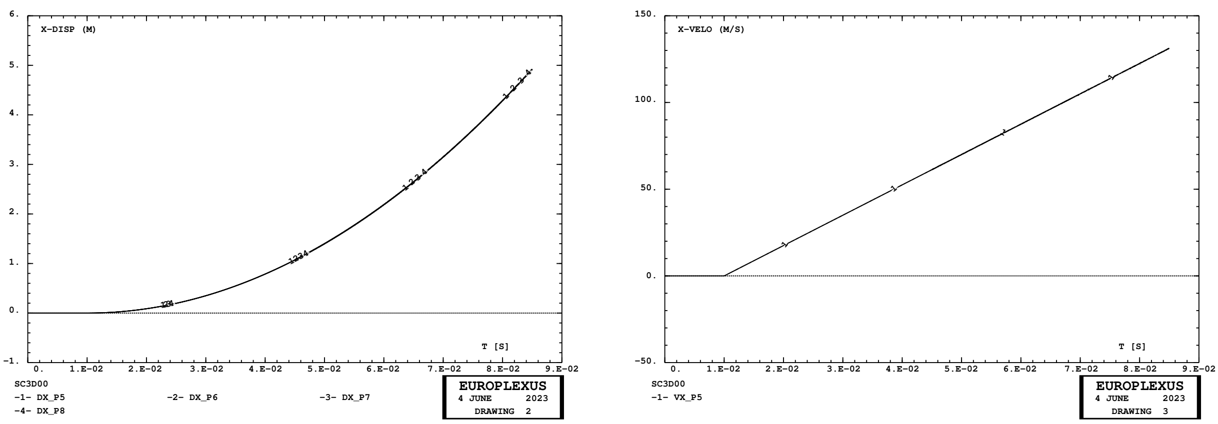
Figure 47 shows the accelerated cube motion by a sequence of deformed mesh images at some characteristic times.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 47: Motion of the cube in test SC3D00.

Figure 48 shows the X-displacement and the X-velocity histories. The velocity is zero until $t_{\text{QS}} = 10$ ms, then it ramps up linearly under constant acceleration, due to the constant applied pressure. These curves can be used as references for comparison of the subsequent simulations, where contact (first without and then with friction) will be added.



(a) X-displacements (b) X-velocities

Figure 48: Further results of test SC3D00.

The curves in Figure 48 are identical to those in Figure 19 for the 2D case SC2D00, as expected. See Figure 49 for a direct comparison, where the red curve is the 2D solution (SC2D00) and the black curve is the 3D solution (SC3D00).

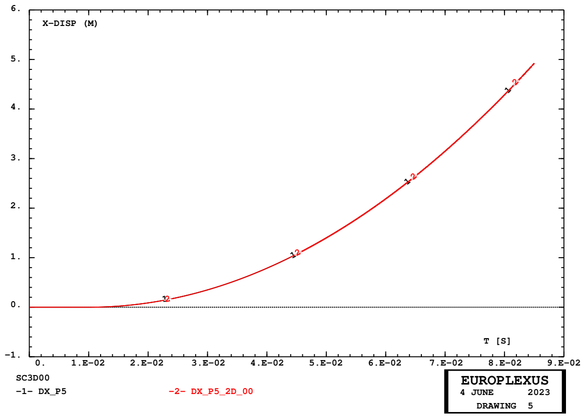


Figure 49: Comparison of horizontal displacements in test SC2D00 and SC3D00.

7.3.2 Case SC3D01

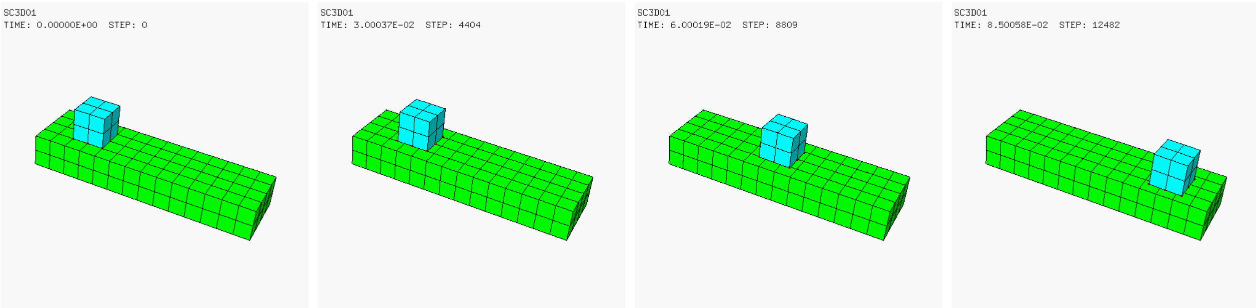
This test is the 3D version of test SC2D01. It is similar to SC3D00 but we add also the vertical pressure and the contact (without friction). The quasi-static damping is prescribed (until $t_{QS} = 10$ ms) in this simulation, like in the previous ones.

Upon first running, the test stopped with the following error message:

```
** ERROR  1 IN THE ROUTINE GEN_CUB8 ** THE MESSAGE IS THE FOLLOWING ONE :
INVALID EL OF K20TYP 14
```

The problem is readily fixed by adding the treatment of elements of type C81L (155) and C82L (156) in the subroutine GEN_CUB8 of module M_PINBALLS_SPLIT. The correction is evolved.

Figure 50 shows the accelerated cube motion by a sequence of deformed mesh images at some characteristic times. The solution is bad. One can see that the cube displaces less than expected, and follows a slightly bent trajectory instead of a straight one.

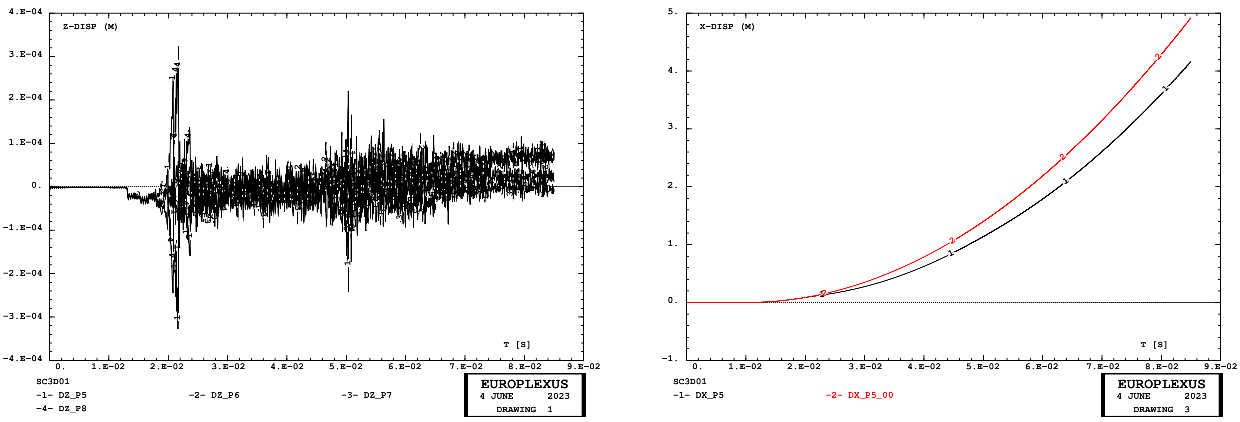


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 50: Motion of the cube in test SC3D01.

These observations are confirmed by the time curves shown in Figure 51, which shows the Z-displacement (vertical) and the X-displacement (horizontal) histories, the latter compared with solution SC3D00. The vertical displacement is very noisy, and unexpectedly becomes greater than zero towards the end of the test.

The horizontal displacement (black curve in Figure 51(b)) has a parabolic shape but is in very poor agreement with the solution SC3D00 (red curve).

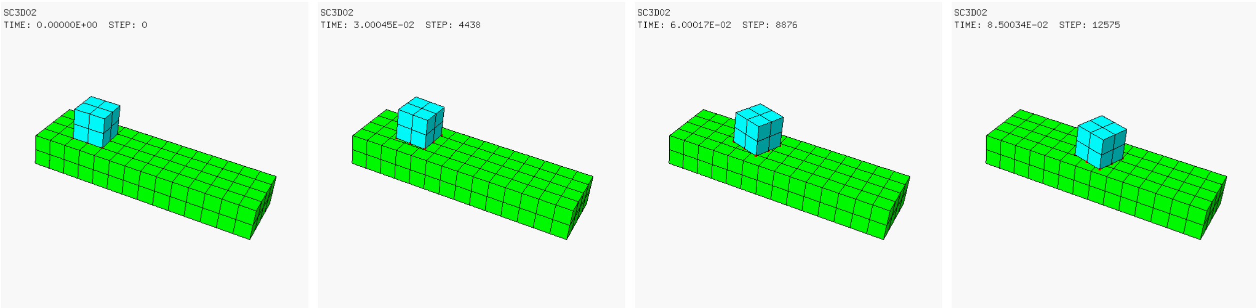


(a) Z-displacements (b) X-displacements

Figure 51: Further results of test SC3D01.

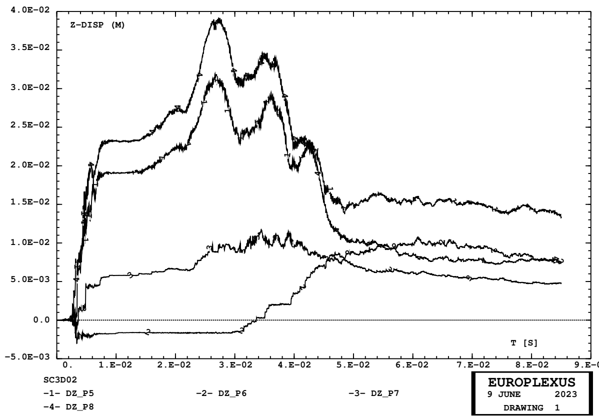
7.3.3 Case SC3D02

This test is similar to SC3D01 but we add friction. It is the 3D version of test SC2D02. The simulation runs until the final time, but the results are weird. The cube follows a curved and non-natural trajectory, see Figures 52 and 53. Also, the displacement seems underestimated, see Figure 53(c).

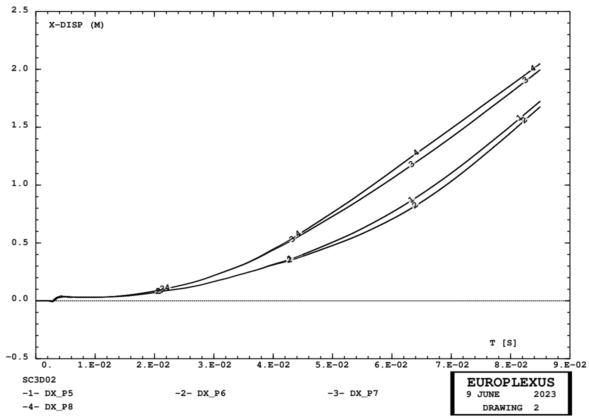


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

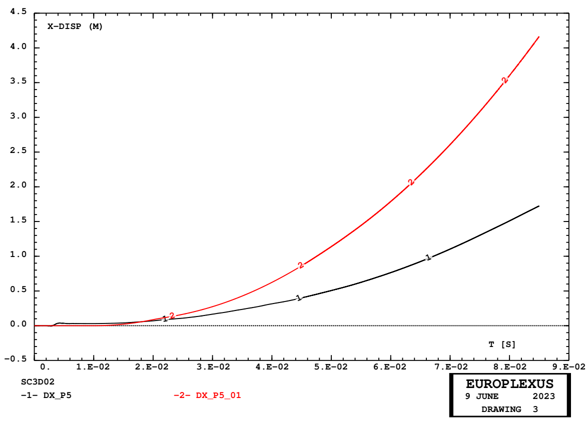
Figure 52: Motion of the cube in test SC3D02.



(a) Z-displacements



(b) X-displacements



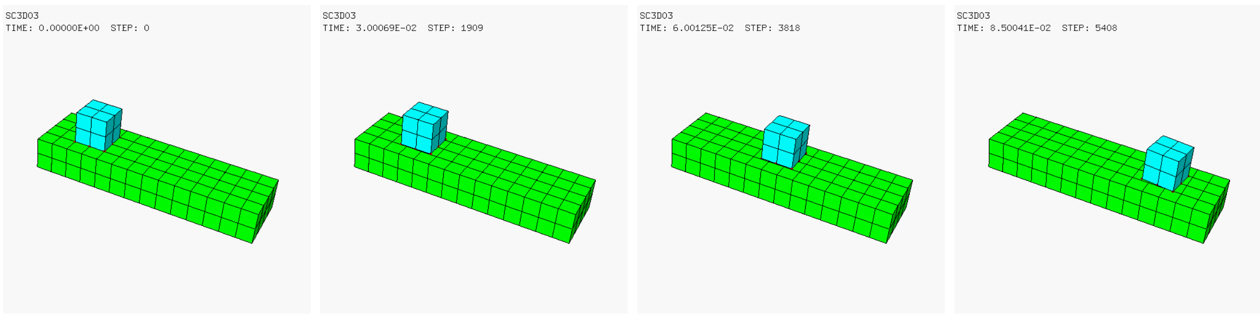
(c) Compared with 01

Figure 53: Further results of test SC3D02.

7.3.4 Case SC3D03

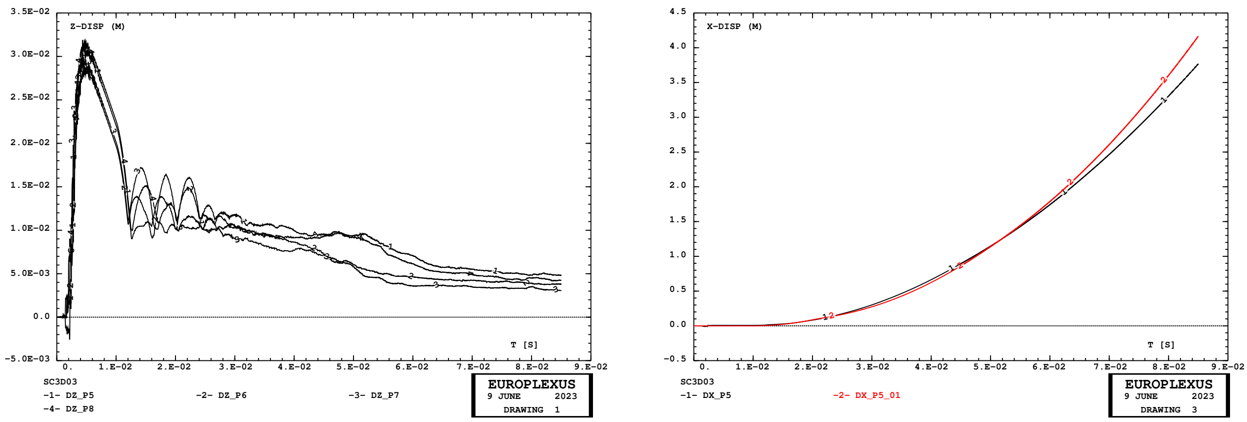
We want to check whether the disappointing results obtained in test SC3D02 are due to the two-tangent friction algorithm recently implemented in the LM-based PINB contact model. To this end, one should run the test with the old EPX executable. However, in case SC3D01 we have seen that a correction in the code is needed when using the C82L (or C81L) element and such correction cannot be implemented in the old executable. Therefore, we first modify the test SC23D02 by replacing the C82L element by CUB8 and the VM23 material by VMIS ISOT (CEA's equivalent material).

By running the test with the current executable (two-tangent friction model), it completes the simulation until the final time. However, the solution is weird. The cube initially moves along the positive Z direction, which is hard to explain, see Figures 54 and 55. Then, towards the end of the simulation, the trajectory bends slightly. The final displacement is slightly less than the one without friction, as expected, but the result is not convincing.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 54: Motion of the cube in test SC3D03.



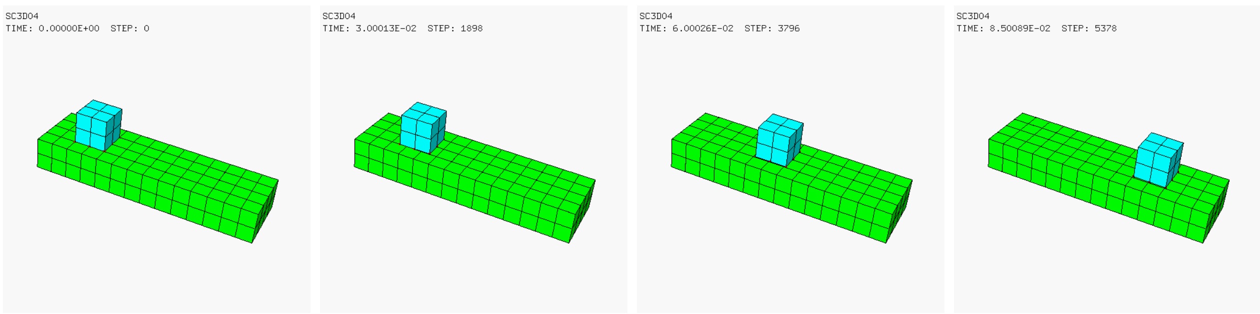
(a) Z-displacements (b) X-displacements

Figure 55: Further results of test SC3D03.

7.3.5 Case SC3D04

This test is strictly identical to SC3D03 but it is run with the old EPX executable, which used the 1-tangent friction model instead of the 2-tangent model.

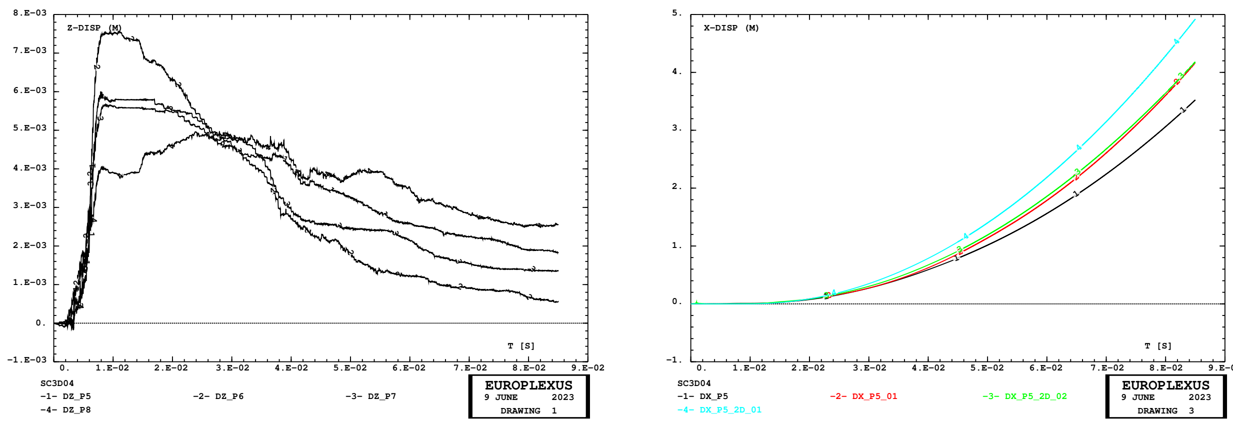
The solution is computed until the final time, and looks decent, but not totally convincing, see Figures 56 and 57.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 56: Motion of the cube in test SC3D04.

In Figure 57(b) the X-displacements are compared between the present solution (in black), the case SC3D01 (without friction, red curve), the 2D case SC2D02 (2D with friction, green curve), and the 2D case SC2D01 (2D without friction, cyan curve).



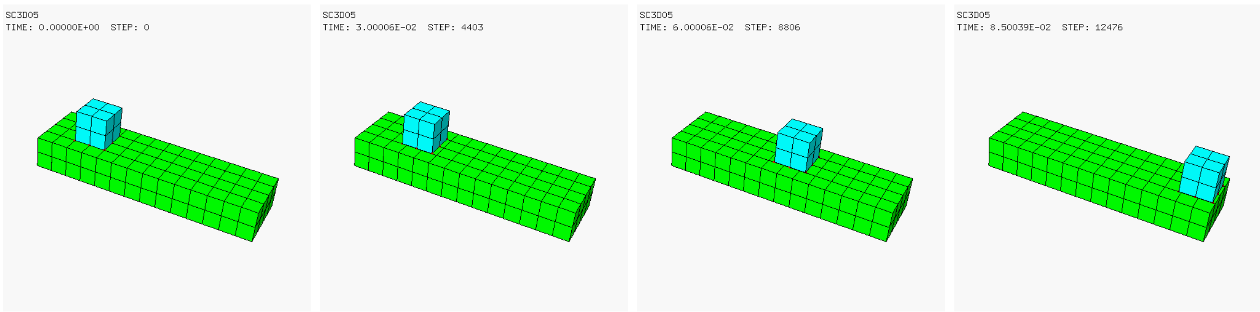
(a) Z-displacements (b) X-displacements

Figure 57: Further results of test SC3D04.

7.3.6 Case SC3D05

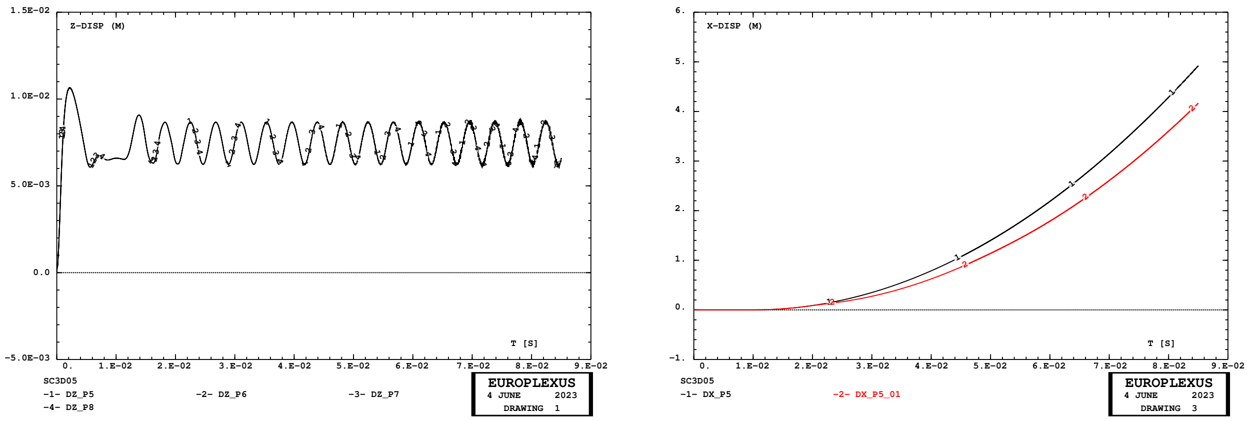
This test is identical to SC3D01 (no friction) but it uses the penalty formulation of the pinball contact model (`LINK DECO PINB PENA`).

The solution looks correct, see Figures 58 and 59. The red curve in Figure 59(b) is the solution with LM (SC3D01) which, as already noted, predicts too small a displacement, while the black curve (present solution with penalty) looks correct.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 58: Motion of the cube in test SC3D05.



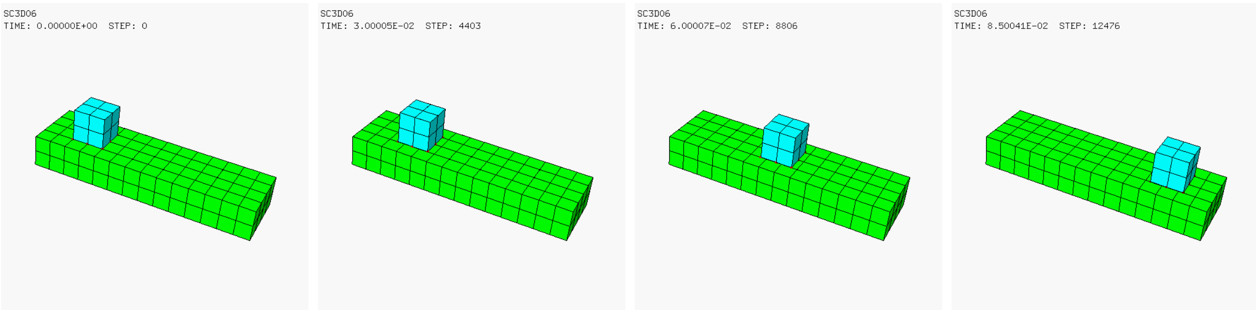
(a) Z-displacements (b) X-displacements

Figure 59: Further results of test SC3D05.

7.3.7 Case SC3D06

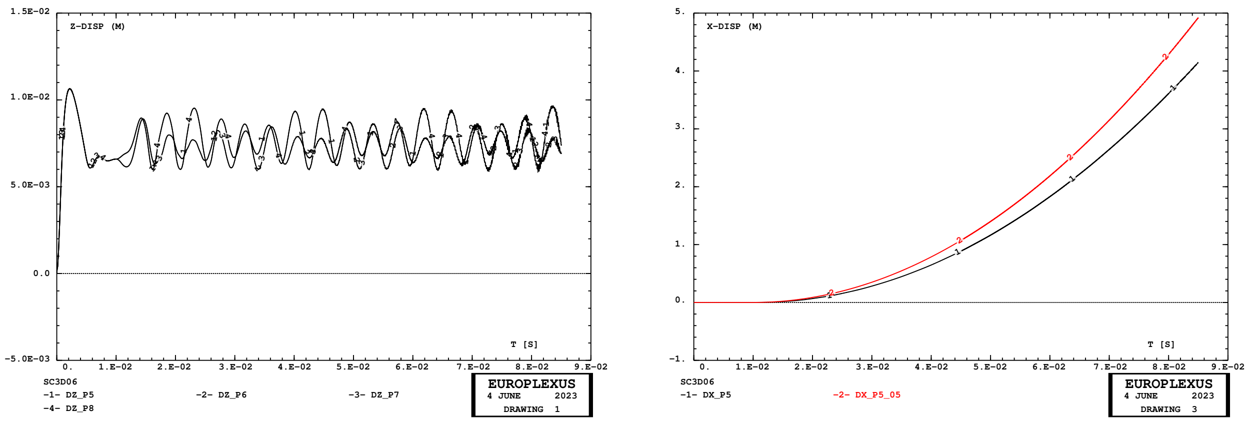
This test is identical to SC3D02 (with friction) but it uses the penalty formulation of the pinball contact model (`LINK DECO PINB PENA`).

The solution is correct, see Figures 60 and 61. The horizontal displacement curve (black curve) is lower than the corresponding one without friction (red curve), as expected.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 60: Motion of the cube in test SC3D06



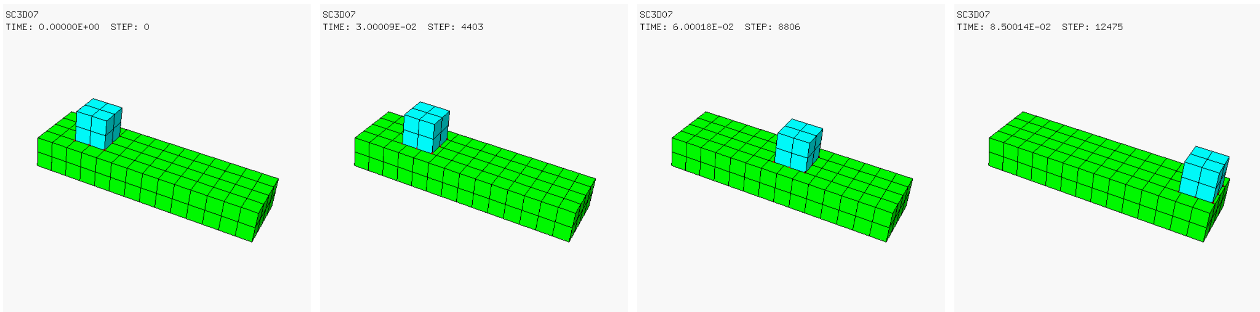
(a) Z-displacements (b) X-displacements

Figure 61: Further results of test SC3D06.

7.3.8 Case SC3D07

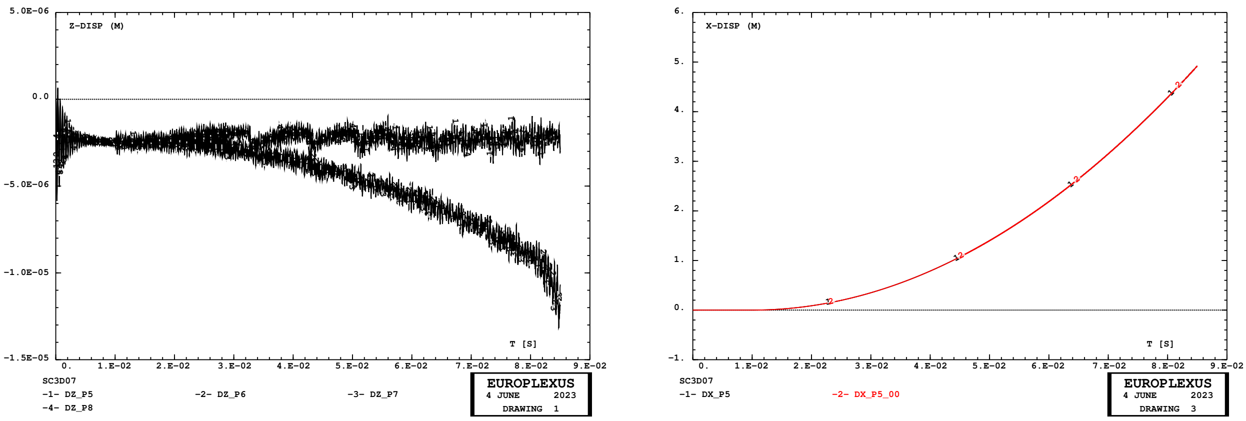
This test is identical to SC3D01 (no friction) but it uses the GLIS contact model.

The solution looks correct, see Figures 62 and 63. The red curve in Figure 63(b) is the solution with LM-PINB and only horizontal pressure (SC3D00).



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 62: Motion of the cube in test SC3D07.



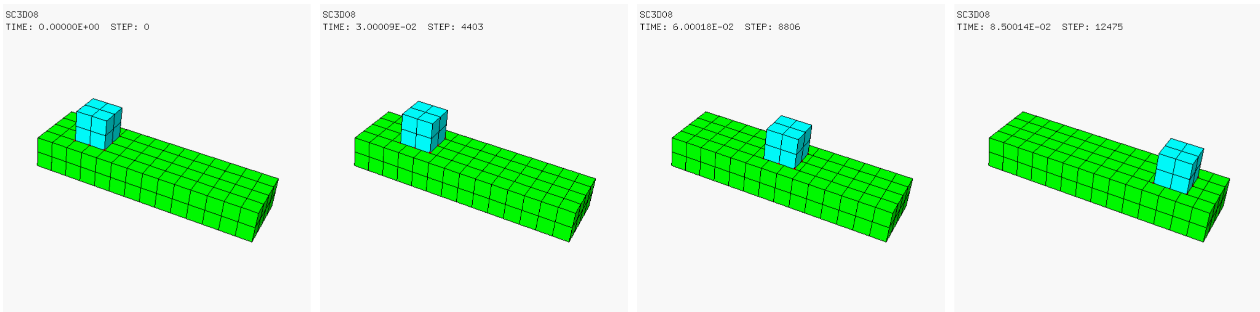
(a) Z-displacements (b) X-displacements

Figure 63: Further results of test SC3D07.

7.3.9 Case SC3D08

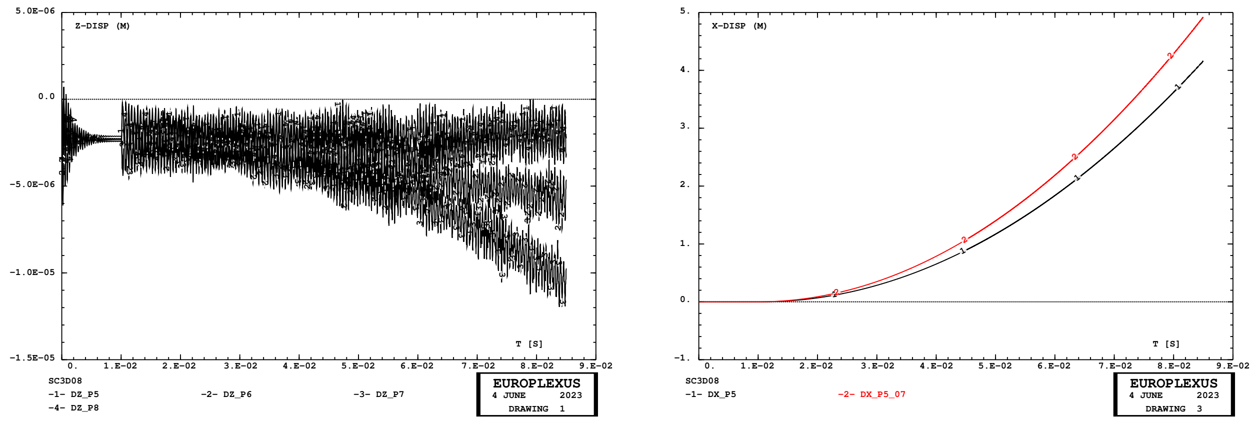
This test is identical to SC3D02 (with friction) but it uses the GLIS contact model.

The solution looks correct, see Figures 64 and 65. The horizontal displacement curve (black curve) is lower than the corresponding one without friction (red curve), as expected.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 64: Motion of the cube in test SC3D08



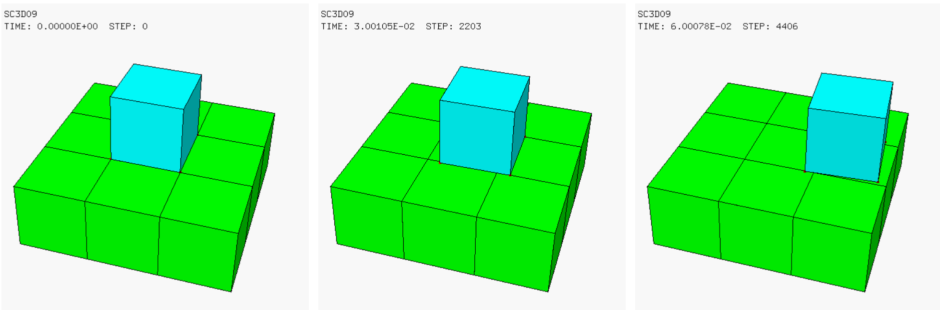
(a) Z-displacements (b) X-displacements

Figure 65: Further results of test SC3D08.

7.3.10 Case SC3D09

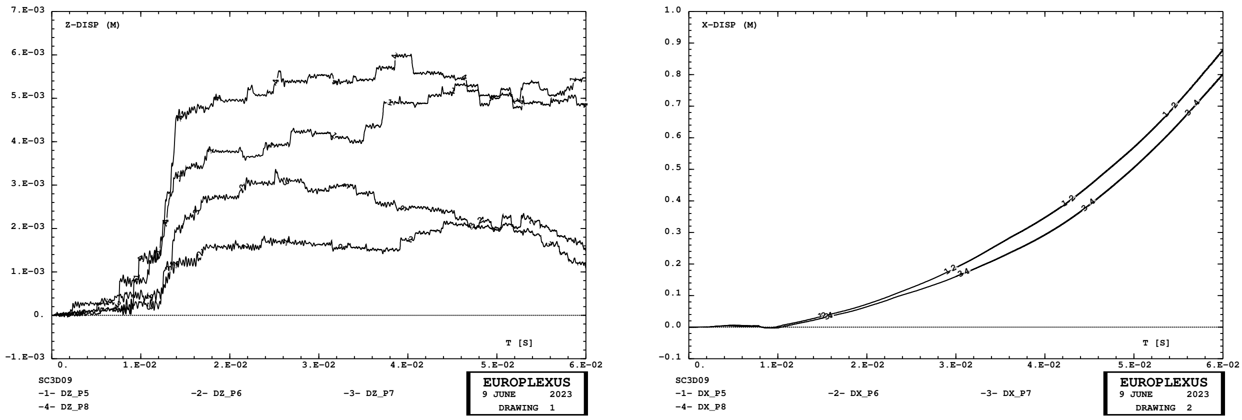
This is a simplified version of test SC3D02 (with friction). The plane is meshed by only 9 cubes and the cube by just one element, see Figure 66(a). The scope of reducing the problem is to facilitate the debugging.

The solution is not correct, the trajectory of the cube is not straight, see Figures 66 and 67.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms

Figure 66: Motion of the cube in test SC3D09

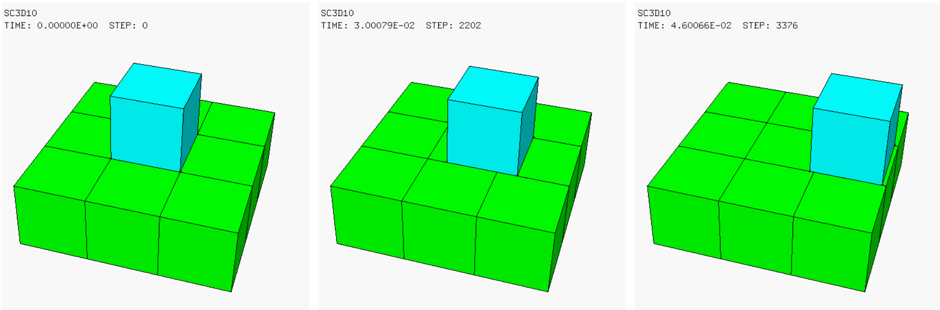


(a) Z-displacements (b) X-displacements

Figure 67: Further results of test SC3D09.

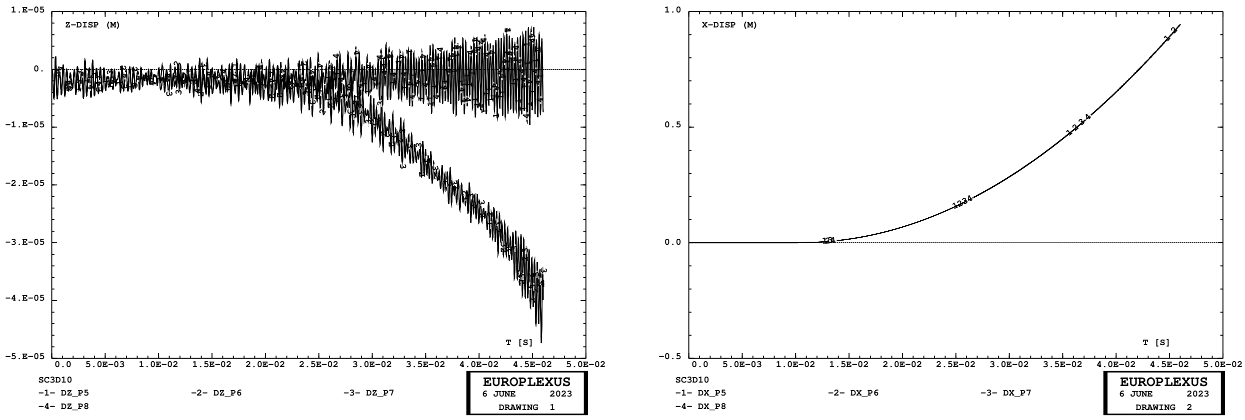
7.3.11 Case SC3D10

This test is similar to SC3D09 but uses GLIS (with friction). The solution looks correct, see Figures 68 and 69. The cube motion is much faster than in the solution SC3D09.



(a) $t = 0$ (b) $t = 30$ ms (c) $t = 46$ ms

Figure 68: Motion of the cube in test SC3D10



(a) Z-displacements (b) X-displacements

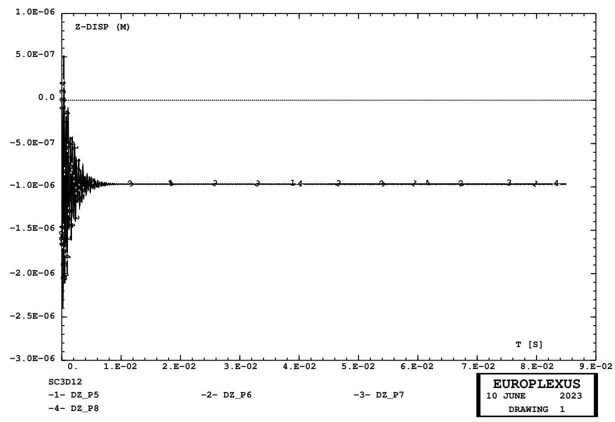
Figure 69: Further results of test SC3D10.

7.3.12 Case SC3D12

This is the 3D version of the 2D punching problem SC2D07. A cube is pushed against a plane, but there is no horizontal pushing force. We assume no friction but, unlike case SC2D07, quasi-static damping is applied until $t_{QS} = 10$ ms.

Scope of the simulation is to check stability and symmetry of the solution, and to verify that the resultants of the contact forces (reactions) in the two contacting bodies are equal and opposite.

All the above requirements are verified. The solution is initially noisy but then becomes smooth thanks to the damping, see Figure 70.

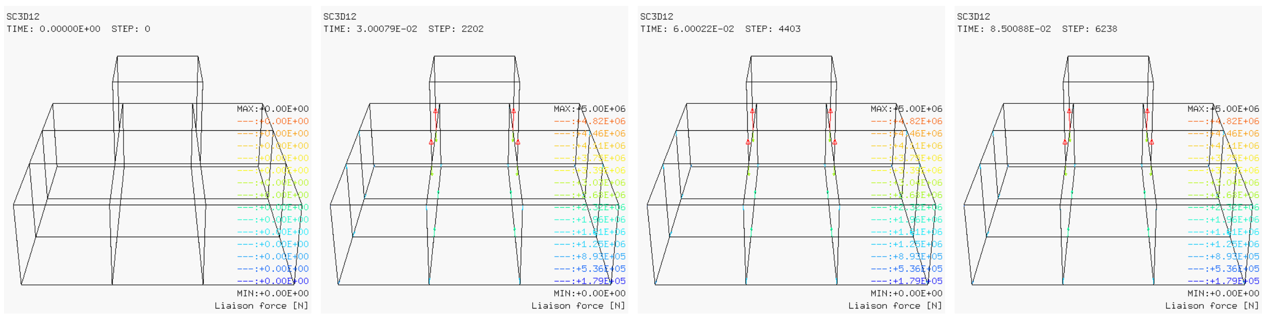


(a) Z-displacements

(b) X-displacements

Figure 70: Some results of test SC3D12.

From the animation, by visualizing the contact forces, we see that what we obtain, after an initial transient, is the expected static solution, see Figure 71. The reactions are vertical, as expected, since there is no friction.



(a) $t = 0$

(b) $t = 30$ ms

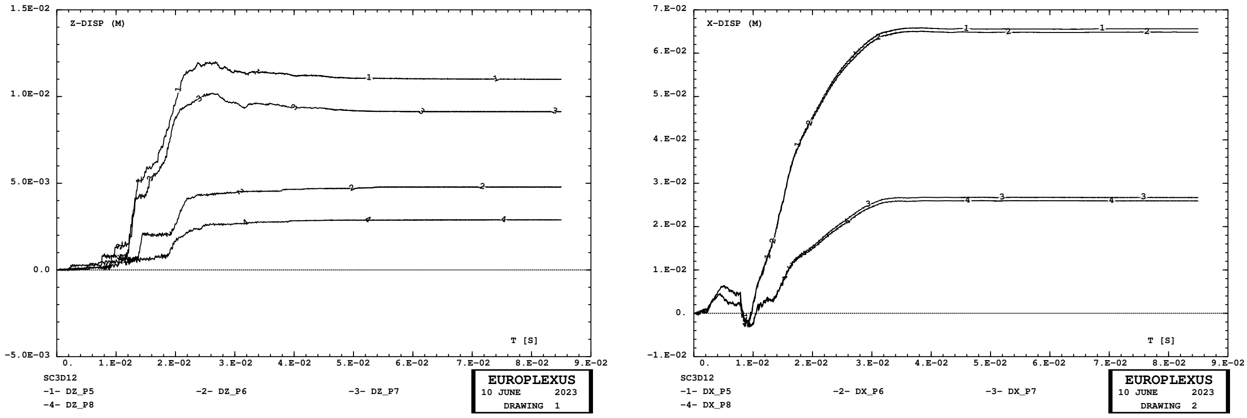
(c) $t = 60$ ms

(d) $t = 85$ ms

Figure 71: Deformed configuration and contact forces in test SC3D12.

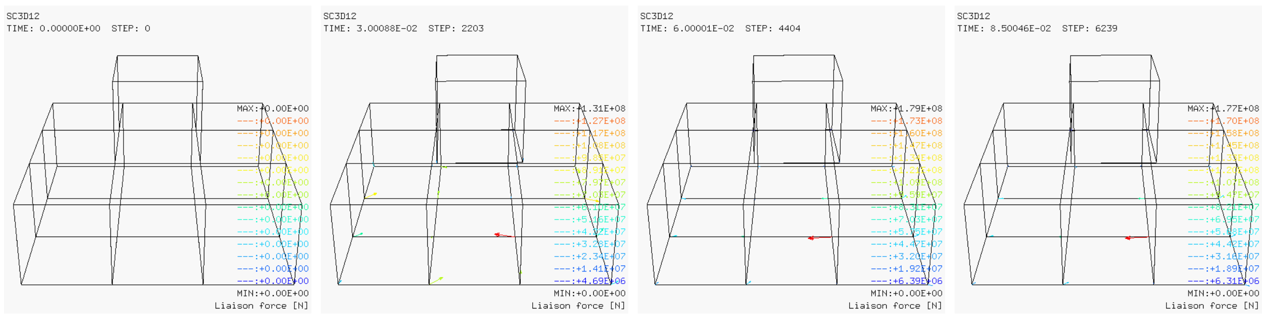
7.3.13 Case SC3D13

This test is equal to SC3D12 but we add friction. The cube undergoes a weird motion, so the solution is incorrect, see Figures 72 and 73. The reactions look particularly weird.



(a) Z-displacements (b) X-displacements

Figure 72: Some results of test SC3D13.

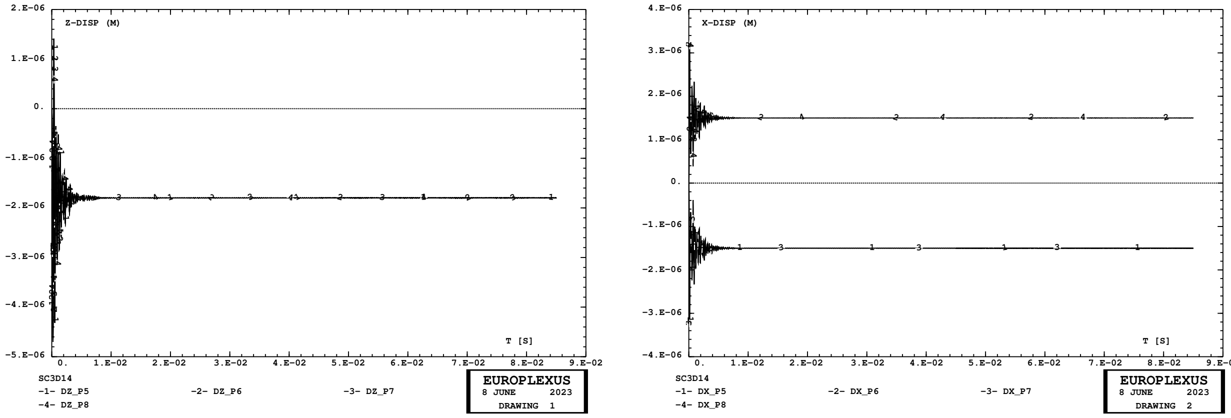


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 73: Deformed configuration and contact forces in test SC3D13.

7.3.14 Case SC3D14

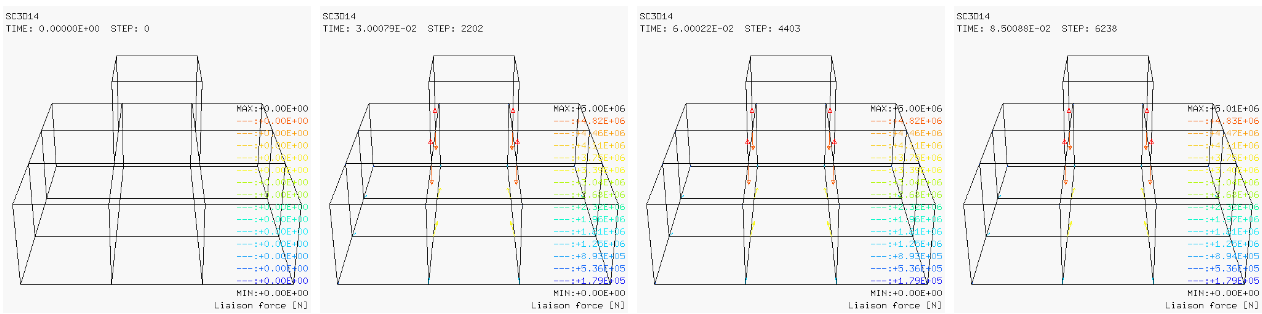
This test is identical to SC3D12 (no friction) but uses the GLIS contact model. The solution looks correct, see Figures 74 and 75. The static solution is recovered thanks to the initial damping. The reactions are perfectly vertical since there is no friction.



(a) Z-displacements

(b) X-displacements

Figure 74: Some results of test SC3D14.



(a) $t = 0$

(b) $t = 30$ ms

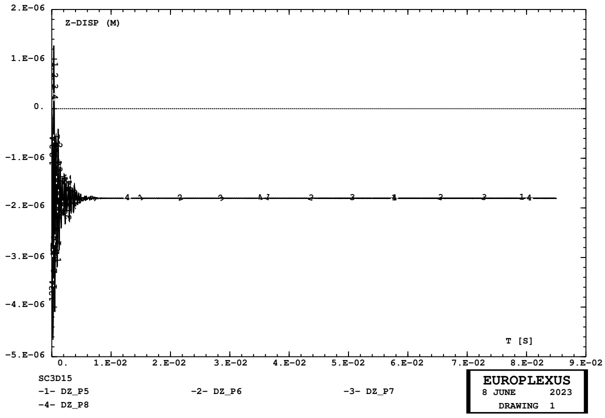
(c) $t = 60$ ms

(d) $t = 85$ ms

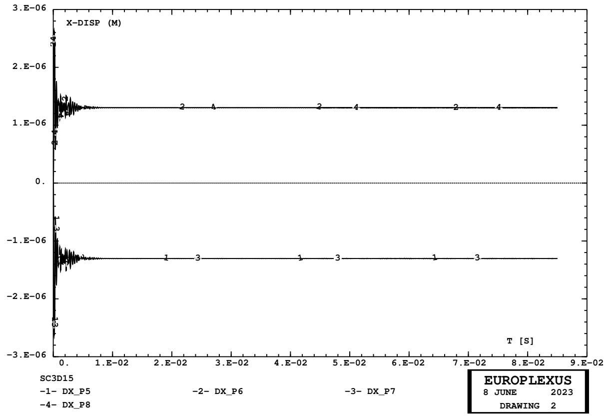
Figure 75: Deformed configuration and contact forces in test SC3D14.

7.3.15 Case SC3D15

This test is identical to SC3D14 (GLIS) but we add friction. The solution looks correct, see Figures 76 and 77. The static solution is recovered thanks to the initial damping. The reactions are slightly inclined because of friction, but they remain within Coulomb's cone.

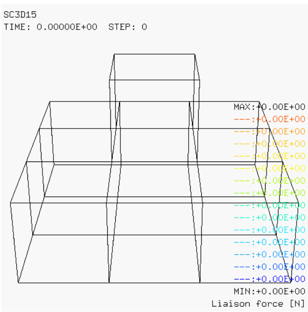


(a) Z-displacements

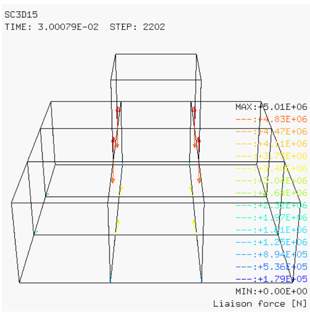


(b) X-displacements

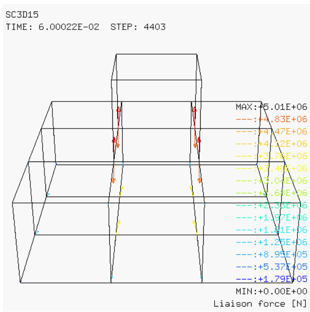
Figure 76: Some results of test SC3D15.



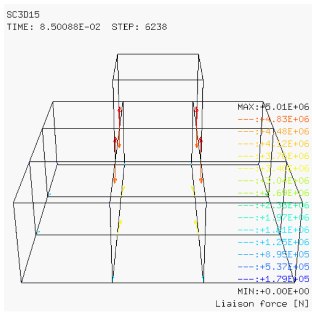
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



(d) $t = 85$ ms

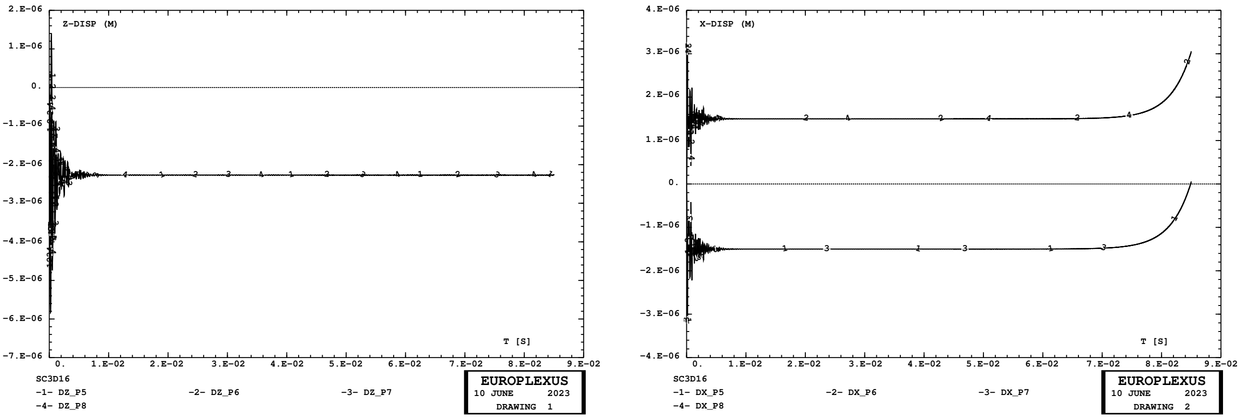
Figure 77: Deformed configuration and contact forces in test SC3D15.

7.3.16 Case SC3D16

The scope of this test is to verify whether the weird results obtained in some of the previous solutions could be due to the presence of multiple contacts. To this end, pseudo-nodal pinballs are embedded ot only in the cube's lower face nodes, but also in the plane's upper face nodes, thus obtaining a node-to-node contact. Of course, this setup is only academic because it may only work as long as the nodes of the contacting bodies match, and there is no (or very little, due to Poisson effect) tangential displacement.

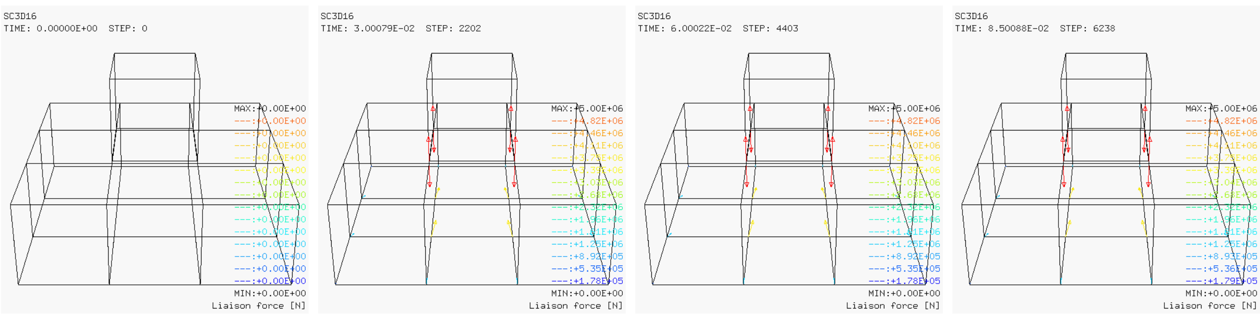
To make the simulation work, the cube must be raised slightly vertically with respect to the plane, so that contact normals can be computed and such that contact is not lost after the small Poisson displacements.

The solution looks substantially correct, see Figures 78 and 79, which was expected since there is no friction. There is only a very slight horizontal translation of the cube towards the end of the solution, see Figure 78(b).



(a) Z-displacements (b) X-displacements

Figure 78: Some results of test SC3D16.



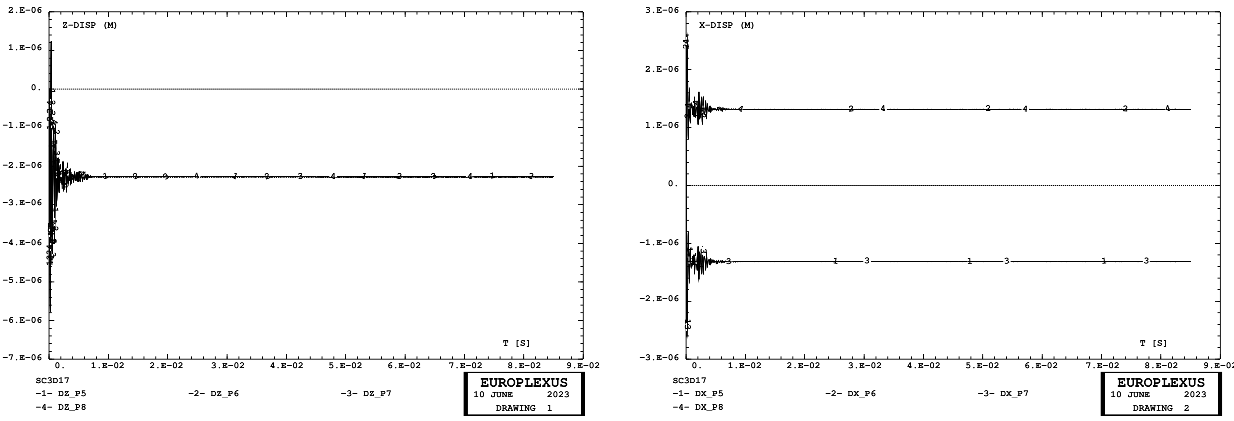
(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 79: Deformed configuration and contact forces in test SC3D16.

7.3.17 Case SC3D17

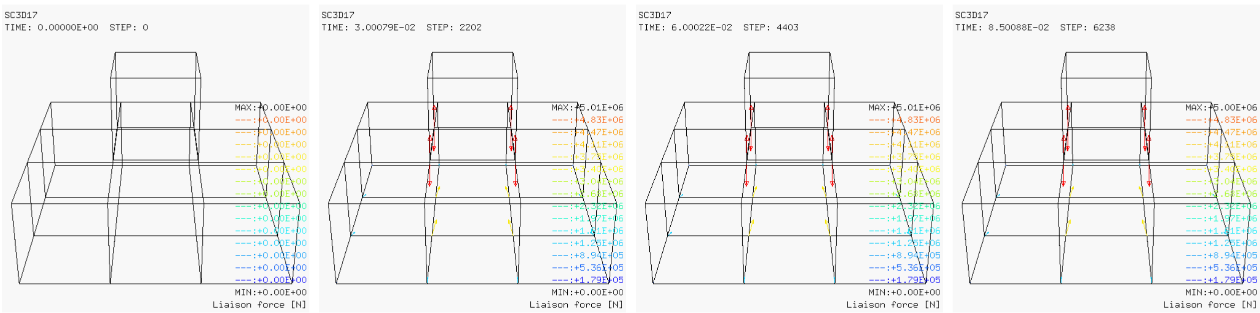
This test is identical to SC3D16 but we add friction.

The solution looks correct, see Figures 80 and 81, which was not the case in previous tests with multiple contacts with friction. There is no horizontal translation of the cube towards the end of the solution, probably prevented by the friction.



(a) Z-displacements (b) X-displacements

Figure 80: Some results of test SC3D17.

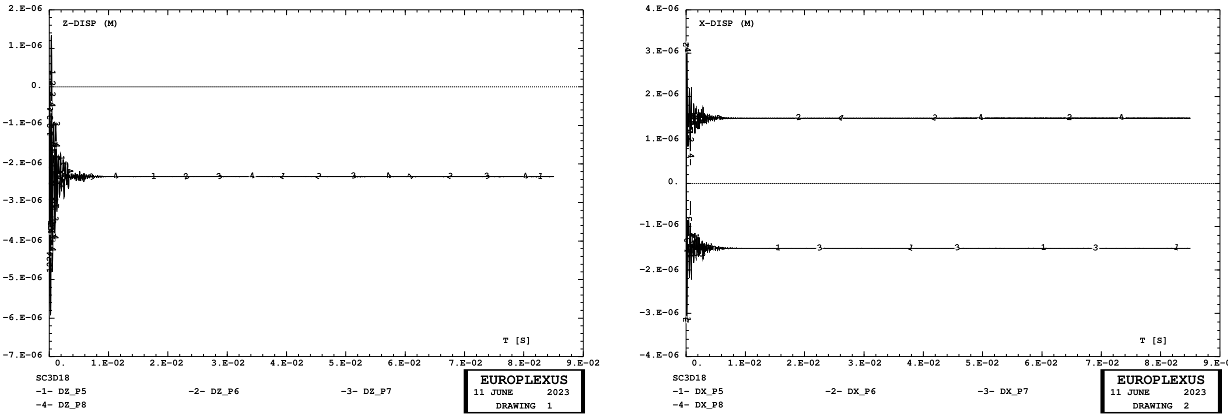


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 81: Deformed configuration and contact forces in test SC3D17.

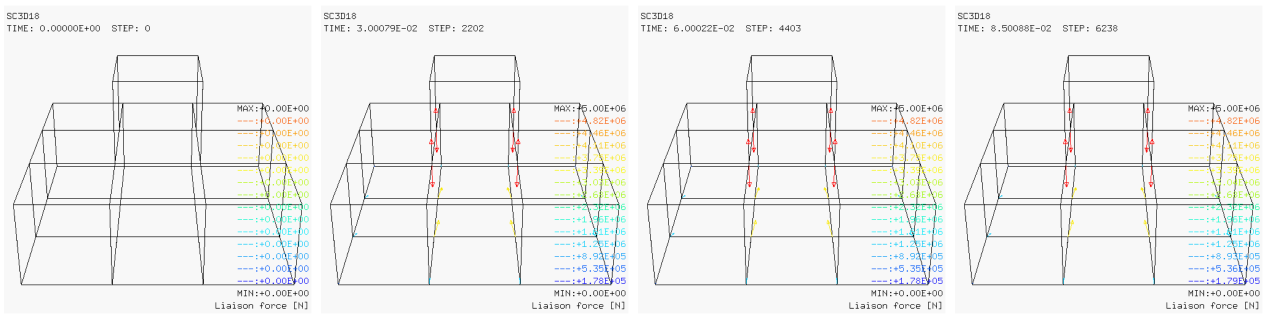
7.3.18 Case SC3D18

This test is similar to SC3D14 (no friction) but uses the GPIN contact model instead of GLIS. The solution looks correct, see Figures 82 and 83.



(a) Z-displacements (b) X-displacements

Figure 82: Some results of test SC3D18.

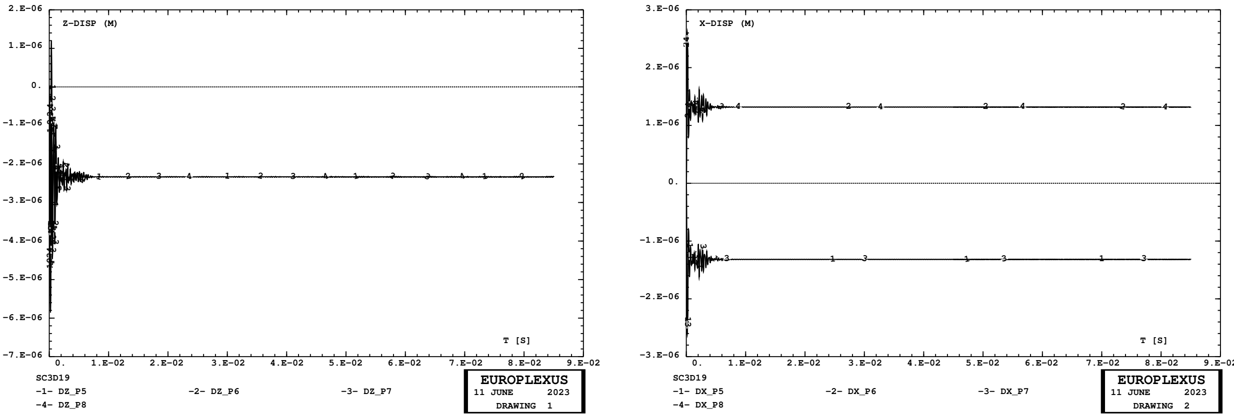


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

Figure 83: Deformed configuration and contact forces in test SC3D18.

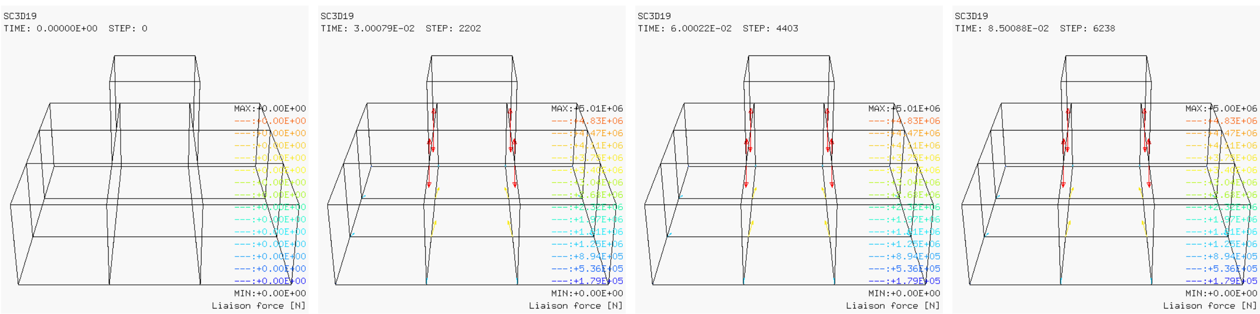
7.3.19 Case SC3D19

This test is similar to SC3D15 (with friction) but uses the GPIN contact model instead of GLIS. The solution looks correct, see Figures 84 and 85.



(a) Z-displacements (b) X-displacements

Figure 84: Some results of test SC3D19.

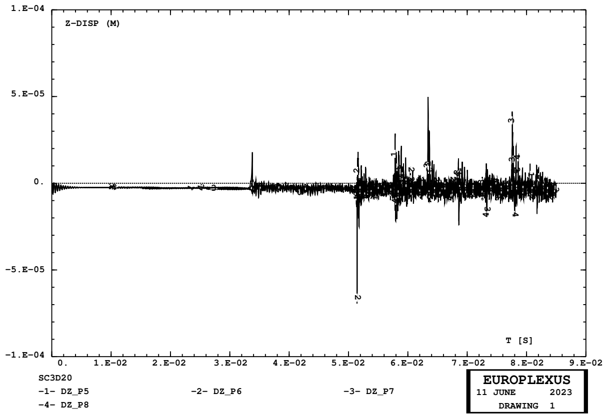


(a) $t = 0$ (b) $t = 30$ ms (c) $t = 60$ ms (d) $t = 85$ ms

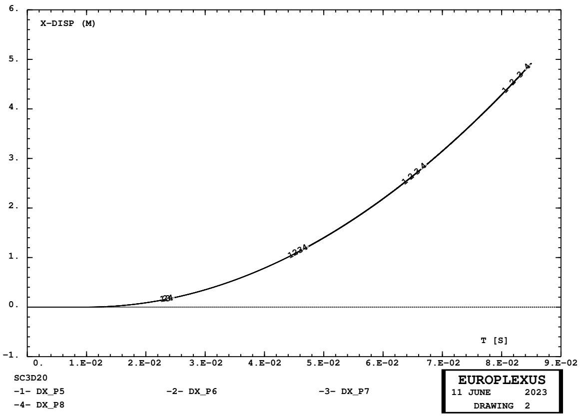
Figure 85: Deformed configuration and contact forces in test SC3D19.

7.3.20 Case SC3D20

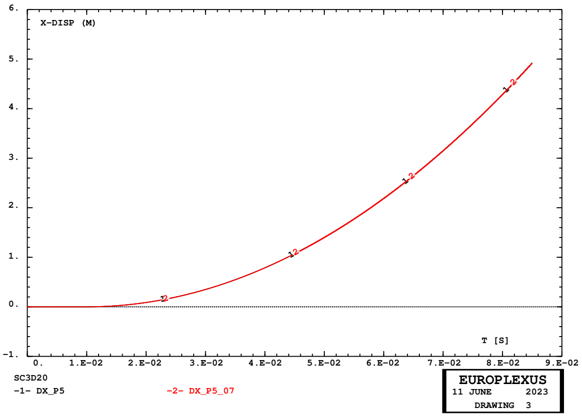
This test is similar to SC3D07 (without friction) but uses the GPIN contact model instead of GLIS. The solution looks correct, see Figures 86 and 87.



(a) Z-displacements

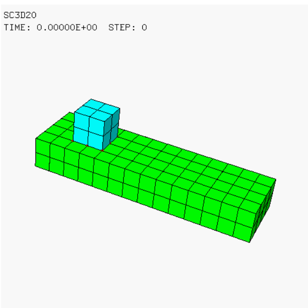


(b) X-displacements

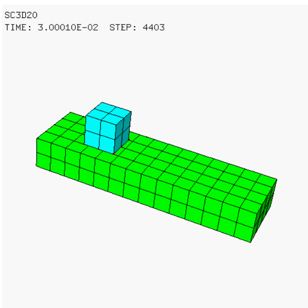


(c) SC3D20 vs. SC3D07

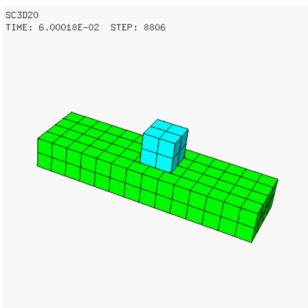
Figure 86: Some results of test SC3D20.



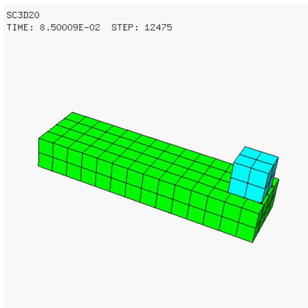
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

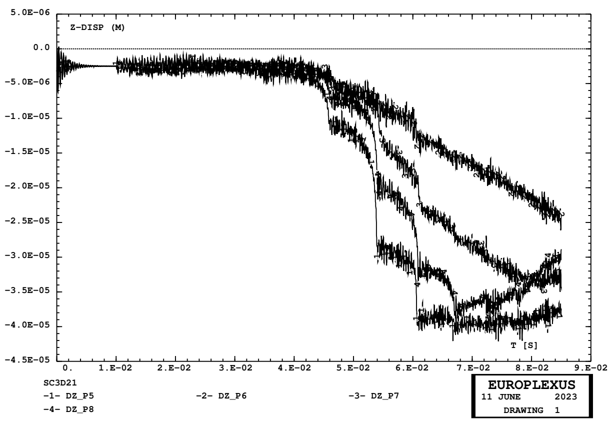


(d) $t = 85$ ms

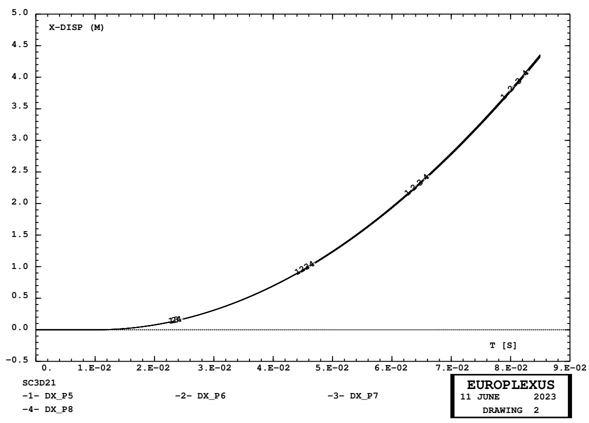
Figure 87: Deformed configuration and contact forces in test SC3D20.

7.3.21 Case SC3D21

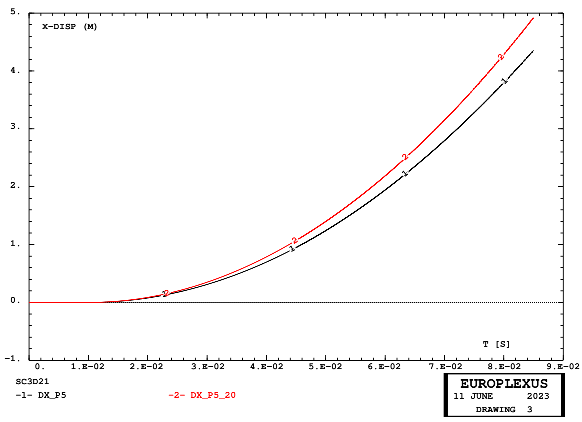
This test is similar to SC3D08 (with friction) but uses the GPIN contact model instead of GLIS. The solution looks correct, see Figures 88 and 89. The displacement is slightly higher than in the corresponding solution with GLIS (SC3D08), but lower than that of case SC3D20 (without friction), as expected.



(a) Z-displacements

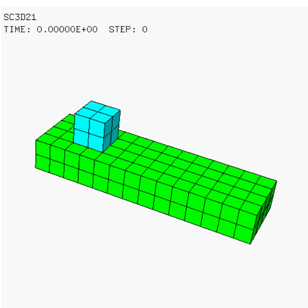


(b) X-displacements

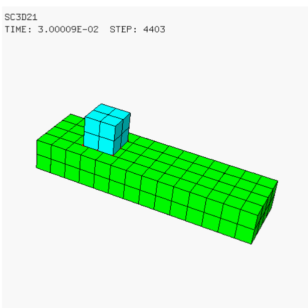


(c) SC3D21 vs. SC3D20

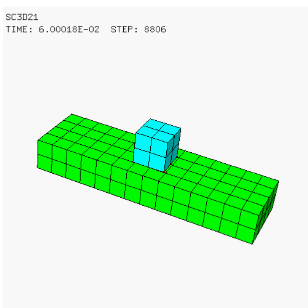
Figure 88: Some results of test SC3D21.



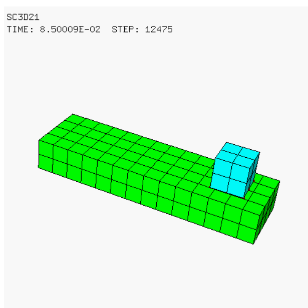
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

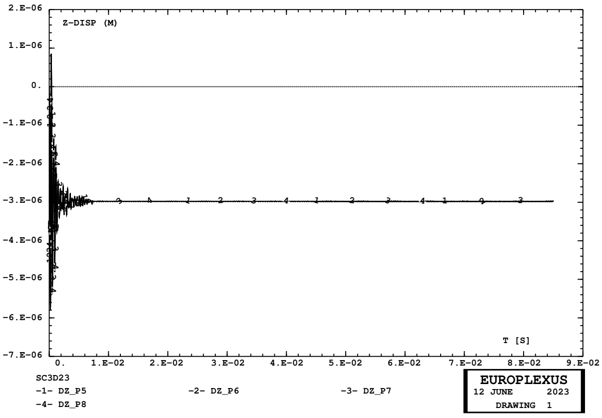


(d) $t = 85$ ms

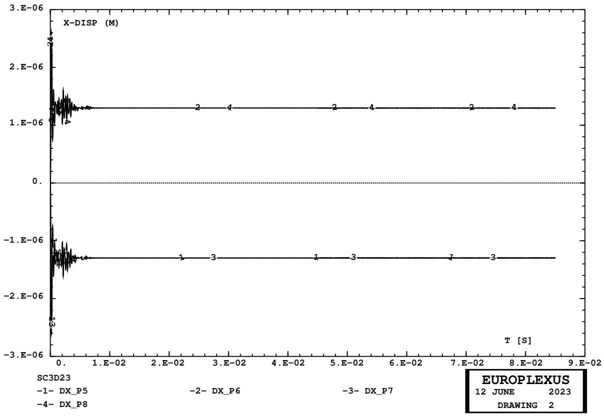
Figure 89: Deformed configuration and contact forces in test SC3D21.

7.3.22 Case SC3D23

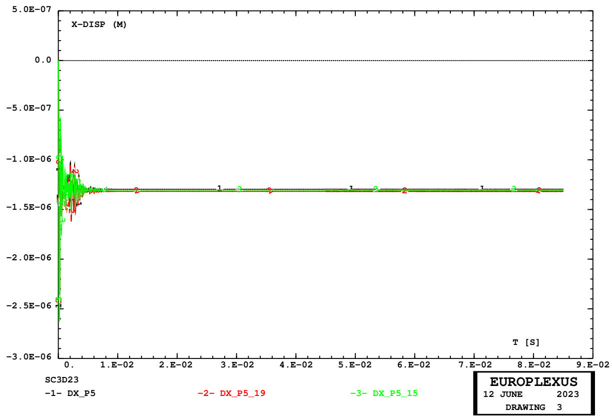
This test is similar to SC3D15 (with friction) but uses the SLID contact model instead of GLIS. The solution looks correct, see Figures 90 and 91, and is very close to those of cases SC3D15 (GLIS) and SC3D19 (GPIN).



(a) Z-displacements

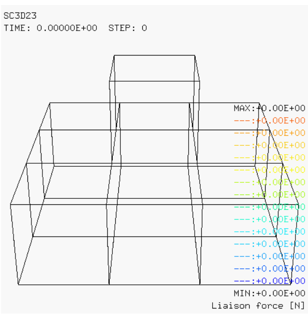


(b) X-displacements

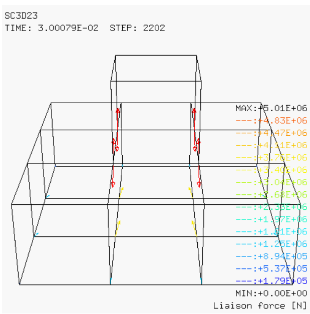


(c) SC3D23 vs. 15&19

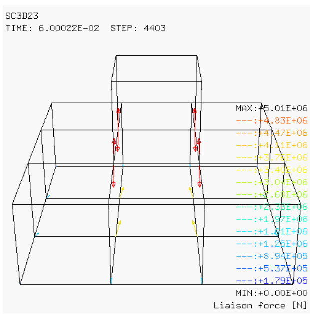
Figure 90: Some results of test SC3D23.



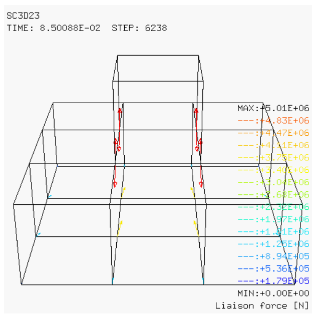
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

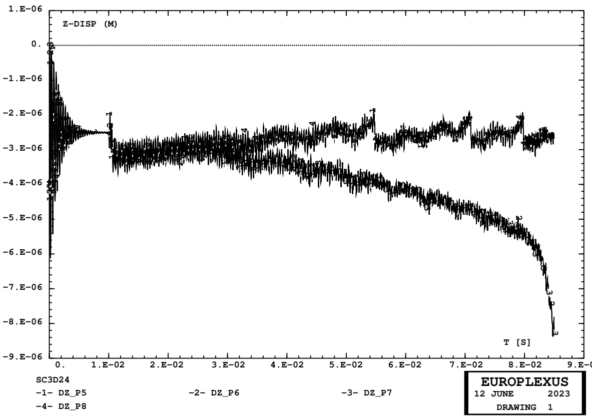


(d) $t = 85$ ms

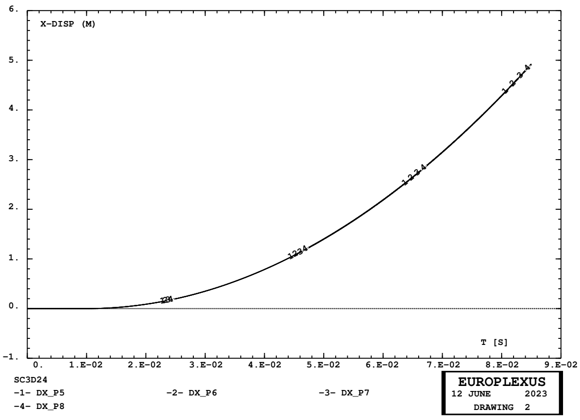
Figure 91: Deformed configuration and contact forces in test SC3D23.

7.3.23 Case SC3D24

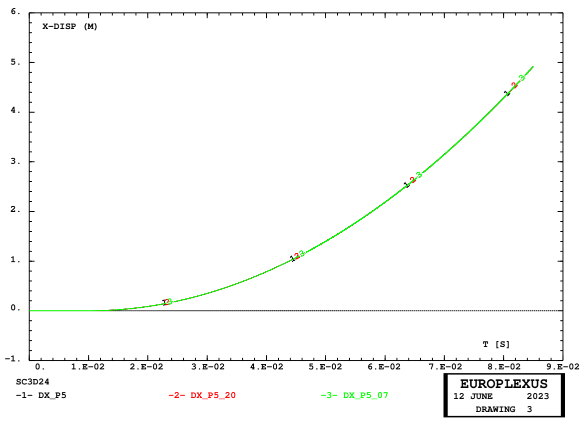
This test is similar to SC3D07 (without friction) but uses the SLID contact model instead of GLIS. The solution looks correct, see Figures 92 and 93, and is very close to those of cases SC3D07 (GLIS) and SC3D20 (GPIN).



(a) Z-displacements

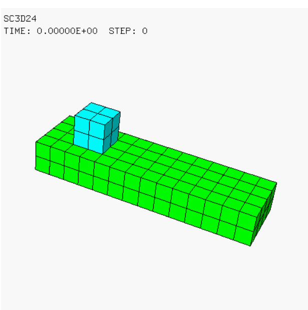


(b) X-displacements

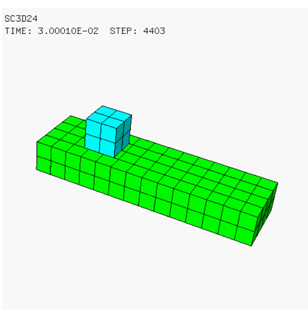


(c) SC3D22 vs. 07&20

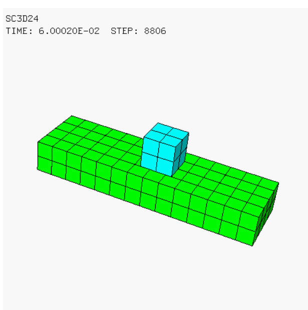
Figure 92: Some results of test SC3D24.



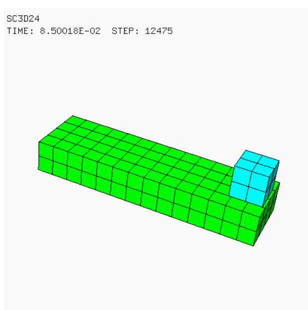
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms

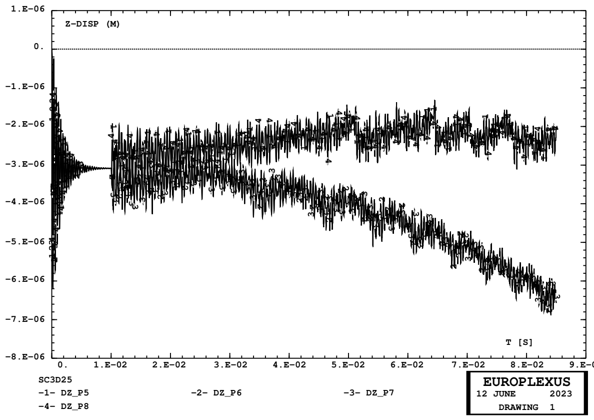


(d) $t = 85$ ms

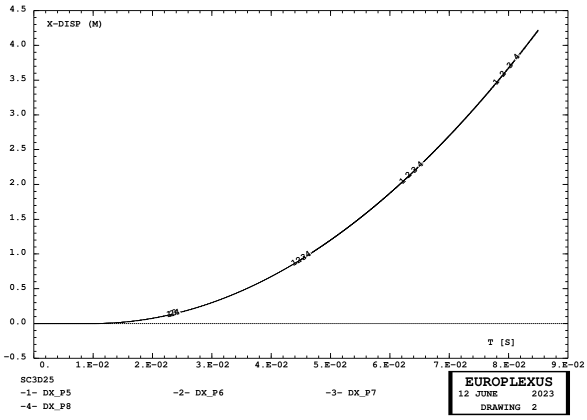
Figure 93: Deformed configuration and contact forces in test SC3D24.

7.3.24 Case SC3D25

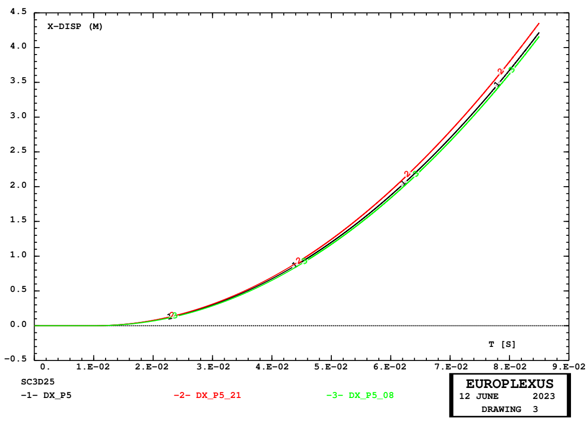
This test is similar to SC3D08 (with friction) but uses the SLID contact model instead of GLIS. The solution looks correct, see Figures 94 and 95, and is very close to those of cases SC3D08 (GLIS) and SC3D21 (GPIN).



(a) Z-displacements

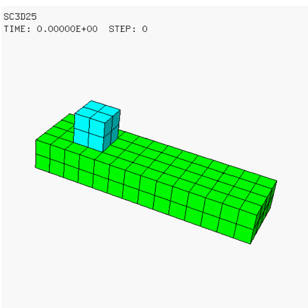


(b) X-displacements

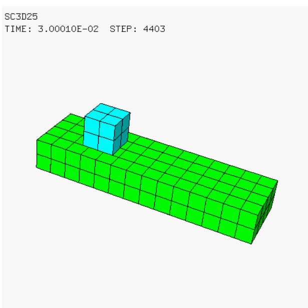


(c) SC3D25 vs. 07&20

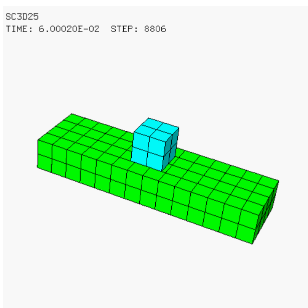
Figure 94: Some results of test SC3D25.



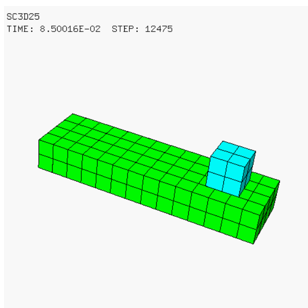
(a) $t = 0$



(b) $t = 30$ ms



(c) $t = 60$ ms



(d) $t = 85$ ms

Figure 95: Deformed configuration and contact forces in test SC3D25.

7.4 Inclined plane in 2D

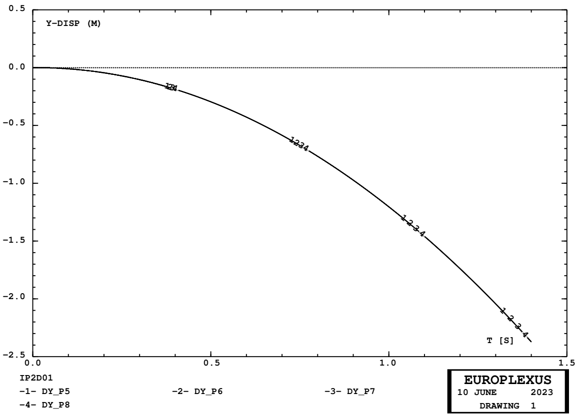
In this test, we consider a cube sliding along an inclined plane under the effect of gravity. The problem is first modeled in 2D. The plane inclination is taken to be $\alpha = 30°$. The simulations performed are summarized in Table 6.

Test	Mesh	Description	t_{fin} [s]	Steps	CPU [s]
IP2D01	32 Q42L 3 PMAT	No friction	1.4	92 820	14.1
IP2D02	32 Q42L 3 PMAT	Idem 01 but with friction	1.4	92 820	15.8
IP2D03	32 Q42L	Idem 01 (no friction) but GPIN	1.4	92 820	14.6
IP2D04	32 Q42L	Idem 02 (with friction) but GPIN	1.4	92 820	14.2
IP2D05	32 Q42L	Idem 01 (no friction) but SLID	1.4	92 820	13.0
IP2D06	32 Q42L	Idem 02 (with friction) but SLID	1.4	92 820	13.8
IP2D07	32 Q42L 3 PMAT	Idem 02 but $\mu = 0.6$	1.4	92 820	19.2
IP2D08	32 Q42L	Idem 04 but $\mu = 0.6$	1.4	92 820	14.7
IP2D09	32 Q42L	Idem 06 but $\mu = 0.6$	1.4	92 820	12.9

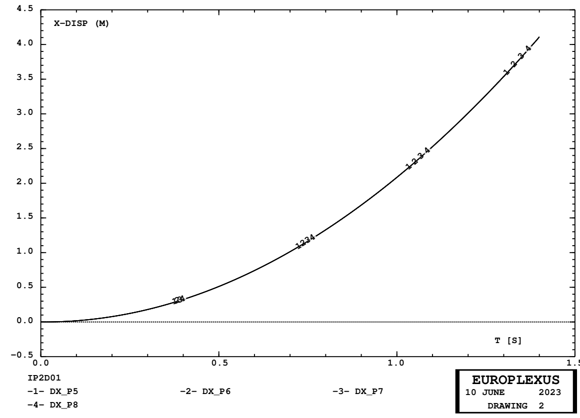
Table 6: Inclined plane tests in 2D.

7.4.1 Case IP2D01

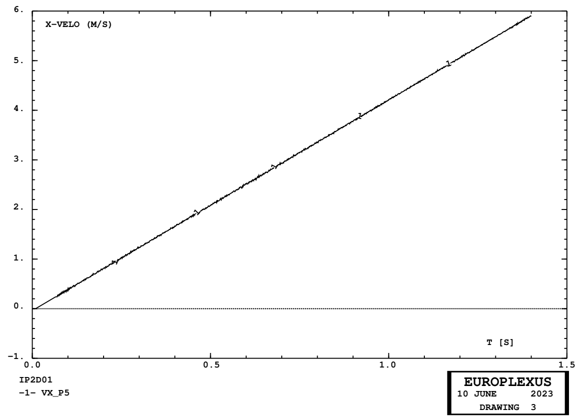
This solution is obtained without friction. The solution looks correct, see Figures 96 and 97. Only the acceleration looks a bit noisy.



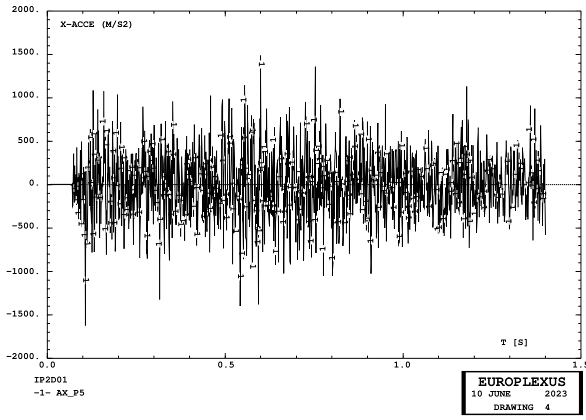
(a) Y-displacements



(b) X-displacements

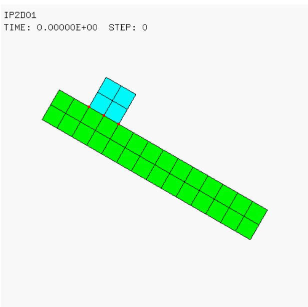


(c) X-velocity

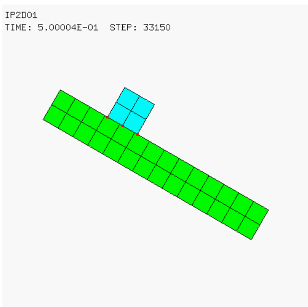


(d) X-acceleration

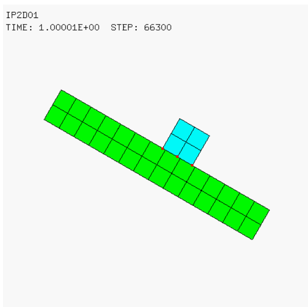
Figure 96: Some results of test IP2D01.



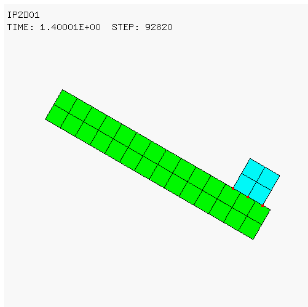
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

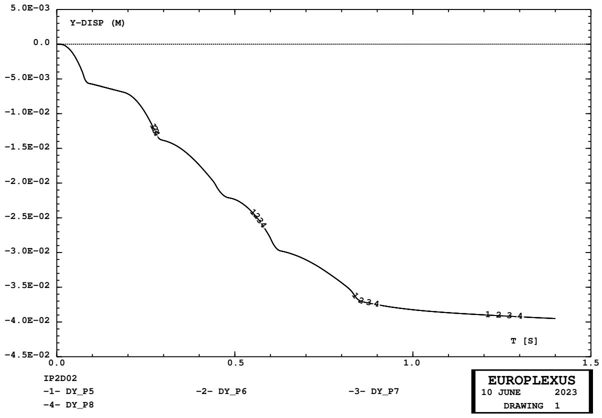


(d) $t = 1.4$ s

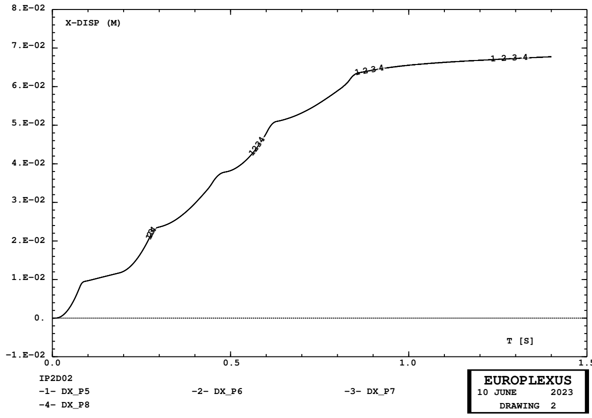
Figure 97: Deformed configuration in test IP2D01.

7.4.2 Case IP2D02

This test is identical to IP2D01 but we add friction. The solution looks weird, see Figures 98 and 99. Initially the cube moves a bit upwards with respect to the plane. Then, it seems to get stuck and slides very little along the plane.

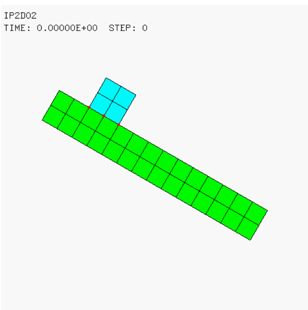


(a) Y-displacements

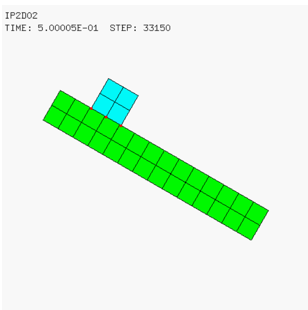


(b) X-displacements

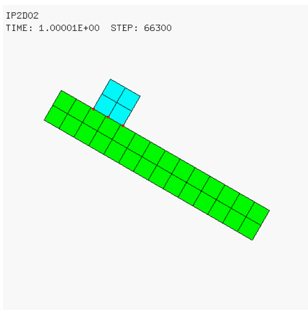
Figure 98: Some results of test IP2D02.



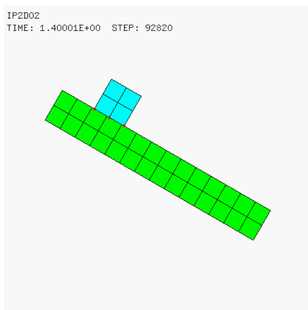
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

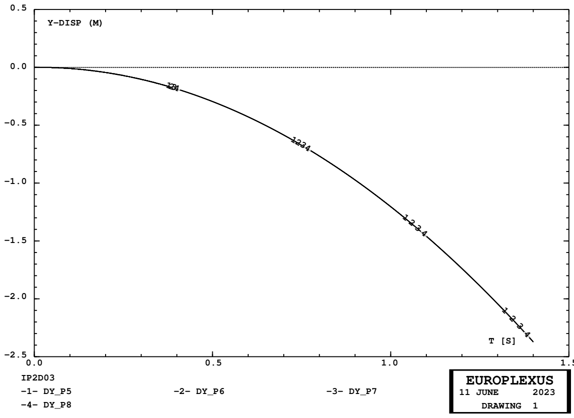


(d) $t = 1.4$ s

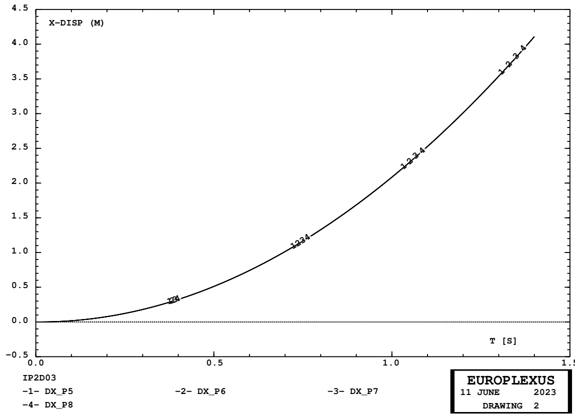
Figure 99: Deformed configuration in test IP2D02.

7.4.3 Case IP2D03

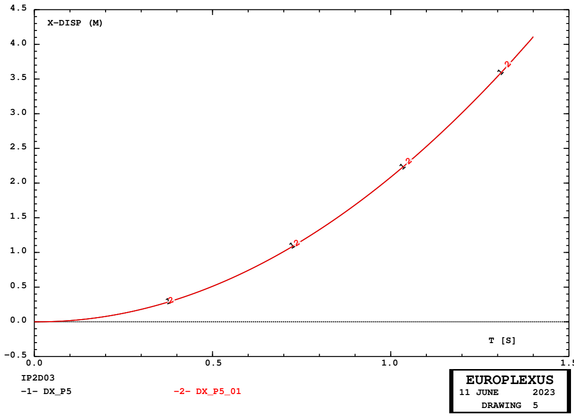
This test is similar to IP2D01 (no friction) but uses the GPIN contact model. The solution is smooth and visually identical (time curves) to that of case IP2D01, see Figures 100 and 101.



(a) Y-displacements

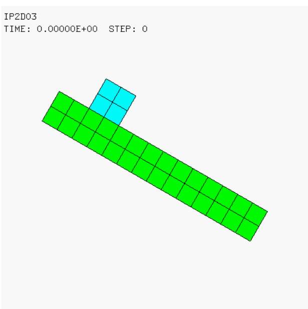


(b) X-displacements

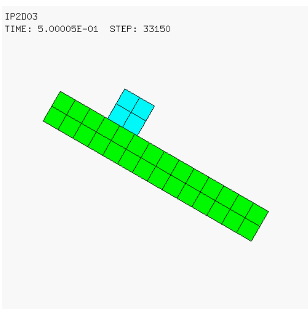


(c) IP2D03 vs. IP2D01

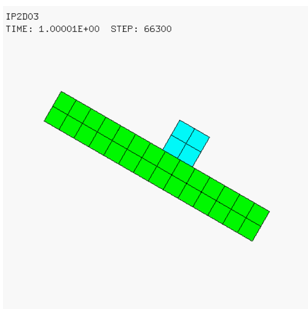
Figure 100: Some results of test IP2D03.



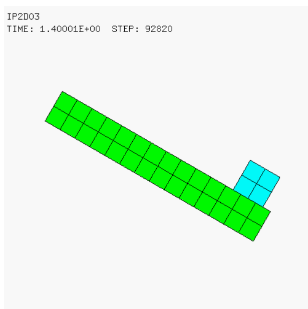
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

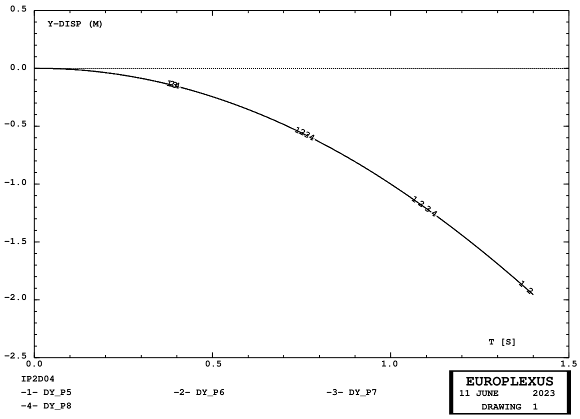


(d) $t = 1.4$ s

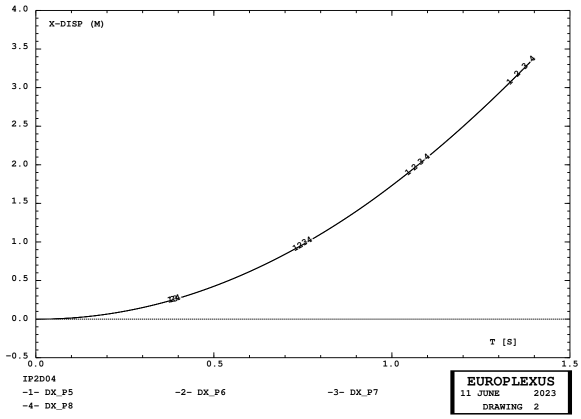
Figure 101: Deformed configuration in test IP2D03.

7.4.4 Case IP2D04

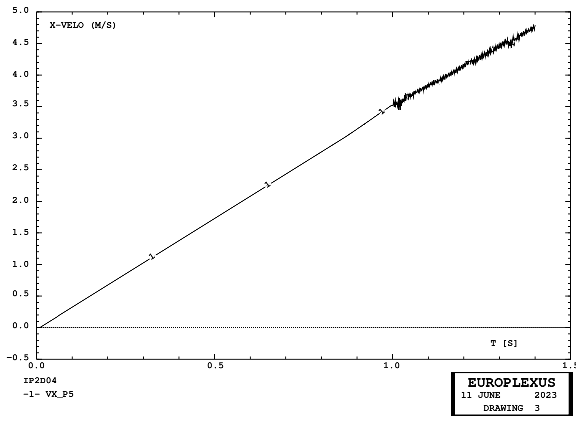
This test is similar to IP2D02 (withj friction) but uses the GPIN contact model. The solution is smooth and visually correct, see Figures 102 and 103. The displacement is lower than that of case IP2D03, as expected. There are small-amplitude high-frequency oscillations in the velocity near the end of the solution.



(a) Y-displacements

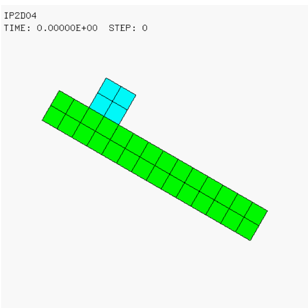


(b) X-displacements

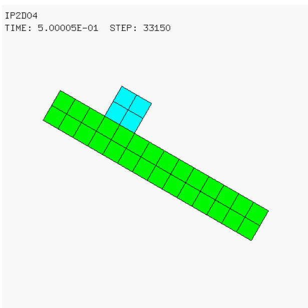


(c) X-velocity

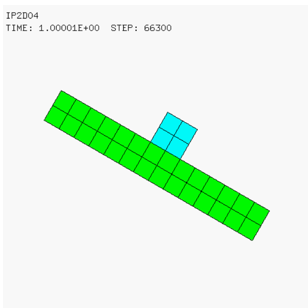
Figure 102: Some results of test IP2D04.



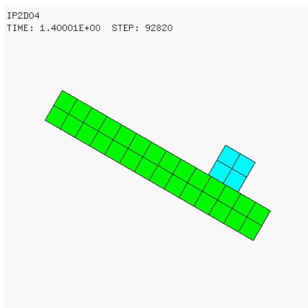
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

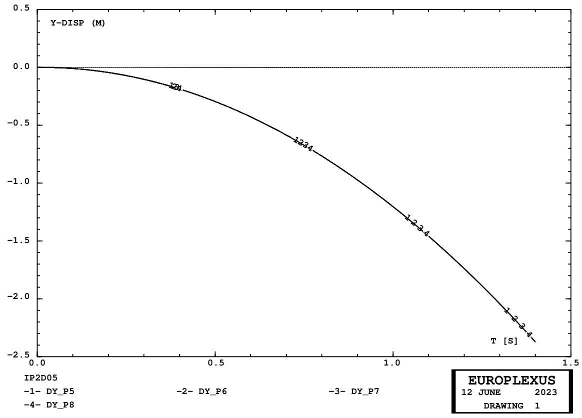


(d) $t = 1.4$ s

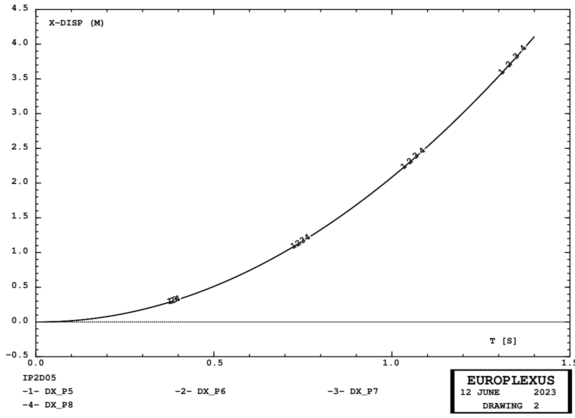
Figure 103: Deformed configuration in test IP2D04.

7.4.5 Case IP2D05

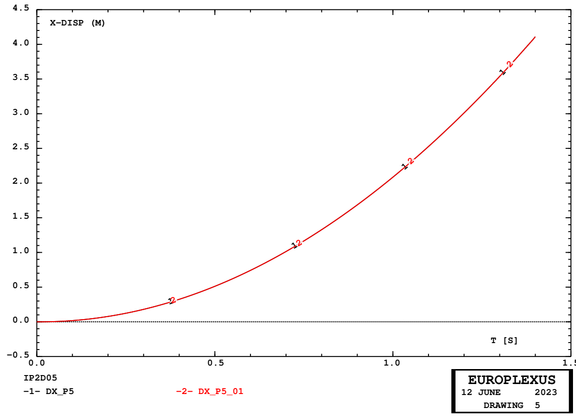
This test is similar to IP2D01 (no friction) but uses the SLID contact model. The solution is smooth and visually identical (time curves) to that of case IP2D01, see Figures 104 and 105.



(a) Y-displacements

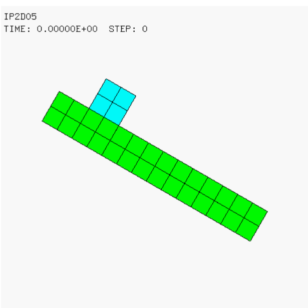


(b) X-displacements

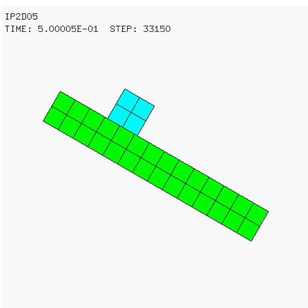


(c) IP2D05 vs. IP2D01

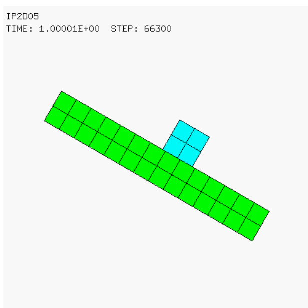
Figure 104: Some results of test IP2D05.



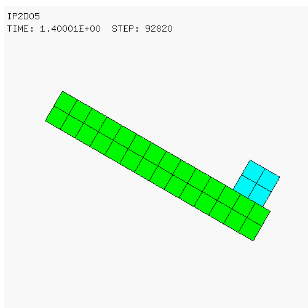
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

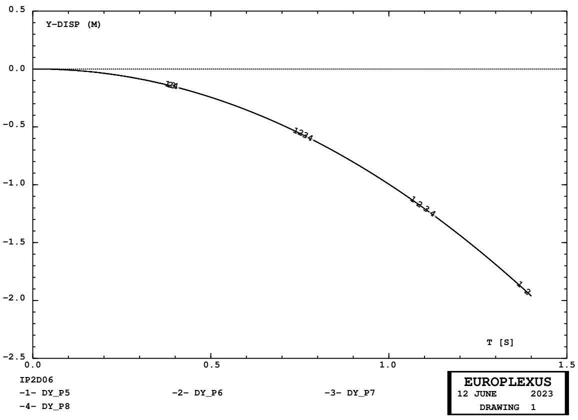


(d) $t = 1.4$ s

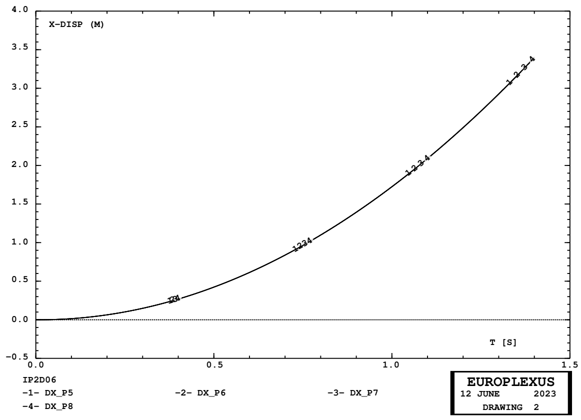
Figure 105: Deformed configuration in test IP2D05.

7.4.6 Case IP2D06

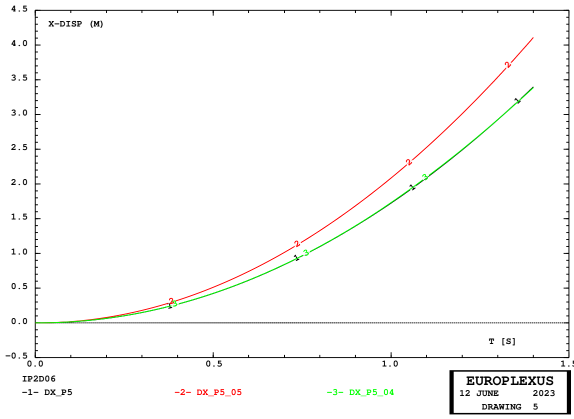
This test is similar to IP2D02 (withj friction) but uses the SLID contact model. The solution is smooth and visually correct, see Figures 106 and 107. The displacement is lower than that of case IP2D05, as expected, and very similar to that of case IP2D04 (GPIN with friction). There are no oscillations in the velocity near the end of the solution.



(a) Y-displacements

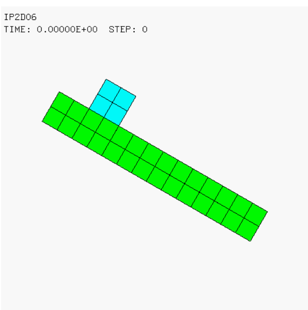


(b) X-displacements

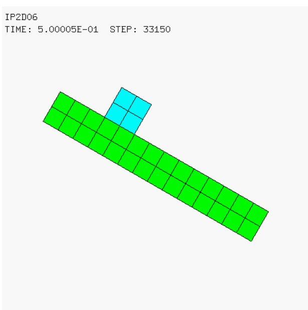


(c) IP2D05 vs. 04&05

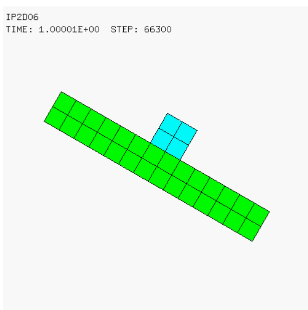
Figure 106: Some results of test IP2D06.



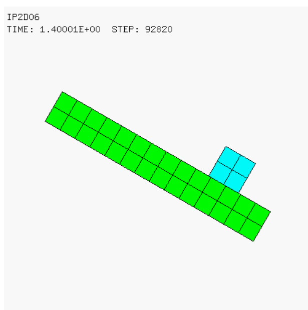
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s



(d) $t = 1.4$ s

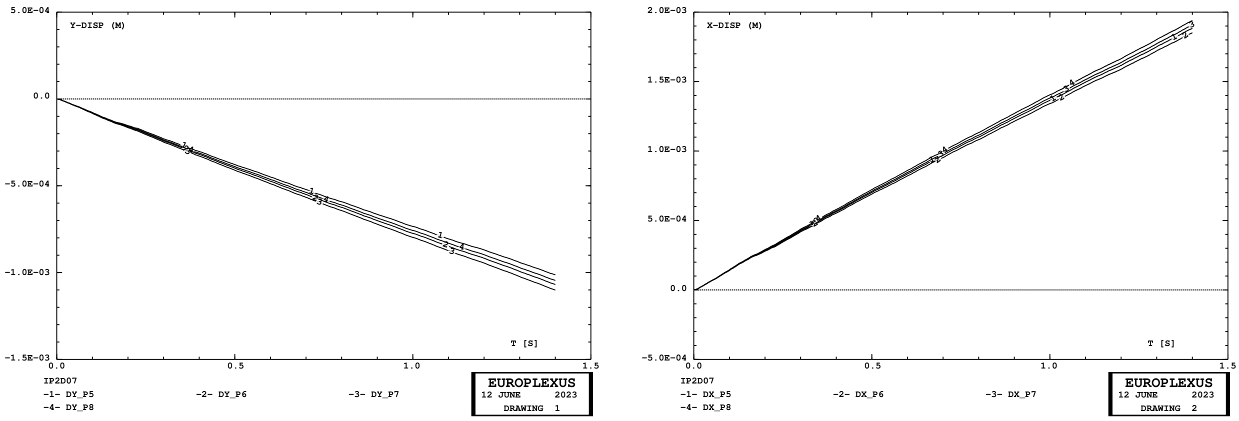
Figure 107: Deformed configuration in test IP2D06.

7.4.7 Case IP2D07

This test is similar to IP2D02 (PINB with friction) but uses $\mu = 0.6$. The scope is to check the performance of the friction algorithm for large values of the friction coefficient.

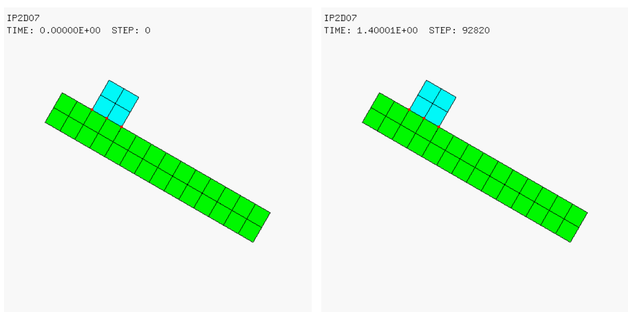
In this example, the angle of inclination of the plane is $\alpha = 30°$. Therefore, one would expect that, by taking μ equal to the critical value $\mu_{\text{crit}} = \tan\alpha \approx 0.577$ or larger, the cube should not slide along the plane and should remain in its initial position.

The solution is not correct, since the cube moves (albeit very slowly), see Figures 108 and 109. The cube displacement is perhaps too small to be appreciated in Figure 108(b), but not negligible.



(a) Y-displacements (b) X-displacements

Figure 108: Some results of test IP2D07.



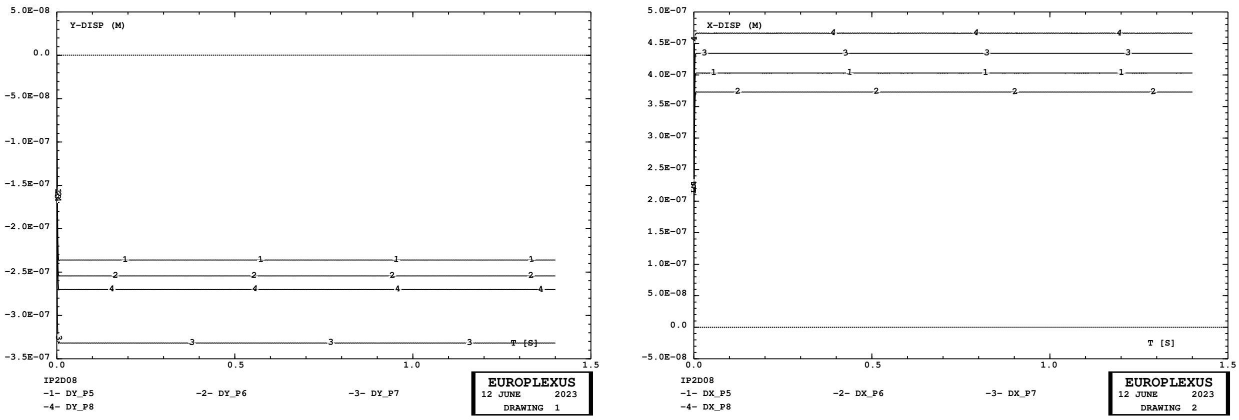
(a) $t = 0$ (b) $t = 1.4$ s

Figure 109: Deformed configuration in test IP2D07.

7.4.8 Case IP2D08

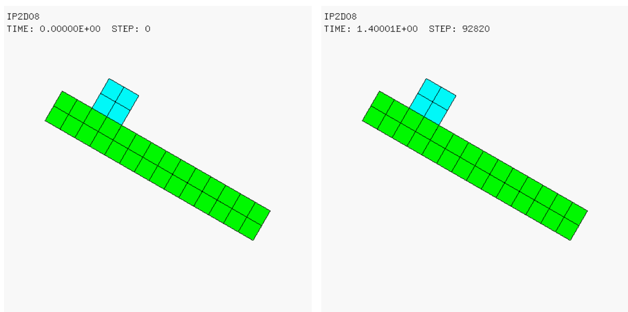
This test is similar to IP2D04 (GPIN with friction) but uses $\mu = 0.6$, so that the cube should not slide along the plane.

The solution is correct, see Figures 110 and 111.



(a) Y-displacements (b) X-displacements

Figure 110: Some results of test IP2D08.



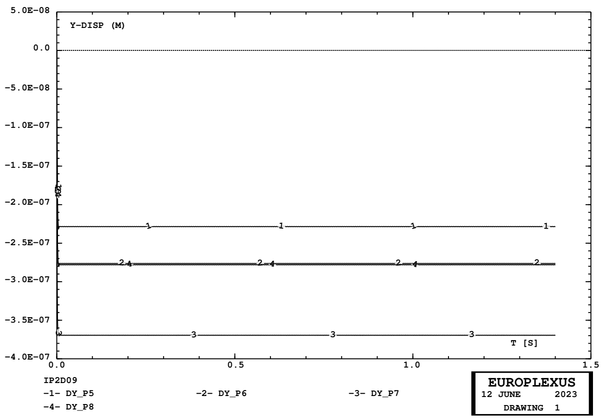
(a) $t = 0$ (b) $t = 1.4$ s

Figure 111: Deformed configuration in test IP2D08.

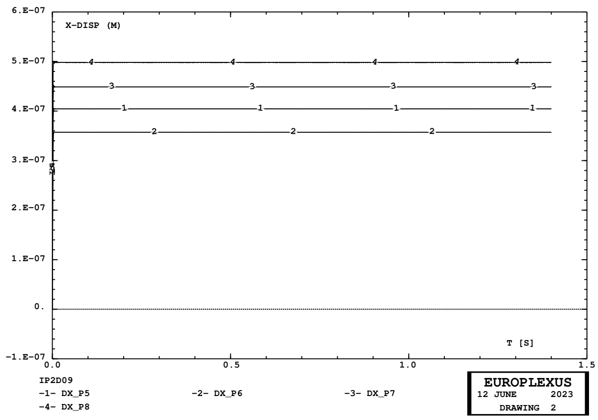
7.4.9 Case IP2D09

This test is similar to IP2D06 (SLID with friction) but uses $\mu = 0.6$, so that the cube should not slide along the plane.

The solution is correct, see Figures 112 and 113.

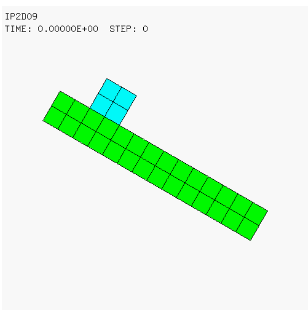


(a) Y-displacements

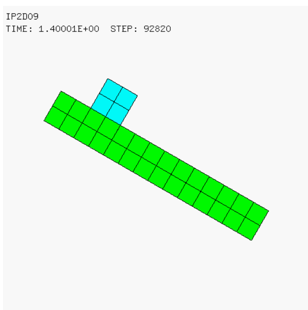


(b) X-displacements

Figure 112: Some results of test IP2D09.



(a) $t = 0$



(b) $t = 1.4$ s

Figure 113: Deformed configuration in test IP2D09.

7.5 Inclined plane in 3D

This is the 3D version of the cube sliding along an inclined plane presented in the previous Section. The inclination angle is $\alpha = 30°$, the same as in the 2D model.

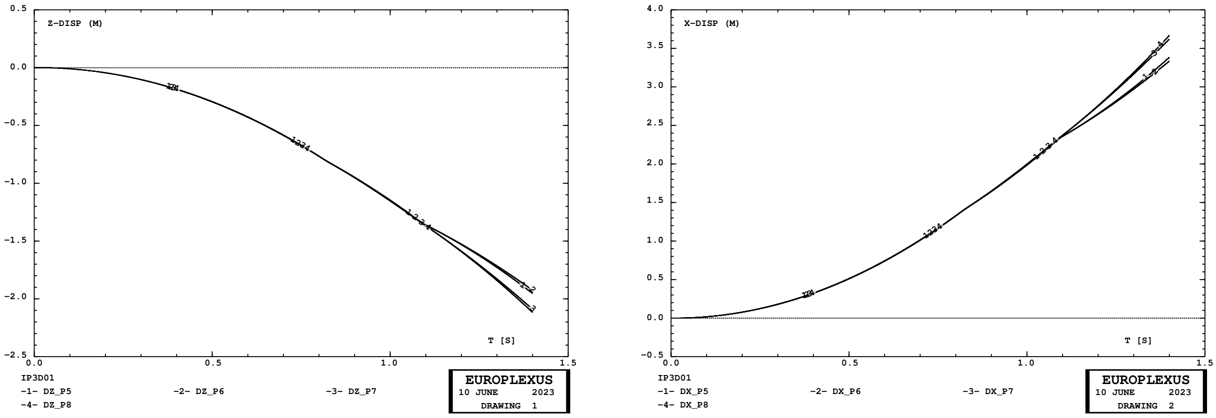
The simulations performed are summarized in Table 7.

Test	Mesh	Description	t_{fin} [s]	Steps	CPU [s]
IP3D01	120 C82L 9 PMAT	No friction	1.4	205 471	210.6
IP3D02	120 C82L 9 PMAT	Idem 01 but with friction	1.4	207 330	138.1
IP3D03	120 C82L 9 PMAT	Idem 01 but GLIS	1.4	205 464	153.9
IP3D04	120 C82L 9 PMAT	Idem 02 but GLIS	1.4	205 464	166.9
IP3D05	120 C82L	Idem 03 (no friction) but GPIN	1.4	205 464	190.9
IP3D06	120 C82L	Idem 04 (with friction) but GPIN	1.4	205 466	195.4
IP3D07	120 C82L	Idem 03 (no friction) but SLID	1.4	205 464	153.1
IP3D08	120 C82L	Idem 04 (with friction) but SLID	1.4	205 464	164.9
IP3D09	120 C82L 9 PMAT	Idem 02 but $\mu = 0.6$	1.4	216 980	161.5
IP3D10	120 C82L 9 PMAT	Idem 04 but $\mu = 0.6$	1.4	205 465	152.0
IP3D11	120 C82L	Idem 06 but $\mu = 0.6$	1.4	205 464	188.8
IP3D12	120 C82L	Idem 08 but $\mu = 0.6$	1.4	205 464	175.0
IP3D13	120 C82L 9 PMAT	Idem 10 but $\mu = 0.7$	1.4	205 464	163.9
IP3D14	120 C82L 9 PMAT	Idem 10 but damped up to 100 ms	1.4	205 465	154.8
IP3D15	120 C82L 9 PMAT	Idem 10 but damped up to 500 ms	1.4	205 464	162.0
IP3D16	120 C82L 9 PMAT	Idem 09 but $\mu = 0.7$	1.4	897 523	523.9
IP3D17	120 C82L 9 PMAT	Idem 09 but PINB PENA	1.4	206 216	1 732.1

Table 7: Inclined plane tests in 3D.

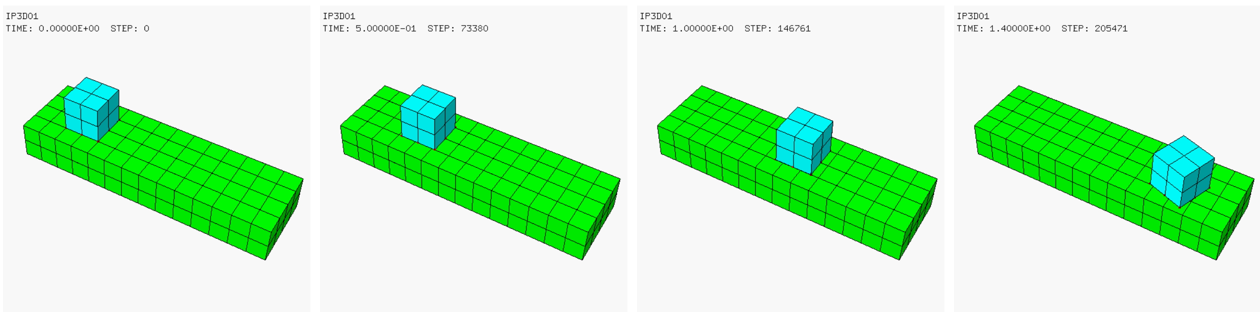
7.5.1 Case IP3D01

This case uses no friction. The cube motion is relatively correct initially, but the trajectory bends towards the end, see Figures 114 and 115. Therefore, the solution cannot be considered satisfactory.



(a) Z-displacements (b) X-displacements

Figure 114: Some results of test IP3D01.

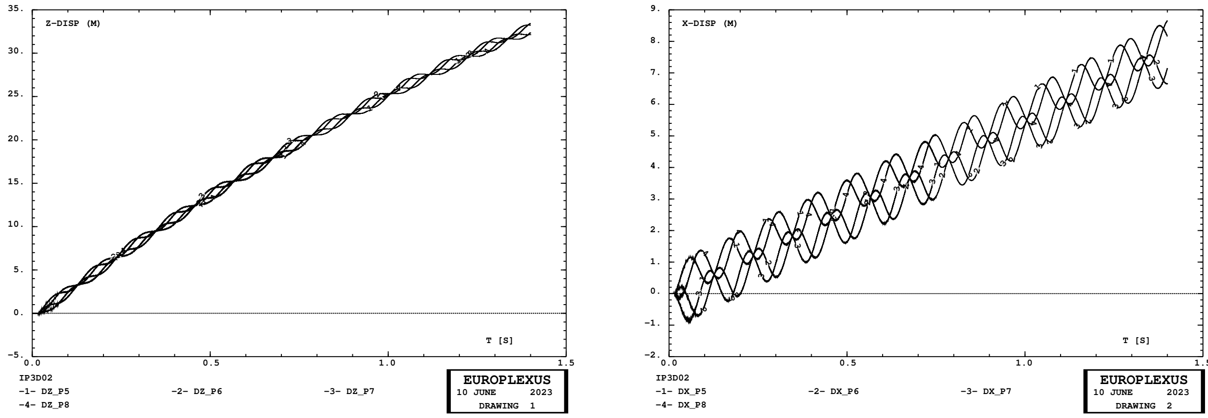


(a) $t = 0$ (b) $t = 0.5$ s (c) $t = 1.0$ s (d) $t = 1.4$ s

Figure 115: Deformed configuration in test IP3D01.

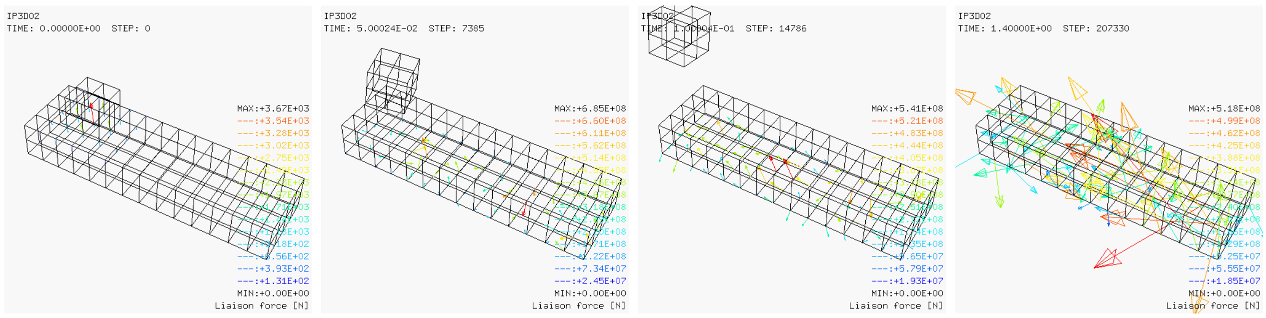
7.5.2 Case IP3D02

This test is identical to IP3D01 but we add friction. The solution is completely weird. After a few steps the cube starts flying away, see Figures 116 and 117.



(a) Z-displacements (b) X-displacements

Figure 116: Some results of test IP3D02.

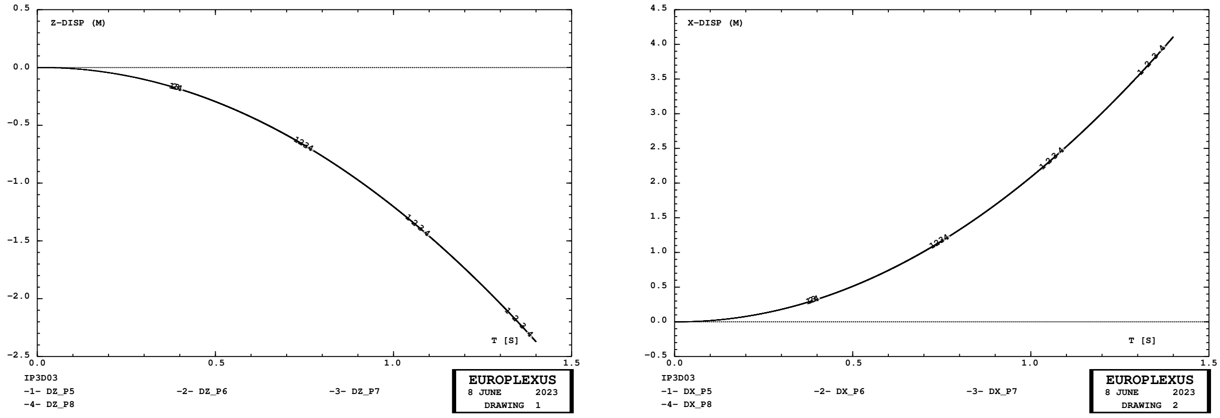


(a) $t = 0$ (b) $t = 0.05$ s (c) $t = 0.01$ s (d) $t = 1.4$ s

Figure 117: Deformed configuration and reactions in test IP3D02.

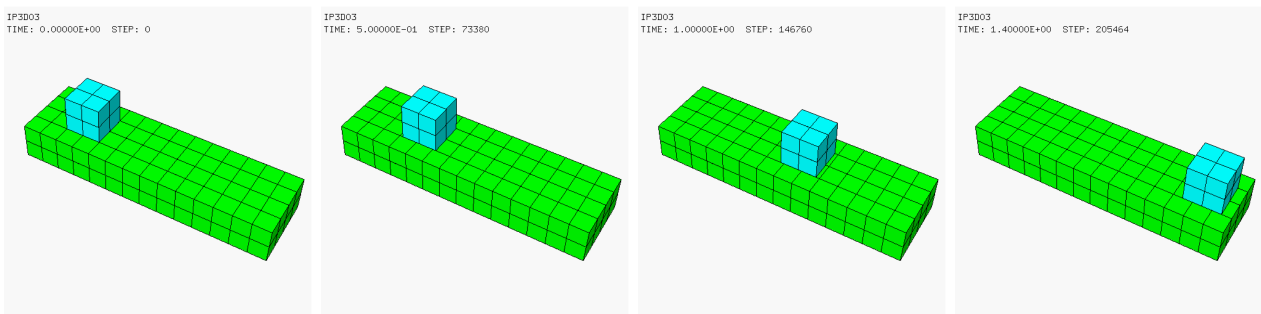
7.5.3 Case IP3D03

This test is identical to IP3D01 (no friction) but we use GLIS. The solution is correct, see Figures 118 and 119.



(a) Z-displacements (b) X-displacements

Figure 118: Some results of test IP3D03.

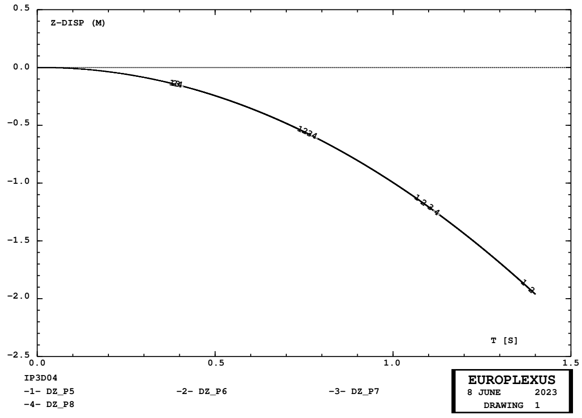


(a) $t = 0$ (b) $t = 0.5$ s (c) $t = 1.0$ s (d) $t = 1.4$ s

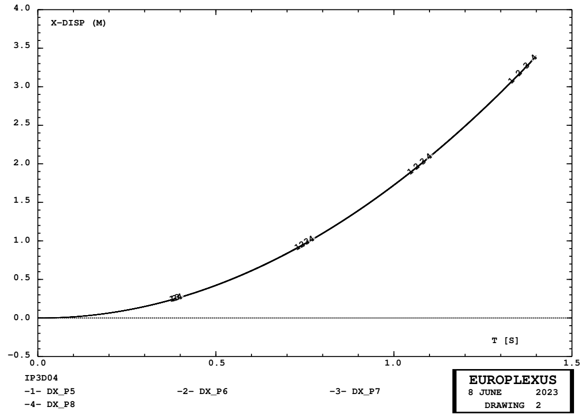
Figure 119: Deformed configuration and reactions in test IP3D03.

7.5.4 Case IP3D04

This test is identical to IP3D02 (with friction) but we use GLIS. The solution is correct, see Figures 120 and 121.

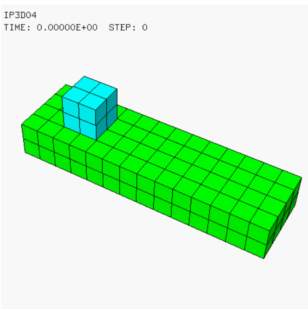


(a) Z-displacements

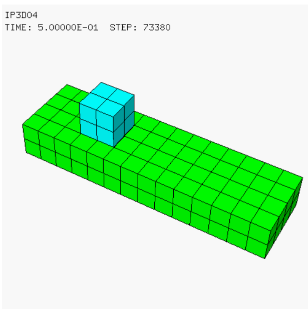


(b) X-displacements

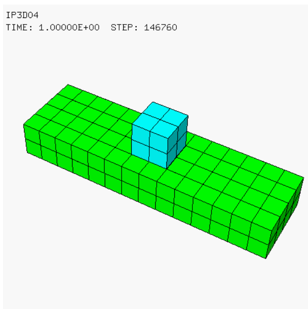
Figure 120: Some results of test IP3D04.



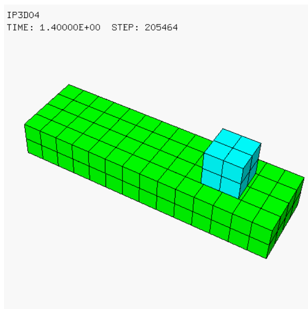
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

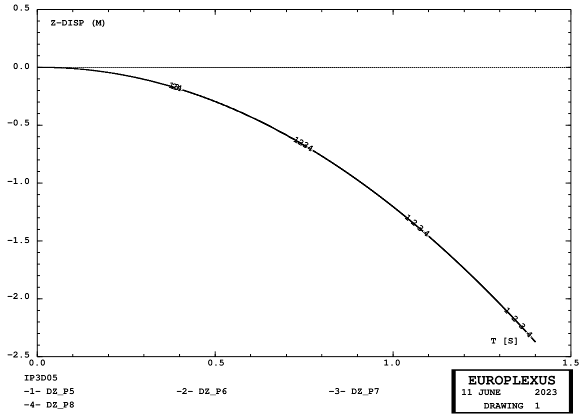


(d) $t = 1.4$ s

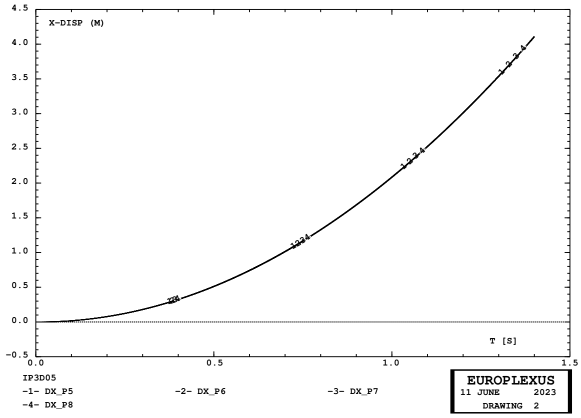
Figure 121: Deformed configuration and reactions in test IP3D04.

7.5.5 Case IP3D05

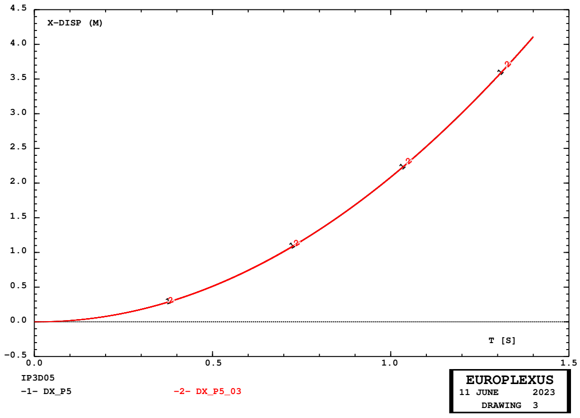
This test is identical to IP3D03 (no friction, GLIS) but we use the GPIN contact model. The solution is correct, and visually identical (time curves) to that of case IP3D03, see Figures 122 and 123.



(a) Z-displacements

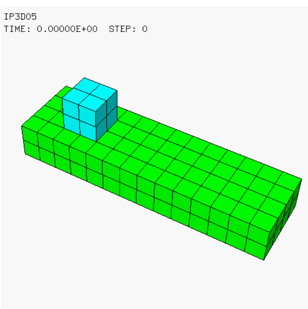


(b) X-displacements

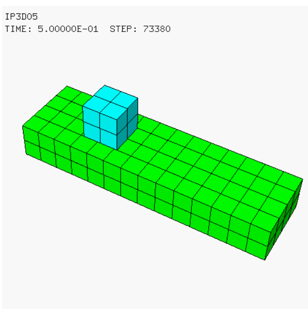


(c) IP3D05 vs. IP3D03

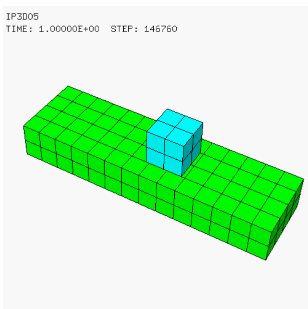
Figure 122: Some results of test IP3D05.



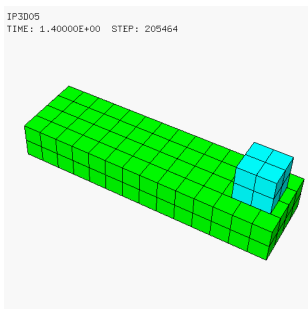
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

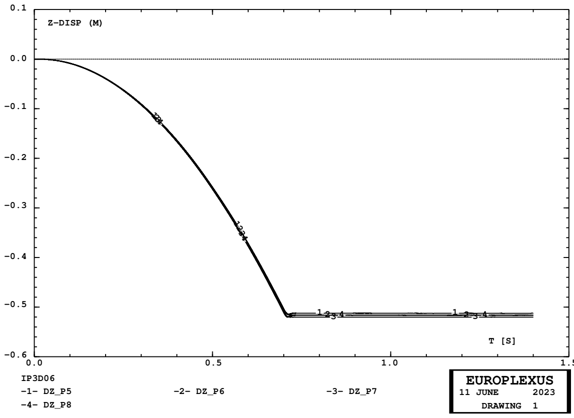


(d) $t = 1.4$ s

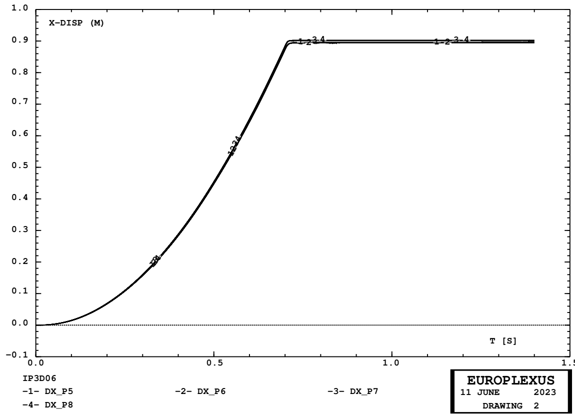
Figure 123: Deformed configuration and reactions in test IP3D05.

7.5.6 Case IP3D06

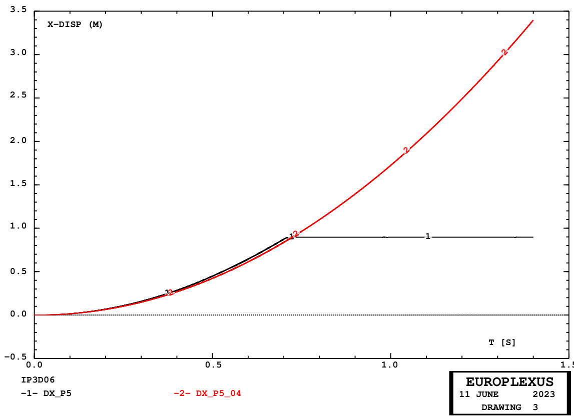
This test is identical to IP3D04 (with friction, GLIS) but we use the GPIN contact model. The solution is fairly correct (compared against IP3D04 using GLIS) until about 0.7 s, then the cube gets stuck, see Figures 124 and 125.



(a) Z-displacements

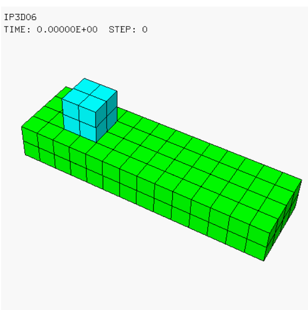


(b) X-displacements

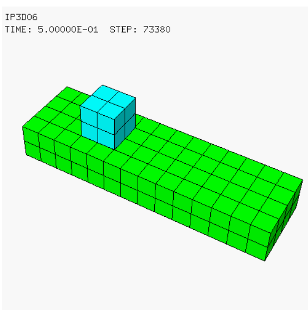


(c) IP3D06 vs. IP3D04

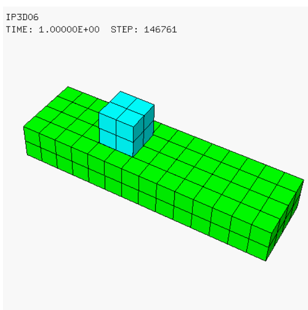
Figure 124: Some results of test IP3D06.



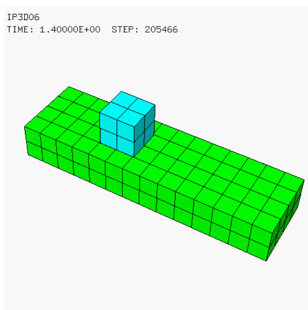
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s

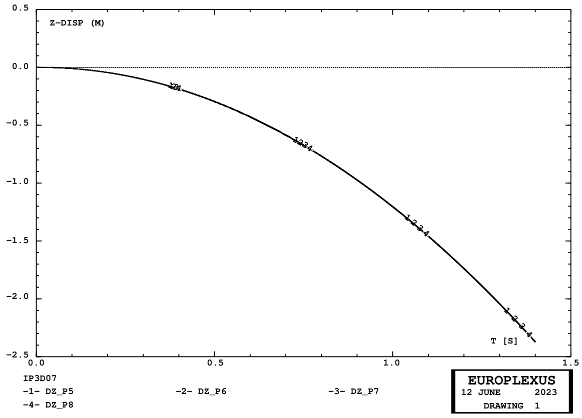


(d) $t = 1.4$ s

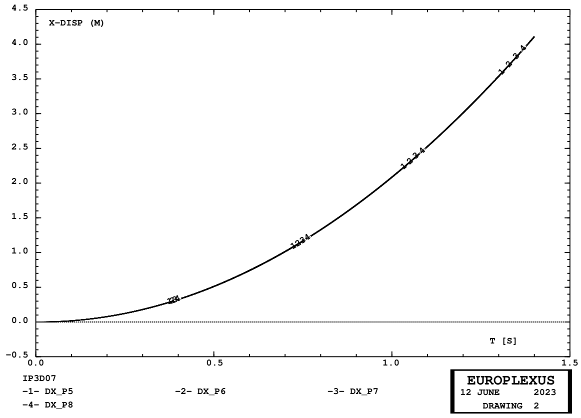
Figure 125: Deformed configuration and reactions in test IP3D06.

7.5.7 Case IP3D07

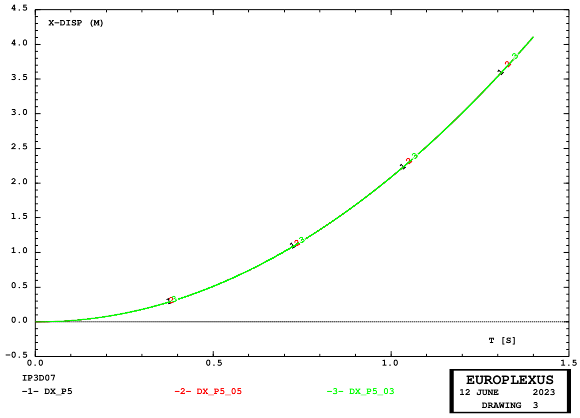
This test is identical to IP3D03 (no friction, GLIS) but we use the SLID contact model. The solution is correct, and visually identical (time curves) to that of cases IP3D03 (GLIS) and IP3D05 (GPIN), see Figures 126 and 127.



(a) Z-displacements

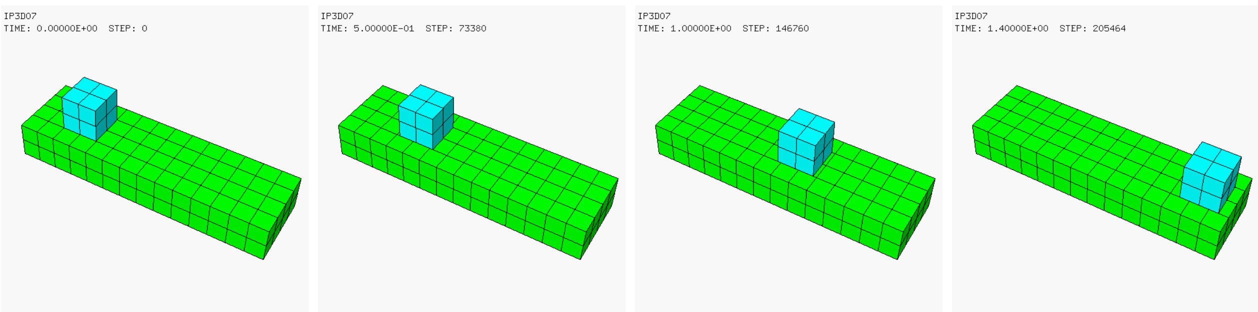


(b) X-displacements



(c) IP3D07 vs. 03&05

Figure 126: Some results of test IP3D07.

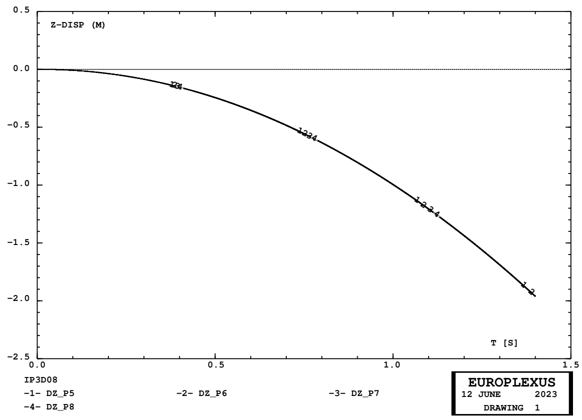


(a) $t = 0$ (b) $t = 0.5$ s (c) $t = 1.0$ s (d) $t = 1.4$ s

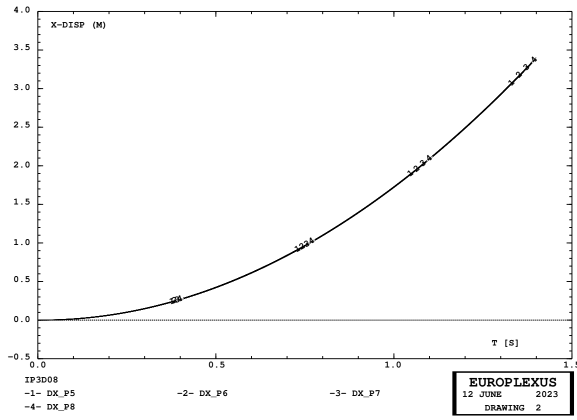
Figure 127: Deformed configuration and reactions in test IP3D07.

7.5.8 Case IP3D08

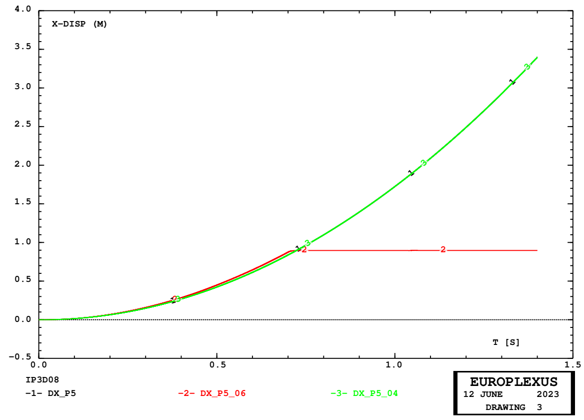
This test is identical to IP3D04 (with friction, GLIS) but we use the SLID contact model. The solution is correct and almost identical to that of case IP3D04 (GLIS) see Figures 128 and 129.



(a) Z-displacements

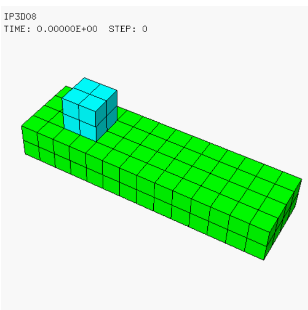


(b) X-displacements

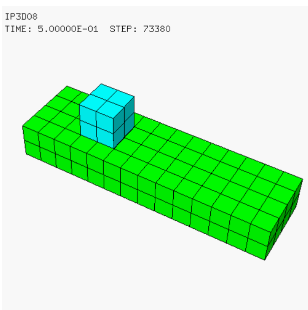


(c) IP3D08 vs. 04&06

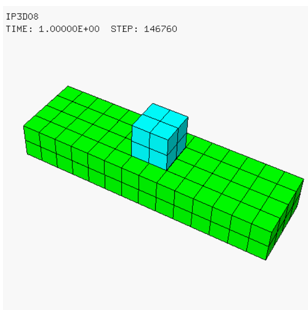
Figure 128: Some results of test IP3D08.



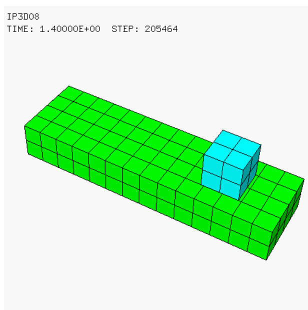
(a) $t = 0$



(b) $t = 0.5$ s



(c) $t = 1.0$ s



(d) $t = 1.4$ s

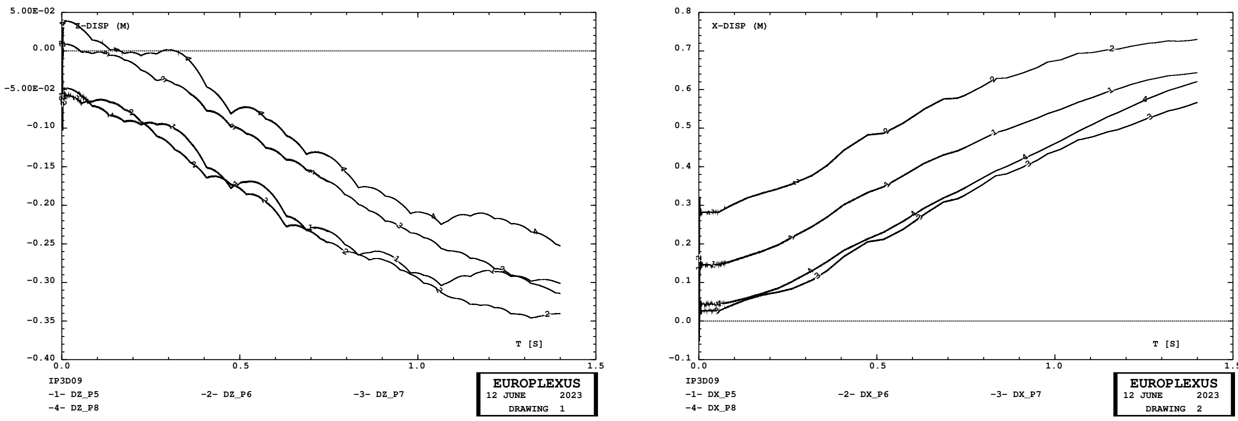
Figure 129: Deformed configuration and reactions in test IP3D08.

7.5.9 Case IP3D09

This test is similar to IP3D02 (PINB with friction) but uses $\mu = 0.6$. The scope is to check the performance of the friction algorithm for large values of the friction coefficient.

In this example, the angle of inclination of the plane is $\alpha = 30°$. Therefore, one would expect that, by taking μ equal to the critical value $\mu_{\text{crit}} = \tan\alpha \approx 0.577$ or larger, the cube should not slide along the plane and should remain in its initial position.

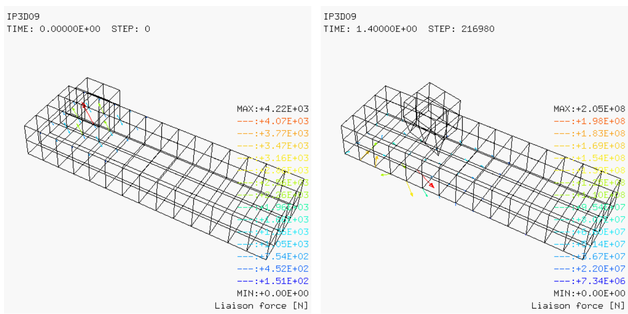
The solution is weird, since the cube visibly moves and deforms, see Figures 130 and 131.



(a) Z-displacements

(b) X-displacements

Figure 130: Some results of test IP3D09.



(a) $t = 0$

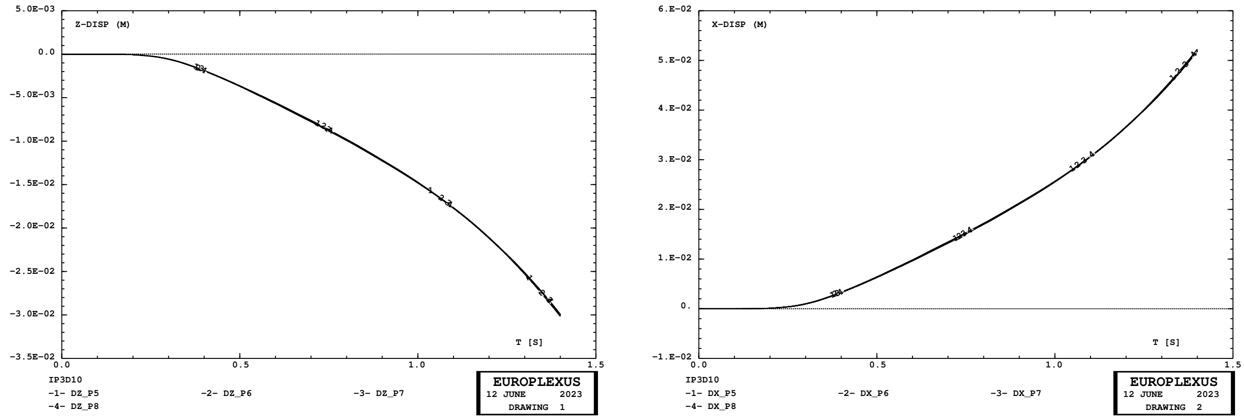
(b) $t = 1.4$ s

Figure 131: Deformed configuration in test IP3D09.

7.5.10 Case IP3D10

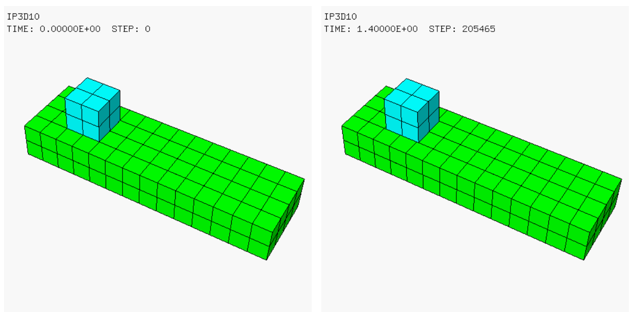
This test is similar to IP3D04 (GLIS with friction) but uses $\mu = 0.6$. The cube should not slide along the plane and should remain in its initial position.

The solution is not correct, since the cube moves (albeit very slowly), see Figures 132 and 133. The cube displacement is perhaps too small to be appreciated in Figure 132(b), but not negligible.



(a) Z-displacements (b) X-displacements

Figure 132: Some results of test IP3D10.



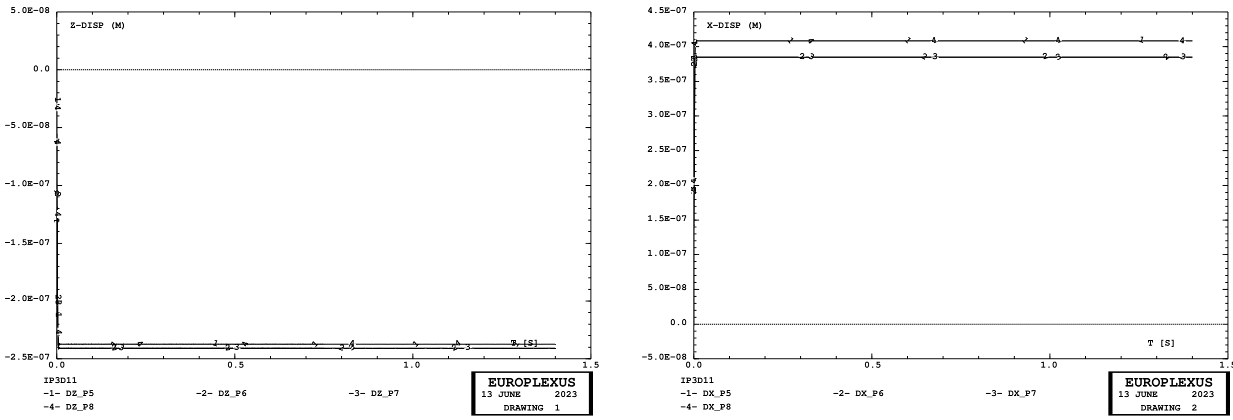
(a) $t = 0$ (b) $t = 1.4$ s

Figure 133: Deformed configuration in test IP3D10.

7.5.11 Case IP3D11

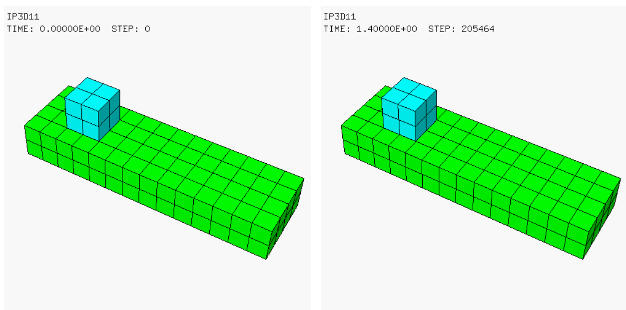
This test is similar to IP3D06 (GPIN with friction) but uses $\mu = 0.6$. The cube should not slide along the plane and should remain in its initial position.

The solution is correct, since the cube remains at its initial position' see Figures 134 and 135.



(a) Z-displacements (b) X-displacements

Figure 134: Some results of test IP3D11.



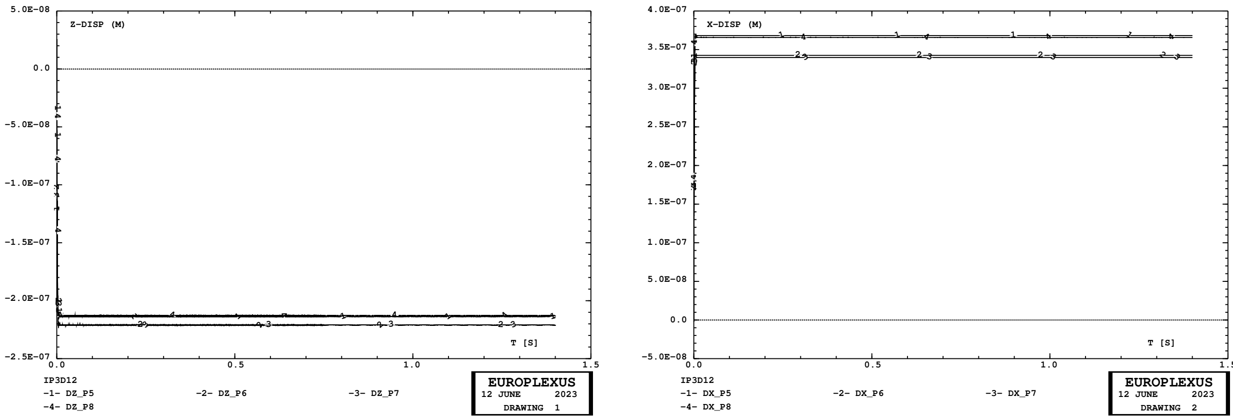
(a) $t = 0$ (b) $t = 1.4$ s

Figure 135: Deformed configuration in test IP3D11.

7.5.12 Case IP3D12

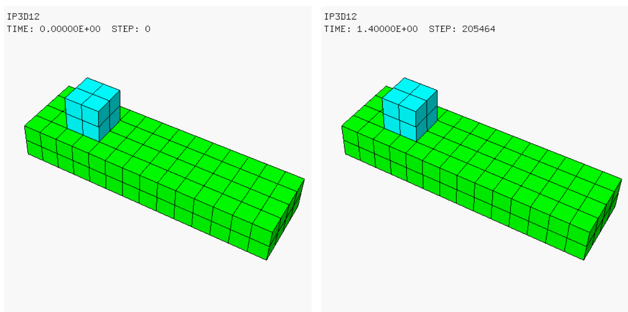
This test is similar to IP3D08 (SLID with friction) but uses $\mu = 0.6$. The cube should not slide along the plane and should remain in its initial position.

The solution is correct, since the cube remains at its initial position' see Figures 136 and 137.



(a) Z-displacements (b) X-displacements

Figure 136: Some results of test IP3D12.



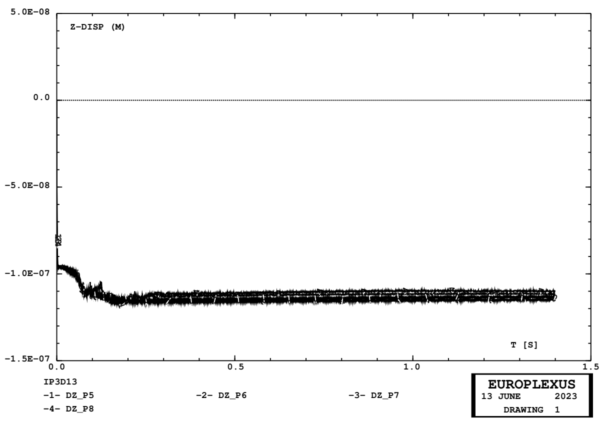
(a) $t = 0$ (b) $t = 1.4$ s

Figure 137: Deformed configuration in test IP3D12.

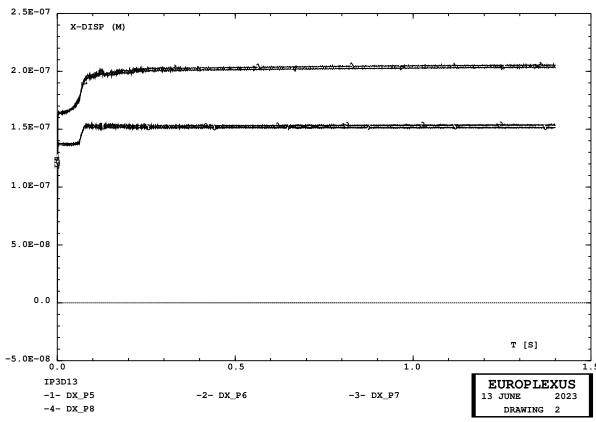
7.5.13 Case IP3D13

This test is similar to IP3D10 (GLIS with friction) but we increase the friction coefficient to $\mu = 0.7$, to see whether this is sufficient to block the cube.

Indeed, the solution looks correct, since the cube does not move appreciably, see Figures 138 and 139.

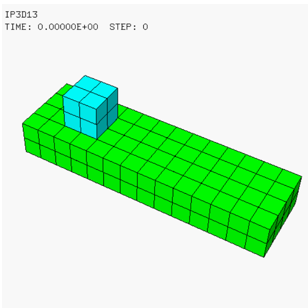


(a) Z-displacements

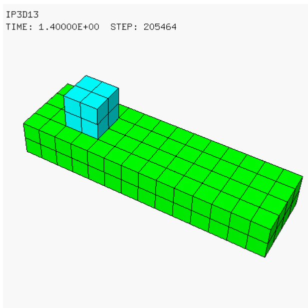


(b) X-displacements

Figure 138: Some results of test IP3D13.



(a) $t = 0$



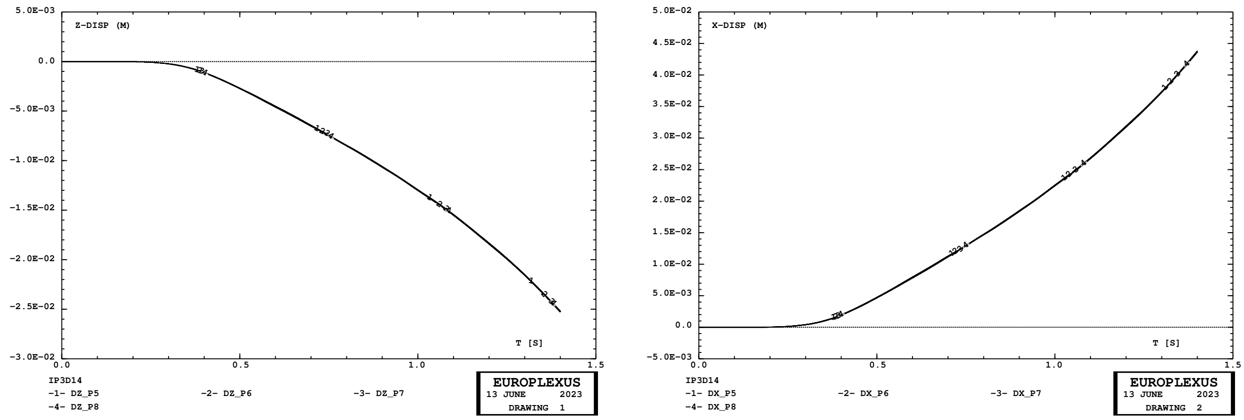
(b) $t = 1.4$ s

Figure 139: Deformed configuration in test IP3D13.

7.5.14 Case IP3D14

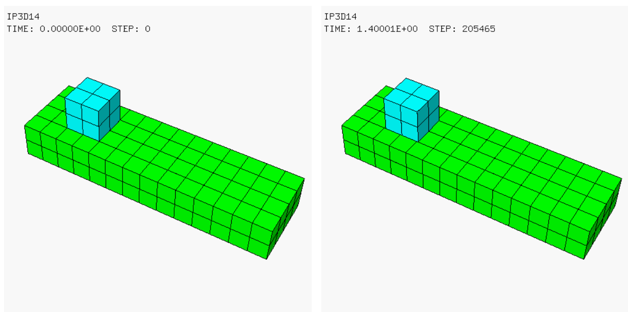
This test is similar to IP3D10 (GLIS with friction) but we apply the quasi-static damping until 100 ms instead of 10 ms, to see whether this is sufficient to block the cube.

The solution is incorrect, since the cube starts moving at about 200 ms, see Figures 140 and 141.



(a) Z-displacements (b) X-displacements

Figure 140: Some results of test IP3D14.



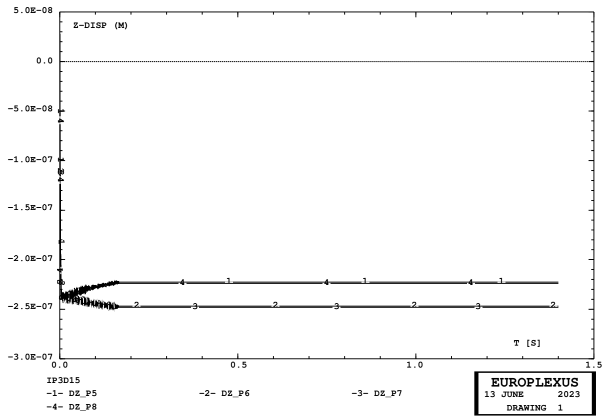
(a) $t = 0$ (b) $t = 1.4$ s

Figure 141: Deformed configuration in test IP3D14.

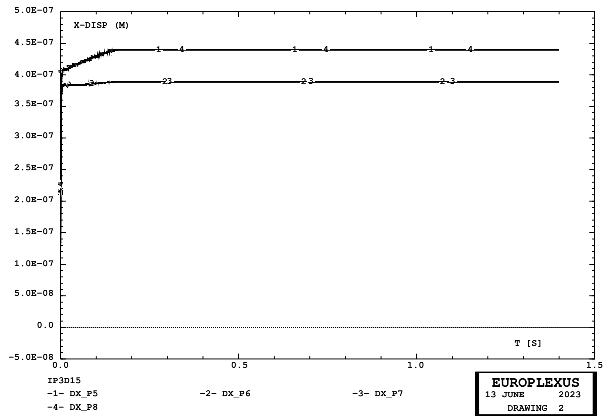
7.5.15 Case IP3D15

This test is similar to IP3D10 (GLIS with friction) but we apply the quasi-static damping until 500 ms instead of 10 ms, to see whether this is sufficient to block the cube.

The solution looks correct, since the cube does not move appreciably, see Figures 142 and 143.

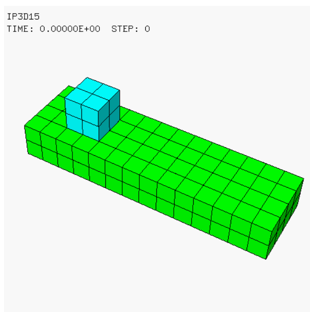


(a) Z-displacements

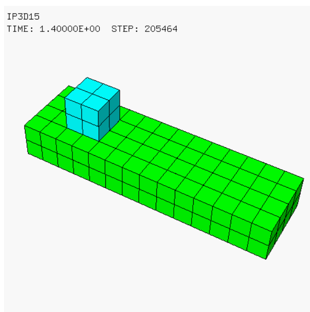


(b) X-displacements

Figure 142: Some results of test IP3D15.



(a) $t = 0$



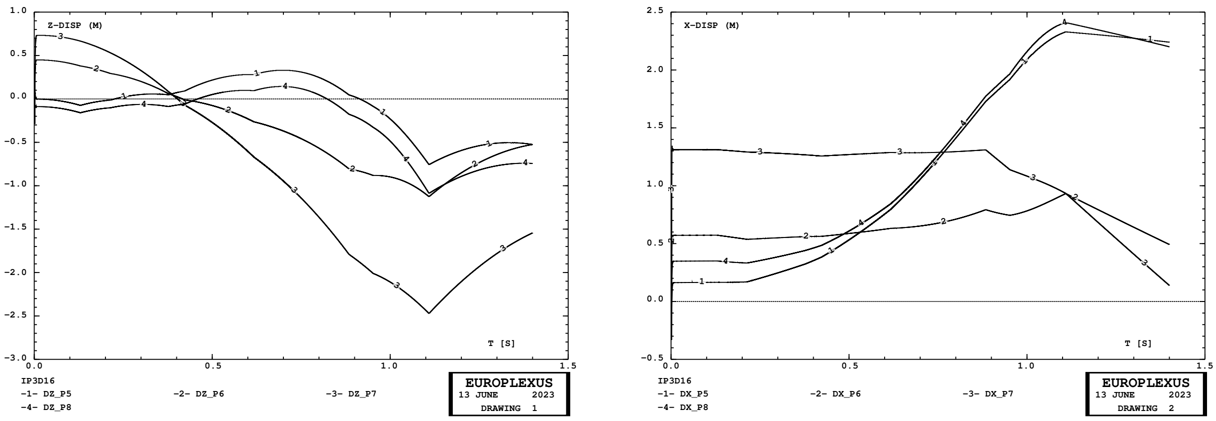
(b) $t = 1.4$ s

Figure 143: Deformed configuration in test IP3D15.

7.5.16 Case IP3D16

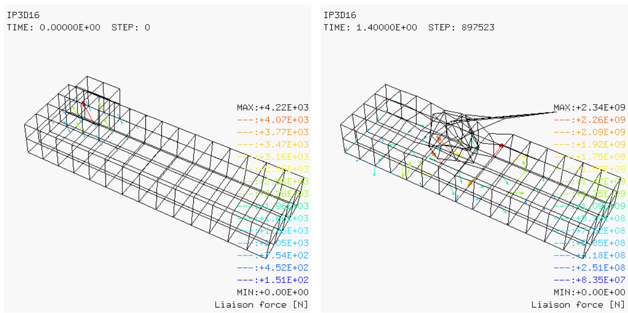
This test is similar to IP3D09 (PINB with friction) but we increase the friction coefficient to $\mu = 0.7$, to see whether this is sufficient to block the cube.

Unfortunately, the solution looks completely weird, since the cube undergoes large deformations and large motions, see Figures 144 and 145.



(a) Z-displacements (b) X-displacements

Figure 144: Some results of test IP3D16.



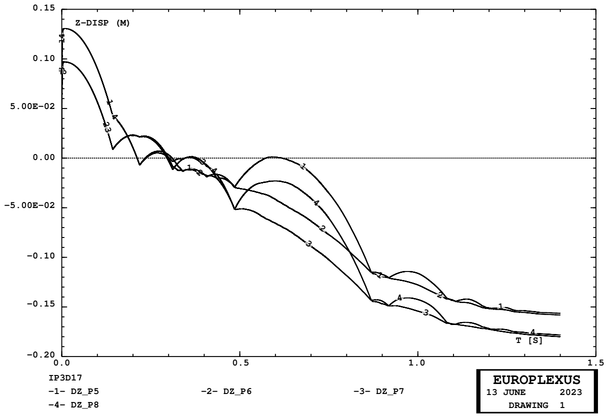
(a) $t = 0$ (b) $t = 1.4$ s

Figure 145: Deformed configuration in test IP3D16.

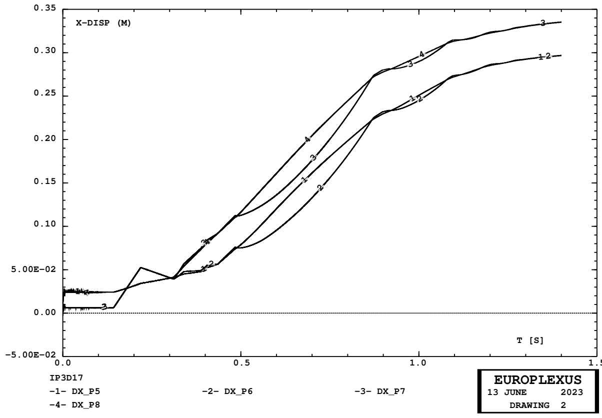
7.5.17 Case IP3D17

This test is similar to IP3D09 (PINB with friction) but we use the penalty formulation of the PINB model rather than the LM formulation, to see whether this is sufficient to block the cube. Hierarchic pinballs of level 5 are embedded also in the (bottom surface of the) cube, besides the top surface of the base.

The number of initial contacts is very large (36 864), but the simulation proceeds until the end (at a large CPU cost, though). However, the solution looks weird, since the cube initially jumps vertically under the effect of the penalty force, see Figures 146 and 147. This phenomenon could probably be avoided by slightly raising the cube in the initial configuration so that there is no (or almost no) initial penetration.

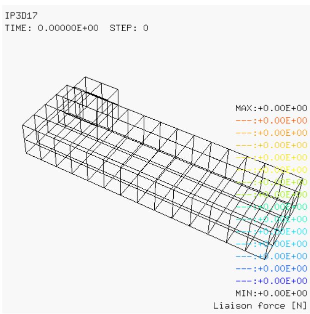


(a) Z-displacements

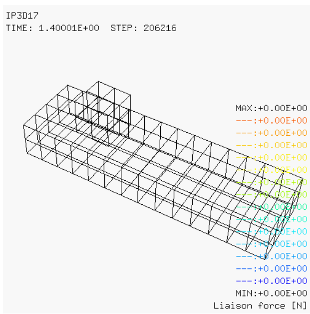


(b) X-displacements

Figure 146: Some results of test IP3D17.



(a) $t = 0$



(b) $t = 1.4$ s

Figure 147: Deformed configuration in test IP3D17.

References

[1] EUROPLEXUS User's Manual, on-line version: http://europlexus.jrc.ec.europa.eu.

[2] Cast3m Software: http://www-cast3m.cea.fr/.

[3] J.O. Hallquist, G.L. Goudreau, D.J. Benson. *Sliding interfaces with contact-impact in large-scale Lagrangian computation.* Computer Methods in Applied Mechanics and Engineering **51** : 107–137, 1985.

[4] D.J. Benson, J.O. Hallquist. *A single surface contact algorithm for the postbuckling analysis of shell structures.* Report to the University of California at San Diego, CA, 1987.

[5] T. Belytschko, S.E. Law. *An Assembled Surface Normal Algorithm for Interior Node Removal in Three-Dimensional Finite Element Meshes.* Engineering with Computers **1** : 55–60, 1985.

[6] T. Belytschko, J.I. Lin. *A Three-Dimensional Impact-Penetration Algorithm with Erosion.* Computers and Structures **25** : 95–104, 1987.

[7] T. Belytschko, M.O. Neal. *Contact-Impact by the Pinball Algorithm with Penalty, Projection and Lagrangian Methods.* Computational Techniques for Contact, Impact, Penetration and Perforation of Solids, Winter Annual Meeting of the American Society of Mechanical Engineers, San Francisco, California **AMD 103** : 97–140, December 10–15, 1989.

[8] T. Belytschko, M.O. Neal. *The Vectorized Pinball Contact Impact Routine.* SMiRT-10 Conference, 1989.

[9] R.E. Sarwas. *Hidden Line Elimination and a Modified Pinball Algorithm for Finite Element Contact-Impact Problems with Shell Elements.* Master's Thesis, Northwestern University, Evanston, Illinois, December 1989.

[10] T. Belytschko, M.O. Neal. *Contact-Impact by the Pinball Algorithm with Penalty and Lagrangian Methods.* International Journal for Numerical Methods in Engineering **31** : 547–572, 1991.

[11] T. Belytschko, I.S. Yeh. *The Splitting Pinball Method for Contact-Impact Problems.* Computer Methods in Applied Mechanics and Engineering **105** : 375–393, 1993.

[12] E. Plaskacz, T. Belytschko, H.Y. Chiang. *Contact-Impact Simulations on Massively Parallel SIMD Supercomputers.* Argonne National Laboratory Report ANL/CP-76686, 1993.

[13] M. Lepareux, B. Schwab, A. Hoffmann, P. Jamet, H. Bung. *Un programme général pour l'analyse dynamique rapide – Cas des tuyauteries.* Colloque: Tendances Actuelles en Calcul des Structures, Bastia, 6–8 November, 1985.

[14] R. Galon, H. Bung, M. Lepareux. *Programme PLEXUS. Algorithme de traitement des surfaces de glissement (3D).* CEA Rapport DEMT 90/092, 1990.

[15] H. Bung, M. Lepareux, R. Galon. *Programme PLEXUS. Prise en compte du frottement dans les lignes et surfaces de glissement.* CEA Rapport DEMT 90/203, 1990.

[16] J. Bonini, H. Bung. *Programme PLEXUS. Fiche théorique. Modélisation des impacts avec frottement par la méthode des multiplicateurs de Lagrange.* CEA Rapport DMT 97/367, 1997.

[17] J. Bonini. *Programme PLEXUS. Fiches de validation. Modélisation des impacts avec frottement par la méthode des multiplicateurs de Lagrange.* CEA Rapport DMT 97/368, 1997.

[18] P. Galon. *Calculs dynamiques d'un essai Charpy miniature avec prise en compte ou non du frottement.* CEA Rapport DMT/DEMT/DYN/RT/00-024/A, 2000.

[19] S. Moulin, S. Potapov. *Modélisation du contact dans EUROPLEXUS. Théorie et éléments de validation.* EDF Rapport HT-62/05/012/A, 2006.

[20] F. Casadei. *A Hierarchic Pinball Method for Contact-Impact in Fast Transient Dynamics.* VI Congresso Nazionale della Società Italiana di Matematica Applicata e Industriale (SIMAI 2002), Chia (Cagliari), Italy, 27–31 May 2002.
178 -- Pdf/2002/Simai2002/Simai2002.pdf

[21] F. Casadei. *A General Impact-Contact Algorithm Based on Hierarchic Pinballs for the EUROPLEXUS Software System.* Technical Note N. I.03.176, December 2003.
202 -- Pdf/2003/Pinballs/Pinballs.pdf

[22] F. Casadei. *Validation of the EUROPLEXUS Pinball Impact-Contact Model on an Indentation Problem.* Technical Note N. I.05.51, July 2005.
214 -- Pdf/2005/Indentation/Indentation.pdf

[23] F. Casadei. *Implementation of Fast Search Algorithms in EUROPLEXUS.* Technical Note, PUBSY No. JRC38086, October 2007.
230 -- Pdf/2007/Fast_Search/FastSearch.pdf

[24] F. Casadei, B. Langrand, M. Larcher, G. Valsamos. *Pinball-based Contact-Impact Model with Parabolic Elements in EUROPLEXUS.* Technical Note, PUBSY No. JRC89913, EUR Report 26629 EN, 2014.
314 -- Pdf/2014/Ortiz_Pinballs_Q9/Report/Pinballs_parabolic_JRC.pdf

[25] F. Casadei, M. Larcher, G. Valsamos, V. Faucher. *Implementation of Assembled Surface Normals and of a Penalty Contact Formulation in the Pinball Model of Europlexus.* Technical Note, PUBSY No. JRC90939, EUR 26714 EN, 2014.
316 -- Pdf/2014/Pinballs_ASN_Penalty/Report/Pinballs_ASN_Penalty_JRC.pdf

[26] F. Casadei. (D. Combescure and J. Ayneto Pou reviewers). *Numerical simulation examples of explosions and of contact-impact problems with EUROPLEXUS.* F4E-IDM Report F4E_D_2TL76L, 2015.
323 -- Pdf/2016/F4E_Reports/01_F4E_Examples.pdf

[27] F. Casadei, G. Valsamos, M. Larcher. *Testing of the GLIS contact model in EUROPLEXUS.* JRC Technical Note Pubsy N. JRC101012, EUR Report 27889 EN, 2016.
346 -- Pdf/2015/Glis_test/Report/Front/JRC101012_fullreportglisnew.pdf

[28] F. Casadei, V. Aune, G. Valsamos, M. Larcher. *Generalization of the pinball contact/impact model for use with mesh adaptivity and element erosion in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC101013, EUR Report 27888 EN, 2016.
351 -- Pdf/2016/ADAP_PINB_EROS/Report/Front/FullRepAdapPinb.pdf

[29] F. Casadei, G. Valsamos, M. Larcher, V. Aune. *Implementation of friction in the pinball-based contact-impact model of EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC112047, 2018.
365 -- Pdf/2016/PINB_Friction/Report/Front/Req_JRC112047.pdf

[30] F. Casadei, M. Larcher, G. Valsamos, V. Faucher. *Implementation of Assembled Surface Normals and of a Penalty Contact Formulation in the Pinball Model of Europlexus – Revision 1.* JRC Technical Report, PUBSY No. JRC111429, EUR Report 29217 EN, 2018.
367 -- Pdf/2016/Langrand_Article_DynaCell/New_Calculations_2016_12/Report/Front/Req_JRC111429.pdf

[31] F. Casadei, V. Aune, G. Valsamos, M. Larcher. *Contact detection by pseudo-nodal pinballs with mesh adaptivity in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC112048, 2018.
372 -- Pdf/2017/Vegard_Aune_2017/01_MirrorPlanes/Report/Front/Req_JRC112048.pdf

[32] V. Faucher, F. Casadei, G. Valsamos, M. Larcher. *An alternative definition of 3D shell bilateral contact conditions in the GLIS contact model of EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC118564, 2019.
399 -- Pdf/2019/VF_JRC_2019/Devel_Ispra_September2019/Folco/Report/Front/Req_JRC118564.pdf

[33] F. Casadei, G. Valsamos, M. Larcher, V. Faucher, A. Cambriani. *On some recent applications and developments concerning contact models in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC119981, 2020.
403 -- Pdf/2019/GV_Space_Battery/Report/Front/REQ_JRC119981.pdf

[34] F. Casadei, G. Valsamos, M. Larcher, V. Faucher. *MPI version of the decoupled pinball contact model in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRC123810, 2021.
413 -- Pdf/2020/PINB_DECO_MPI/Report/Front/Req_JRC123810.pdf

[35] F. Casadei, G. Valsamos, M. Larcher. *SLID : a sliding-line and sliding-surface algorithm for smooth contact in EUROPLEXUS.* JRC Technical Report, in preparation, 2021.
706 -- Pdf/2021/SLID/Report/SLID.pdf

[36] F. Casadei, G. Valsamos, M. Larcher. *A contact-impact model based on generalized pinballs in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRCxxxxxx, in preparation.
708 -- Pdf/2019/Gpinballs_2019/Report/Gpinballs.pdf
See also: Pdf/2014/Gpinballs/Report/Gpinballs.pdf
To be merged with: Pdf/2018/Gpinballs_2018/Report/Gpinballs.pdf

[37] F. Casadei, G. Valsamos, M. Larcher. *Treatment of friction in the PINB and GPIN contact-impact models of EUROPLEXUS.* JRC Technical Report, PUBSY No. JRCxxxxxx, in preparation.
709 -- Pdf/2020/GPIN_Friction/Report/GPIN_Friction.pdf

[38] F. Casadei, G. Valsamos, M. Larcher. *Treating side-on shell contact with penalty-based pinballs in EUROPLEXUS.* JRC Technical Report, PUBSY No. JRCxxxxxx, in preparation.
711 -- Pdf/2020/PINB_PENA_SHELL/Report/PINB_PENA_SHELL.pdf

[39] S. Potapov. *Correction de l'algorithme contact frottant 3D dans EUROPLEXUS.* Compte Rendu, EDF, CR-T61-2018-xxx, 2018.

Appendix I — Input files

All the input files used in the previous Sections are listed below.

c2el.epx

```
C2EL
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -10.0
     LECT el2 TERM
  VITE 2 0.0
     LECT el2 TERM
  DEPL 1 0.022
     LECT el2 TERM
  DEPL 2 -0.0
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI COOR DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 18
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  0.00000E+00 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE  0.00000E+00 TOLE 1.E-3
FIN
```

c2el03.epx

```
C2EL03
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -10.0
     LECT el2 TERM
  VITE 2 0.0
     LECT el2 TERM
  DEPL 1 0.022
     LECT el2 TERM
  DEPL 2 -0.0
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE NORE
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 18
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  0.00000E+00 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE  0.00000E+00 TOLE 1.E-3
FIN
```

c2el04.epx

```
C2EL04
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
```

```
         3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -9.659258262890683
     LECT el2 TERM
  VITE 2 2.5881904510252074
     LECT el2 TERM
  DEPL 1 0.021250368178359503
     LECT el2 TERM
  DEPL 2 -0.005694018992255456
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE NORE
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 11
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  8.17540E+06 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE -8.17540E+06 TOLE 1.E-3
FIN
```

c2el05.epx

```
C2EL05
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -10.0
     LECT el2 TERM
  VITE 2 0.0
     LECT el2 TERM
  DEPL 1 0.022
     LECT el2 TERM
  DEPL 2 -0.0
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE NORE
     PINB
       BODY MLEV 1
          LECT el1 TERM
       BODY MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 18
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  0.00000E+00 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE  0.00000E+00 TOLE 1.E-3
FIN
```

c2el06.epx

```
C2EL06
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -9.659258262890683
     LECT el2 TERM
  VITE 2 2.5881904510252074
     LECT el2 TERM
  DEPL 1 0.021250368178359503
     LECT el2 TERM
  DEPL 2 -0.005694018992255456
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE NORE
     PINB
       BODY MLEV 1
          LECT el1 TERM
       BODY MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 11
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.09597E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  2.40920E+06 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.09597E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE -2.40920E+06 TOLE 1.E-3
FIN
```

c2el2.epx

```
C2EL2
ECHO
```

```
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -10.0
     LECT el2 TERM
  VITE 2 0.0
     LECT el2 TERM
  DEPL 1 0.022
     LECT el2 TERM
  DEPL 2 -0.0
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 18
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  0.00000E+00 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.15873E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE  0.00000E+00 TOLE 1.E-3
FIN
```

c2el_b.epx

```
C2EL_B
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -9.659258262890683
     LECT el2 TERM
  VITE 2 2.5881904510252074
     LECT el2 TERM
  DEPL 1 0.021250368178359503
     LECT el2 TERM
  DEPL 2 -0.005694018992255456
     LECT el2 TERM
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP SPLT NONE
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI COOR DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 11
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  8.17540E+06 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE -8.17540E+06 TOLE 1.E-3
FIN
```

c2el_b2.epx

```
C2EL_B2
ECHO
!CONV WIN
NONL CPLA LAGC
GEOM LIBR POIN 8  Q42L 2 TERM
   0.1    0.1
   0.1    0.0
   0.1    0.1
   0.1    0.0
   0.0    0.1
   0.0    0.0
   0.2    0.1
   0.2    0.0
   1 2 6 5
   4 3 7 8
OPTI DUMP
COMP
 GROU 2
  'el1' LECT 1 TERM
  'el2' LECT 2 TERM
 NGROU 4
  'nc11' LECT 1 TERM
  'nc12' LECT 2 TERM
  'nc21' LECT 3 TERM
  'nc22' LECT 4 TERM
 COUL VERT LECT el1 TERM
      TURQ LECT el2 TERM
MATE
  VM23 RO 7800.0e3 YOUN 2.100000e+12 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e1
          3.50007e+10 1e2
    LECT el1 TERM
  VM23 RO 7800.0 YOUN 2.100000e+11 NU 0.280000
          ELAS 3.500070e+10
          TRAC 2
          3.50007e+10 0.0016667e2
          3.50007e+10 1e2
    LECT el2 TERM
INIT
  VITE 1 -9.659258262890683
     LECT el2 TERM
  VITE 2 2.5881904510252074
     LECT el2 TERM
  DEPL 1 0.021250368178359503
     LECT el2 TERM
  DEPL 2 -0.005694018992255456
     LECT el2 TERM
```

```
OPTI PINS DUMP STAT
     LNKS DUMP STAT
LINK COUP
     PINB
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el1 TERM
       BODY FROT MUST 0.9  MUDY 0.9  GAMM 0.0 MLEV 1
          LECT el2 TERM
OPTI NOTE CSTA 0.4
     LOG 1000
     PINS GRID DPIN 1.0001
  LIAJ ALPH 1
ECRI DEPL VITE ACCE FEXT FLIA FREQ 1
     FICH ALIT TFREQ 1e-5
          POIN nc11 nc12 nc21 nc22 TERM
     FICH ALIC TFREQ 1e-5
     FICH PVTK TFRE 1e-4
          GROU AUTO
          VARI DEPL ECRO VITE CONT FLIA DTST
CALC TINI 0 TEND 1e-3 NMAX 11
QUAL FLIA COMP 1 LECT nc11 TERM REFE -3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc11 TERM REFE  8.17540E+06 TOLE 1.E-3
     FLIA COMP 1 LECT nc21 TERM REFE  3.05110E+07 TOLE 1.E-3
     FLIA COMP 2 LECT nc21 TERM REFE -8.17540E+06 TOLE 1.E-3
FIN
```

ip2d01.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d01_mesh.ps';
opti sauv form 'ip2d01.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d01.epx

```
IP2D01
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
FIN
```

ip2d02.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d02_mesh.ps';
opti sauv form 'ip2d02.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d02.epx

```
IP2D02
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
```

```
         TURQ LECT cube TERM
         ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
         TRAC 2 2.0E10 0.01
                2.0E10 1.0
         LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
!OPTI PINS STAT
!    LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
              MLEV 5
              LECT pbas TERM
         BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
              DIAM 0.01
              LECT ncub TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
         NOEL
         POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
         POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'ip2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip2d03.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d03_mesh.ps';
opti sauv form 'ip2d03.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d03.epx

```
IP2D03
ECHO
 CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
         TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
         TRAC 2 2.0E10 0.01
                2.0E10 1.0
         LECT base cube TERM
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!             LECT pbas TERM
!        BODY DIAM 0.01
!             LECT ncub TERM
     GPIN BODY LECT base TERM
         BODY LECT cube TERM
         DIAM 0.01 LECT c3 c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
         NOEL
         POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
         POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
```

```
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'ip2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip2d04.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d04_mesh.ps';
opti sauv form 'ip2d04.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d04.epx

```
IP2D04
ECHO
 CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT c3 c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
!fin
*=======================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'ip2d03.pun' RENA 'dx_p5_03'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip2d05.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d05_mesh.ps';
opti sauv form 'ip2d05.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d05.epx

```
IP2D05
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     SLID 1 SURF EMAS LECT base TERM
                 NMAS LECT c3 TERM
                 NSLA LECT c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
```

```
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=======================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'ip2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip2d06.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d06_mesh.ps';
opti sauv form 'ip2d06.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d06.epx

```
IP2D06
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     SLID 1 SURF FROT MUST 0.1 MUDY 0.1 GAMM 0.0
                 EMAS LECT base TERM
                 NMAS LECT c3 TERM
                 NSLA LECT c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=======================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'ip2d05.pun' RENA 'dx_p5_05'
RCOU 25 'dx_p5' FICH 'ip2d04.pun' RENA 'dx_p5_04'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

ip2d07.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d07_mesh.ps';
opti sauv form 'ip2d07.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
```

```
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d07.epx

```
IP2D07
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub TERM
COMP EPAI 1.0  LECT base cube TERM
         0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
!RCOU 15 'dx_p5' FICH 'ip2d01.pun' RENA 'dx_p5_01'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip2d08.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d08_mesh.ps';
opti sauv form 'ip2d08.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d08.epx

```
IP2D08
ECHO
 CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT c3 c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
```

```
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
!RCOU 15 'dx_p5' FICH 'ip2d03.pun' RENA 'dx_p5_03'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip2d09.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'ip2d09_mesh.ps';
opti sauv form 'ip2d09.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
depl mesh tour -30.0 p2;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip2d09.epx

```
IP2D09
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     SLID 1 SURF FROT MUST 0.6 MUDY 0.6 GAMM 0.0
                 EMAS LECT base TERM
                 NMAS LECT c3 TERM
                 NSLA LECT c5 TERM
CHAR CONST GRAV 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-3
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-2
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   4.22493E+00  2.36603E+00  2.03274E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.22493E+00  2.36603E+00  0.00000E+00
!RSPHERE:  4.06549E+00
!RADIUS :  2.03274E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.58554E+01
!FAR    :  2.84584E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
!RCOU 15 'dx_p5' FICH 'ip2d05.pun' RENA 'dx_p5_05'
!RCOU 25 'dx_p5' FICH 'ip2d04.pun' RENA 'dx_p5_04'
!TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG VERT
FIN
```

ip3d01.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d01_mesh.ps';
opti sauv form 'ip3d01.msh';
lx = 7.0;
ly = 2.0;
```

```
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d01.epx

```
IP3D01
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d02.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d02_mesh.ps';
opti sauv form 'ip3d02.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
p2p = p2 plus (0 -1 0);
```

```
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d02.epx

```
IP3D02
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FEXT FLIA ! FREQ 1
                                 TFRE 0.01
          NOEL
          POIN LECT ncub TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     LIMA ON
SLER CAM1 1 NFRA 1
  FREQ 0 TFRE 1.E-2
! FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d03.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d03_mesh.ps';
opti sauv form 'ip3d03.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d03.epx

```
IP3D03
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
```

```
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d04.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d04_mesh.ps';
opti sauv form 'ip3d04.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d04.epx

```
IP3D04
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.1 MUDY 0.1 GAMM 0.0
            MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
```

```
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d05.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d05_mesh.ps';
opti sauv form 'ip3d05.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d05.epx

```
IP3D05
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
!OPTI GLIS NORM ELEM
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    GLIS 1 MAIT NODE LECT bas2 TERM
!           PESC      LECT cub1 TERM
     GPIN BODY LECT base TERM
          BODY LECT cube TERM
          DIAM 0.01 LECT bas2 cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'ip3d03.pun' RENA 'dx_p5_03'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip3d06.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
```

```
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d06_mesh.ps';
opti sauv form 'ip3d06.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d06.epx

```
IP3D06
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
!OPTI GLIS NORM ELEM
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    GLIS 1 FROT MUST 0.1 MUDY 0.1 GAMM 0.0
!           MAIT NODE LECT bas2 TERM
!           PESC      LECT cub1 TERM
     GPIN BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT bas2 cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'ip3d04.pun' RENA 'dx_p5_04'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

ip3d07.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d07_mesh.ps';
opti sauv form 'ip3d07.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
```

```
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d07.epx

```
IP3D07
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     SLID 1 SURF EMAS LECT pbas TERM
                 NMAS LECT bas2 TERM
                 NSLA LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'ip3d05.pun' RENA 'dx_p5_05'
RCOU 25 'dx_p5' FICH 'ip3d03.pun' RENA 'dx_p5_03'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

ip3d08.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d08_mesh.ps';
opti sauv form 'ip3d08.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d08.epx

```
IP3D08
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     SLID 1 SURF FROT MUST 0.1 MUDY 0.1 GAMM 0.0
                 EMAS LECT pbas TERM
                 NMAS LECT bas2 TERM
                 NSLA LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
```

```
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'ip3d06.pun' RENA 'dx_p5_06'
RCOU 25 'dx_p5' FICH 'ip3d04.pun' RENA 'dx_p5_04'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

ip3d09.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d09_mesh.ps';
opti sauv form 'ip3d09.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d09.epx

```
IP3D09
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FEXT FLIA ! FREQ 1
                                    TFRE 0.01
          NOEL
          POIN LECT ncub TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     LIMA ON
SLER CAM1 1 NFRA 1
  FREQ 0 TFRE 1.E-2
! FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
```

```
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d10.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d10_mesh.ps';
opti sauv form 'ip3d10.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d10.epx

```
IP3D10
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.6 MUDY 0.6 GAMM 0.0
            MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!       Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
        VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
        RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
        UP    4.99905E-02  5.85651E-01  8.09020E-01
        FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d11.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
```

```
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d11_mesh.ps';
opti sauv form 'ip3d11.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d11.epx

```
IP3D11
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
!OPTI GLIS NORM ELEM
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GPIN BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT bas2 cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'ip3d04.pun' RENA 'dx_p5_04'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d12.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d12_mesh.ps';
opti sauv form 'ip3d12.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
```

```
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d12.epx

```
IP3D12
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     SLID 1 SURF FROT MUST 0.6 MUDY 0.6 GAMM 0.0
                 EMAS LECT pbas TERM
                 NMAS LECT bas2 TERM
                 NSLA LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'ip3d06.pun' RENA 'dx_p5_06'
!RCOU 25 'dx_p5' FICH 'ip3d04.pun' RENA 'dx_p5_04'
!TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG VERT
FIN
```

ip3d13.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d13_mesh.ps';
opti sauv form 'ip3d13.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d13.epx

```
IP3D13
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
```

```
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.7 MUDY 0.7 GAMM 0.0
            MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d14.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d14_mesh.ps';
opti sauv form 'ip3d14.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d14.epx

```
IP3D14
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.6 MUDY 0.6 GAMM 0.0
            MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.1
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
```

```
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d15.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d15_mesh.ps';
opti sauv form 'ip3d15.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d15.epx

```
IP3D15
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI GLIS NORM ELEM
!OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.6 MUDY 0.6 GAMM 0.0
            MAIT NODE LECT bas2 TERM
            PESC      LECT cub1 TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.5
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
*=====================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-2
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d16.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
```

```
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d16_mesh.ps';
opti sauv form 'ip3d16.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d16.epx

```
IP3D16
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     MASS 0.0 LECT ncub TERM
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.7 MUDY 0.7 GAMM 0.0
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.7 MUDY 0.7 GAMM 0.0
               DIAM 0.01
               LECT ncub TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FEXT FLIA ! FREQ 1
                                    TFRE 0.01
          NOEL
          POIN LECT ncub TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     LIMA ON
SLER CAM1 1 NFRA 1
  FREQ 0 TFRE 1.E-2
! FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

ip3d17.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'ip3d17_mesh.ps';
opti sauv form 'ip3d17.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
```

```
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
p2p = p2 plus (0 -1 0);
depl mesh tour -30.0 p2 p2p;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

ip3d17.epx

```
IP3D17
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube TERM
COMP GROU 2 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
            'pcub' LECT cube TERM COND APPU LARG LECT cub1 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAD
LINK DECO
     BLOQ 123 LECT bas1 TERM
     PINB PENA SFAC 1.0
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.6 MUDY 0.6 GAMM 0.0
               MLEV 5
               LECT pcub TERM
CHAR CONST GRAV 0.0 0.0 -9.81 LECT cube TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FEXT FLIA ! FREQ 1
                                    TFRE 0.01
          NOEL
          POIN LECT ncub TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 1.40
!fin
*======================================================================
PLAY
CAME   1 EYE   1.57006E+01 -1.04295E+01  9.93321E+00
!        Q     8.33034E-01  4.34939E-01  3.14452E-01  1.34175E-01
         VIEW -6.40613E-01  6.40255E-01 -4.23897E-01
         RIGH  7.66235E-01  4.97078E-01 -4.07182E-01
         UP    4.99905E-02  5.85651E-01  8.09020E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  4.26466E+00  1.00000E+00  2.36603E+00
!RSPHERE:  4.25035E+00
!RADIUS :  1.78515E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.36011E+01
!FAR    :  2.63522E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
     LIMA ON
SLER CAM1 1 NFRA 1
  FREQ 0 TFRE 1.E-2
! FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 141 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 139 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc2d00.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d00_mesh.ps';
opti sauv form 'sc2d00.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d00.epx

```
SC2D00
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 0.0   PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
```

```
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
FIN
```

sc2d01.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d01_mesh.ps';
opti sauv form 'sc2d01.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d01.epx

```
SC2D01
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d02.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d02_mesh.ps';
opti sauv form 'sc2d02.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
```

```
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d02.epx

```
SC2D02
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d03.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d03_mesh.ps';
opti sauv form 'sc2d03.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d03.epx

```
SC2D03
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
```

```
        POIN LECT p5 p6 p7 p8 TERM
    FICH ALIT TFRE 1.E-5
        POIN LECT p5 p6 p7 p8 TERM
    FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d04.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d04_mesh.ps';
opti sauv form 'sc2d04.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d04.epx

```
SC2D04
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
        0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 YOUN 2.0E12 NU 0.3 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK DECO
     BLOQ 12 LECT c1 TERM
     PINB PENA
          BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d05.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d05_mesh.ps';
opti sauv form 'sc2d05.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
```

```
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d05.epx

```
SC2D05
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 YOUN 2.0E12 NU 0.3 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK DECO
     BLOQ 12 LECT c1 TERM
     PINB PENA
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d04.pun' RENA 'dx_p5_04'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d07.epx

```
SC2D07
ECHO
 CONV WIN
LAGR CPLA
GEOM LIBR POIN 18 Q42L 8 PMAT 2 CL2D 1 TERM
 0 0  1 0  2 0  3 0
 0 1  1 1  2 1  3 1
 0 2  1 2  2 2  3 2
 1 2  2 2
 1 3  2 3
 1 4  2 4
 1 2 6 5
 2 3 7 6
 3 4 8 7
 5 6 10 9
 6 7 11 10
 7 8 12 11
 13 14 16 15
 15 16 18 17
 13
 14
 18 17
COMP GROU 4 'base' LECT 1 PAS 1 6 TERM
            'cube' LECT 7 8 TERM
            'ncub' LECT 9 10 TERM
            'c7'   LECT 11 TERM
     EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT 4 5 6 TERM
     NGRO 3 'bloc' LECT 1 2 3 4 TERM
            'p5'   LECT 13 TERM
            'p6'   LECT 14 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT bloc TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
   ! QUAS STAT 100. 1.0 UPTO 0.01
REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00  2.00000E+00  1.08916E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  2.00000E+00  0.00000E+00
!RSPHERE:  2.72291E+00
!RADIUS :  1.08916E+01
!ASPECT :  1.00000E+00
!NEAR   :  8.16872E+00
!FAR    :  1.63374E+01
```

```
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
TRAC 1 2 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 AXES 1. 'Y-DISP (m)'
LIST 5 6 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc2d08.epx

```
SC2D08
ECHO
 CONV WIN
LAGR CPLA
GEOM LIBR POIN 18 Q42L 8 PMAT 2 CL2D 1 TERM
 0 0  1 0  2 0  3 0
 0 1  1 1  2 1  3 1
 0 2  1 2  2 2  3 2
 1 2  2 2
 1 3  2 3
 1 4  2 4
 1 2 6 5
 2 3 7 6
 3 4 8 7
 5 6 10 9
 6 7 11 10
 7 8 12 11
 13 14 16 15
 15 16 18 17
 13
 14
 18 17
COMP GROU 4 'base' LECT 1 PAS 1 6 TERM
            'cube' LECT 7 8 TERM
            'ncub' LECT 9 10 TERM
            'c7'   LECT 11 TERM
     EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT 4 5 6 TERM
     NGRO 3 'bloc' LECT 1 2 3 4 TERM
            'p5'   LECT 13 TERM
            'p6'   LECT 14 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE ! NORE
     BLOQ 12 LECT bloc TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA ! FREQ 1
                              TFRE 0.001
          NOEL
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00  2.00000E+00  1.08916E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  2.00000E+00  0.00000E+00
!RSPHERE:  2.72291E+00
!RADIUS :  1.08916E+01
!ASPECT :  1.00000E+00
!NEAR   :  8.16872E+00
!FAR    :  1.63374E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA
SLER CAM1 1 NFRA 1
FREQ ! 1
       0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
TRAC 1 2 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 AXES 1. 'Y-DISP (m)'
LIST 5 6 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc2d09.epx

```
SC2D09
ECHO
!CONV WIN
LAGR CPLA
GEOM LIBR POIN 18 Q42L 8 PMAT 2 CL2D 1 TERM
 0 0  1 0  2 0  3 0
 0 1  1 1  2 1  3 1
 0 2  1 2  2 2  3 2
 1 2  2 2
 1 3  2 3
 1 4  2 4
 1 2 6 5
 2 3 7 6
 3 4 8 7
 5 6 10 9
 6 7 11 10
 7 8 12 11
 13 14 16 15
 15 16 18 17
 13
 14
 18 17
COMP GROU 4 'base' LECT 1 PAS 1 6 TERM
            'cube' LECT 7 8 TERM
            'ncub' LECT 9 10 TERM
            'c7'   LECT 11 TERM
     EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT 4 5 6 TERM
     NGRO 3 'bloc' LECT 1 2 3 4 TERM
            'p5'   LECT 13 TERM
            'p6'   LECT 14 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE NORE
     BLOQ 12 LECT bloc TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
```

```
               DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA ! FREQ 1
                                TFRE 0.001
          NOEL
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00  2.00000E+00  1.08916E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  2.00000E+00  0.00000E+00
!RSPHERE:  2.72291E+00
!RADIUS :  1.08916E+01
!ASPECT :  1.00000E+00
!NEAR   :  8.16872E+00
!FAR    :  1.63374E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA
SLER CAM1 1 NFRA 1
FREQ ! 1
       0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
TRAC 1 2 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 AXES 1. 'Y-DISP (m)'
LIST 5 6 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d08.pun' RENA 'dx_p5_08'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d10.epx

```
SC2D10
ECHO
!CONV WIN
LAGR CPLA
GEOM LIBR POIN 18 Q42L 8 PMAT 2 CL2D 1 TERM
 0 0  1 0  2 0  3 0
 0 1  1 1  2 1  3 1
 0 2  1 2  2 2  3 2
 1 2  2 2
 1 3  2 3
 1 4  2 4
 1 2 6 5
 2 3 7 6
 3 4 8 7
 5 6 10 9
 6 7 11 10
 7 8 12 11
 13 14 16 15
 15 16 18 17
 13
 14
 18 17
COMP GROU 4 'base' LECT 1 PAS 1 6 TERM
            'cube' LECT 7 8 TERM
            'ncub' LECT 9 10 TERM
            'c7'   LECT 11 TERM
     EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT 4 5 6 TERM
     NGRO 3 'bloc' LECT 1 2 3 4 TERM
            'p5'   LECT 13 TERM
            'p6'   LECT 14 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE ! NORE
     BLOQ 12 LECT bloc TERM
     PINB BODY ! FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               MLEV 5
               LECT pbas TERM
          BODY ! FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA ! FREQ 1
                                TFRE 0.001
          NOEL
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT 5 PAS 1 14 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00  2.00000E+00  1.08916E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  2.00000E+00  0.00000E+00
!RSPHERE:  2.72291E+00
!RADIUS :  1.08916E+01
!ASPECT :  1.00000E+00
!NEAR   :  8.16872E+00
!FAR    :  1.63374E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA
SLER CAM1 1 NFRA 1
FREQ ! 1
       0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
TRAC 1 2 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 AXES 1. 'Y-DISP (m)'
LIST 5 6 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d08.pun' RENA 'dx_p5_08'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d11.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d11_mesh.ps';
opti sauv form 'sc2d11.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
```

```
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d11.epx

```
SC2D11
ECHO
 CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
         0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY LECT base TERM
          BODY LECT cube TERM
          DIAM 0.01 LECT c3 c5 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d12.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d12_mesh.ps';
opti sauv form 'sc2d12.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d12.epx

```
SC2D12
ECHO
 CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
         0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT c3 c5 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
```

```
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d11.pun' RENA 'dx_p5_11'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d13.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d13_mesh.ps';
opti sauv form 'sc2d13.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d13.epx

```
SC2D13
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
FONC NUM 1 TABL 2 0.0 1.0
                1.0 1.0
     NUM 2 TABL 4 0.0    0.0
                0.01    0.0
                0.01001 1.0
                1.0     1.0
!OPTI PINS ASN
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     SLID 1 SURF EMAS LECT base TERM
                 NMAS LECT c3 TERM
                 NSLA LECT c5 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc2d14.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2d14_mesh.ps';
opti sauv form 'sc2d14.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
```

```
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2d14.epx

```
SC2D14
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT c8 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI PINS ASN
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     SLID 1 SURF FROT MUST 0.1 MUDY 0.1 GAMM 0.0
                EMAS LECT base TERM
                NMAS LECT c3 TERM
                NSLA LECT c5 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc2d12.pun' RENA 'dx_p5_12'
RCOU 25 'dx_p5' FICH 'sc2d13.pun' RENA 'dx_p5_13'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

sc2dqs.dgibi

```
opti echo 1;
opti dime 2 elem qua4;
opti trac psc ftra 'sc2dqs_mesh.ps';
opti sauv form 'sc2dqs.msh';
lx = 7.0;
ly = 1.0;
den = 0.5;
dens den;
p1 = 0 0;
p2 = lx 0;
p3 = lx ly;
p4 = 0 ly;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
base = dall c1 c2 c3 c4 plan;
trac qual base;
p5 = 1.0 1.0;
p6 = 2.0 1.0;
p7 = 2.0 2.0;
p8 = 1.0 2.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cube = dall c5 c6 c7 c8 plan;
ncub = chan poi1 c5;
mesh = base et cube et ncub;
trac qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc2dqs.epx

```
SC2DQS
ECHO
!CONV WIN
CAST mesh
LAGR CPLA
GEOM Q42L base cube PMAT ncub CL2D c7 c8 TERM
COMP EPAI 1.0  LECT base cube TERM
          0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT c3 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 c8 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     IMPE PIMP RO 8000. PRES 0.0   PREF 0.0 FONC 2
          LECT c8 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 12 LECT c1 TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   3.50000E+00  1.00000E+00  2.06602E+01
!        Q     1.00000E+00  0.00000E+00  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  0.00000E+00 -1.00000E+00
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  1.00000E+00  0.00000E+00
```

```
        FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  0.00000E+00
!RSPHERE:  3.75641E+00
!RADIUS :  2.06602E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69038E+01
!FAR    :  2.81731E+01
SCEN GEOM NAVI FREE
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dy_p5' DEPL COMP 2 NOEU LECT p5 TERM
COUR 2 'dy_p6' DEPL COMP 2 NOEU LECT p6 TERM
COUR 3 'dy_p7' DEPL COMP 2 NOEU LECT p7 TERM
COUR 4 'dy_p8' DEPL COMP 2 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Y-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
FIN
```

sc3d00.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d00_mesh.ps';
opti sauv form 'sc3d00.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d00.epx

```
SC3D00
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 0.0   PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!    LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    PINB BODY MLEV 5
!             LECT pbas TERM
!         BODY DIAM 0.01
!             LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
```

```
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
COUR 9 'vx_p5' VITE COMP 1 NOEU LECT p5 TERM
COUR 10 'ax_p5' ACCE COMP 1 NOEU LECT p5 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
TRAC 9 AXES 1. 'X-VELO (m/s)' YZER
TRAC 10 AXES 1. 'X-ACCE (m/s2)' YZER
RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_2D_00'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d01.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d01_mesh.ps';
opti sauv form 'sc3d01.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d01.epx

```
SC3D01
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!    LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d02.dgibi

```
opti echo 0;
*
```

```
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d02_mesh.ps';
opti sauv form 'sc3d02.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d02.epx

```
SC3D02
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
        VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
        RIGH  9.05932E-01  4.09467E-01  1.07814E-01
        UP   -3.01151E-01  4.44099E-01  8.43851E-01
        FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d03.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
```

```
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d03_mesh.ps';
opti sauv form 'sc3d03.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d03.epx

```
SC3D03
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM CUB8 base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VMIS ISOT RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d04.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d04_mesh.ps';
opti sauv form 'sc3d04.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
```

```
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d04.epx

```
SC3D04
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM CUB8 base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VMIS ISOT RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
RCOU 25 'dx_p5' FICH 'sc2d02.pun' RENA 'dx_p5_2D_02'
RCOU 35 'dx_p5' FICH 'sc2d01.pun' RENA 'dx_p5_2D_01'
TRAC 5 15 25 35 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT TURQ
FIN
```

sc3d05.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                       : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d05_mesh.ps';
opti sauv form 'sc3d05.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d05.epx

```
SC3D05
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 YOUN 2.0E12 NU 0.3 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK DECO
     BLOQ 123 LECT bas1 TERM
     PINB PENA
          BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d06.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                    y1*'FLOTTANT' y2*'FLOTTANT'
                    z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d06_mesh.ps';
opti sauv form 'sc3d06.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d06.epx

```
SC3D06
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 YOUN 2.0E12 NU 0.3 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
```

```
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK DECO
     BLOQ 123 LECT bas1 TERM
     PINB PENA
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5 DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d05.pun' RENA 'dx_p5_05'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d07.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d07_mesh.ps';
opti sauv form 'sc3d07.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d07.epx

```
SC3D07
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI GLIS NORM ELEM
!OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GLIS 1 MAIT NODE LECT bas2 TERM
            PESC      LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
```

```
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d00.pun' RENA 'dx_p5_00'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d08.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                  y1*'FLOTTANT' y2*'FLOTTANT'
                  z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d08_mesh.ps';
opti sauv form 'sc3d08.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d08.epx

```
SC3D08
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI GLIS NORM ELEM
!OPTI PINS ASN
!OPTI PINS STAT
!     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GLIS 1 FROT MUST 0.3 MUDY 0.1 GAMM 0.5
            MAIT NODE LECT bas2 TERM
            PESC      LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
```

```
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=========================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d07.pun' RENA 'dx_p5_07'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d09.epx

```
SC3D09
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 4 CL3D 2 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 33
 34
 35
 36
 37 39 40 38
 33 35 39 37
COMP GROU 6 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'pbas' LECT base TERM
            'ncub' LECT 11 12 13 14 TERM
            'cub5' LECT 15 TERM
            'cub6' LECT 16 TERM
     NGRO 6 'bas1' LECT 1 PAS 1 16 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 36 TERM
            'p8'   LECT 35 TERM
            'pout' LECT p5 p6 p7 p8 SUIT 22 23 26 27 TERM
     EPAI 0.01 LECT ncub TERM ! Only for visualization
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI DUMP
OPTI PINS ASN
OPTI PINS STAT ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               DIAM 0.01
               LECT ncub TERM
!LINK DECO
!     BLOQ 123 LECT bas1 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
REGI 'rbas' RESU LECT base TERM
     'rcub' RESU LECT cube TERM
ECRI COOR DEPL VITE ACCE FLIA FDEC ! FREQ 1
                                   TFRE 0.001
          NOEL
          POIN LECT pout TERM ! p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT pout TERM ! p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.060
*=========================================================================
PLAY
CAME   1 EYE   3.73835E+00 -6.52700E+00  6.56254E+00
!        Q     8.75236E-01  4.61569E-01  5.60163E-02  1.33330E-01
         VIEW -2.21137E-01  7.93027E-01 -5.67632E-01
         RIGH  9.58170E-01  2.85102E-01  2.50276E-02
         UP   -1.81681E-01  5.38353E-01  8.22902E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  1.01220E+01
!ASPECT :  1.00000E+00
!NEAR   :  7.22998E+00
!FAR    :  1.59060E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 61 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 59 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=========================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d10.epx

```
SC3D10
ECHO
!CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 4 CL3D 2 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
```

```
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 33
 34
 35
 36
 37 39 40 38
 33 35 39 37
COMP GROU 6 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'pbas' LECT base TERM
            'ncub' LECT 11 12 13 14 TERM
            'cub5' LECT 15 TERM
            'cub6' LECT 16 TERM
     NGRO 7 'bas1' LECT 1 PAS 1 16 TERM
            'bas2' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 36 TERM
            'p8'   LECT 35 TERM
            'pout' LECT p5 p6 p7 p8 SUIT 22 23 26 27 TERM
     EPAI 0.01 LECT ncub TERM ! Only for visualization
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI GLIS NORM ELEM
!OPTI DUMP
!OPTI PINS ASN
!OPTI PINS STAT ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     GLIS 1 FROT MUST 0.3 MUDY 0.1 GAMM 0.5
            MAIT NODE LECT bas2 TERM
            PESC      LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
   ! QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rbas' RESU LECT base TERM
!     'rcub' RESU LECT cube TERM
 REGI 'rba2' RESU POIN LECT bas2 TERM
      'rcu1' RESU POIN LECT ncub TERM
ECRI COOR DEPL VITE ACCE FLIA FDEC ! FREQ 1
                                   TFRE 0.001
          NOEL
          POIN LECT pout TERM ! p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT pout TERM ! p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.046
*=====================================================================
PLAY
CAME   1 EYE   3.73835E+00 -6.52700E+00  6.56254E+00
!        Q     8.75236E-01  4.61569E-01  5.60163E-02  1.33330E-01
         VIEW -2.21137E-01  7.93027E-01 -5.67632E-01
         RIGH  9.58170E-01  2.85102E-01  2.50276E-02
         UP   -1.81681E-01  5.38353E-01  8.22902E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  1.01220E+01
!ASPECT :  1.00000E+00
!NEAR   :  7.22998E+00
!FAR    :  1.59060E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 47 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 45 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc3d01.pun' RENA 'dx_p5_01'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d12.epx

```
SC3D12
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 4 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 33
 34
 35
 36
 37 39 40 38
COMP GROU 5 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'ncub' LECT 11 12 13 14 TERM
            'c7'   LECT 15 TERM
            'pbas' LECT 1 PAS 1 9 TERM
     EPAI 0.01 LECT ncub TERM ! Only for visualization
     NGRO 5 'bloc' LECT 1 PAS 1 16 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     PINB BODY MLEV 5
               LECT pbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
```

```
        VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
        RIGH  1.00000E+00  0.00000E+00  0.00000E+00
        UP    0.00000E+00  4.22618E-01  9.06308E-01
        FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d13.epx

```
SC3D12
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 4 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 33
 34
 35
 36
 37 39 40 38
COMP GROU 5 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'ncub' LECT 11 12 13 14 TERM
            'c7'   LECT 15 TERM
            'pbas' LECT 1 PAS 1 9 TERM
     EPAI 0.01 LECT ncub TERM ! Only for visualization
     NGRO 5 'bloc' LECT 1 PAS 1 16 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     PINB BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               MLEV 5
               LECT pbas TERM
          BODY FROT MUST 0.3 MUDY 0.1 GAMM 0.5
               DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*======================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d14.epx

```
SC3D14
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
```

```
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI GLIS NORM ELEM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     GLIS 1 MAIT NODE LECT nmai TERM
            PESC      LECT p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d15.epx

```
SC3D15
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI GLIS NORM ELEM
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     GLIS 1 FROT MUST 0.1 MUDY 0.1 GAMM 0.0
            MAIT NODE LECT nmai TERM
            PESC      LECT p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
```

```
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d16.epx

```
SC3D16
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 8 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1.030  2 1 1.030
 1 2 1.030  2 2 1.030
 1 1 2.030  2 1 2.030
 1 2 2.030  2 2 2.030
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 22
 23
 26
 27
 33
 34
 35
 36
 37 39 40 38
COMP GROU 5 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'nbas' LECT 11 12 13 14 TERM
            'ncub' LECT 15 16 17 18 TERM
            'c7'   LECT 19 TERM
     EPAI 0.06 LECT nbas TERM ! Only for visualization
          0.01 LECT ncub TERM ! Only for visualization
     NGRO 9 'bloc' LECT 1 PAS 1 16 TERM
            'p1'   LECT 22 TERM
            'p2'   LECT 23 TERM
            'p3'   LECT 26 TERM
            'p4'   LECT 27 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT nbas ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT nbas ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     PINB BODY DIAM 0.06
               LECT nbas TERM
          BODY DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p1 p2 p3 p4 p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p1 p2 p3 p4 p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*======================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d17.epx

```
SC3D17
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 PMAT 8 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1.030  2 1 1.030
 1 2 1.030  2 2 1.030
 1 1 2.030  2 1 2.030
 1 2 2.030  2 2 2.030
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 22
 23
 26
 27
 33
 34
 35
 36
 37 39 40 38
COMP GROU 5 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'nbas' LECT 11 12 13 14 TERM
            'ncub' LECT 15 16 17 18 TERM
            'c7'   LECT 19 TERM
     EPAI 0.06 LECT nbas TERM ! Only for visualization
          0.01 LECT ncub TERM ! Only for visualization
     NGRO 9 'bloc' LECT 1 PAS 1 16 TERM
            'p1'   LECT 22 TERM
```

```
          'p2'   LECT 23 TERM
          'p3'   LECT 26 TERM
          'p4'   LECT 27 TERM
          'p5'   LECT 33 TERM
          'p6'   LECT 34 TERM
          'p7'   LECT 35 TERM
          'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT nbas ncub TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
     MASS 0.0 LECT nbas ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI PINS ASN
OPTI PINS STAT
     LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     PINB BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               DIAM 0.06
               LECT nbas TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               DIAM 0.01
               LECT ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p1 p2 p3 p4 p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p1 p2 p3 p4 p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d18.epx

```
SC3D18
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
!OPTI GLIS NORM ELEM
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
!    GLIS 1 MAIT NODE LECT nmai TERM
!           PESC      LECT p5 p6 p7 p8 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY LECT base TERM
          BODY LECT cube TERM
          DIAM 0.01 LECT nmai p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=====================================================================
SUITE
Post-treatment
```

```
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d19.epx

```
SC3D19
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
!OPTI GLIS NORM ELEM
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
!    GLIS 1 MAIT NODE LECT nmai TERM
!           PESC      LECT p5 p6 p7 p8 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
     GPIN BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT nmai p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
         FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
!RCOU 15 'dx_p5' FICH 'sc2d00.pun' RENA 'dx_p5_00'
!TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
!COLO NOIR ROUG
FIN
```

sc3d20.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d20_mesh.ps';
opti sauv form 'sc3d20.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
```

```
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d20.epx

```
SC3D20
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI GLIS NORM ELEM
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
!    GLIS 1 MAIT NODE LECT bas2 TERM
!           PESC      LECT ncub TERM
     GPIN BODY LECT base TERM
          BODY LECT cube TERM
          DIAM 0.01 LECT bas2 ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d07.pun' RENA 'dx_p5_07'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d21.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d21_mesh.ps';
opti sauv form 'sc3d21.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
```

```
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d21.epx

```
SC3D21
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube PMAT ncub CL3D cub5 cub6 TERM
COMP EPAI 0.01 LECT ncub TERM ! Only for visualization
     GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          ROUG LECT ncub TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
     MASS 0.0 LECT ncub TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
!OPTI GLIS NORM ELEM
!OPTI PINS ASN
OPTI LNKS STAT
     GPNS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
!    PINB BODY MLEV 5
!              LECT pbas TERM
!         BODY DIAM 0.01
!              LECT ncub TERM
!    GLIS 1 MAIT NODE LECT bas2 TERM
!           PESC      LECT ncub TERM
     GPIN BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT base TERM
          BODY FROT MUST 0.1 MUDY 0.1 GAMM 0.0
               LECT cube TERM
          DIAM 0.01 LECT bas2 ncub TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=====================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d20.pun' RENA 'dx_p5_20'
TRAC 5 15 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG
FIN
```

sc3d22.epx

```
SC3D22
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
!    GLIS 1 MAIT NODE LECT nmai TERM
!           PESC      LECT p5 p6 p7 p8 TERM
     SLID 1 SURF EMAS LECT base TERM
                 NMAS LECT nmai TERM
                 NSLA LECT p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=====================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
```

```
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=========================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d18.pun' RENA 'dx_p5_18'
RCOU 25 'dx_p5' FICH 'sc3d14.pun' RENA 'dx_p5_14'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

sc3d23.epx

```
SC3D23
ECHO
 CONV WIN
LAGR TRID
GEOM LIBR POIN 40 C82L 10 CL3D 1 TERM
 0 0 0  1 0 0  2 0 0  3 0 0
 0 1 0  1 1 0  2 1 0  3 1 0
 0 2 0  1 2 0  2 2 0  3 2 0
 0 3 0  1 3 0  2 3 0  3 3 0
 0 0 1  1 0 1  2 0 1  3 0 1
 0 1 1  1 1 1  2 1 1  3 1 1
 0 2 1  1 2 1  2 2 1  3 2 1
 0 3 1  1 3 1  2 3 1  3 3 1
 1 1 1  2 1 1
 1 2 1  2 2 1
 1 1 2  2 1 2
 1 2 2  2 2 2
 1 2 6 5 17 18 22 21
 2 3 7 6 18 19 23 22
 3 4 8 7 19 20 24 23
 5 6 10 9 21 22 26 25
 6 7 11 10 22 23 27 26
 7 8 12 11 23 24 28 27
 9 10 14 13 25 26 30 29
 10 11 15 14 26 27 31 30
 11 12 16 15 27 28 32 31
 33 34 36 35 37 38 40 39
 37 39 40 38
COMP GROU 3 'base' LECT 1 PAS 1 9 TERM
            'cube' LECT 10 TERM
            'c7'   LECT 11 TERM
     NGRO 6 'bloc' LECT 1 PAS 1 16 TERM
            'nmai' LECT 17 PAS 1 32 TERM
            'p5'   LECT 33 TERM
            'p6'   LECT 34 TERM
            'p7'   LECT 35 TERM
            'p8'   LECT 36 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT c7 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT c7 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bloc TERM
     SLID 1 SURF FROT MUST 0.1 MUDY 0.1 GAMM 0.0
                 EMAS LECT base TERM
                 NMAS LECT nmai TERM
                 NSLA LECT p5 p6 p7 p8 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     QUAS STAT 100. 1.0 UPTO 0.01
!REGI 'rba2' RESU POIN LECT 5 PAS 1 12 TERM
!     'rcu1' RESU POIN LECT 13 PAS 1 16 TERM
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.001
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=========================================================================
PLAY
CAME   1 EYE   1.50000E+00 -6.36311E+00  4.48361E+00
!        Q     8.43391E-01  5.37300E-01  0.00000E+00  0.00000E+00
         VIEW  0.00000E+00  9.06308E-01 -4.22618E-01
         RIGH  1.00000E+00  0.00000E+00  0.00000E+00
         UP    0.00000E+00  4.22618E-01  9.06308E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  1.50000E+00  1.50000E+00  8.16987E-01
!RSPHERE:  2.89199E+00
!RADIUS :  8.67598E+00
!ASPECT :  1.00000E+00
!NEAR   :  5.78399E+00
!FAR    :  1.44600E+01
SCEN GEOM NAVI FREE
          FACE HFRO
     VECT SCCO FIEL FLIA SCAL A14
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube TERM  REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube TERM  REND
ENDP
*=========================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Y-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d19.pun' RENA 'dx_p5_19'
RCOU 25 'dx_p5' FICH 'sc3d15.pun' RENA 'dx_p5_15'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

sc3d24.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*--------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*--------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d24_mesh.ps';
opti sauv form 'sc3d24.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
```

```
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d24.epx

```
SC3D24
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube CL3D cub5 cub6 TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     SLID 1 SURF EMAS LECT base TERM
                 NMAS LECT bas2 TERM
                 NSLA LECT cub1 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d20.pun' RENA 'dx_p5_20'
RCOU 25 'dx_p5' FICH 'sc3d07.pun' RENA 'dx_p5_07'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

sc3d25.dgibi

```
opti echo 0;
*
'DEBPROC' pxextr3d m*'MAILLAGE' x1*'FLOTTANT' x2*'FLOTTANT'
                   y1*'FLOTTANT' y2*'FLOTTANT'
                   z1*'FLOTTANT' z2*'FLOTTANT';
*
*-------------------------------------------------
* Extracts from the 3D mesh m the elements whose nodes are
* located in the box [x1-x2,y1-y2,z1-z2].
*
* Input :
* -----
*     m                      : 3D mesh
*     x1, x2, y1, y2, z1, z2 : extremes of the box
* Output :
* ------
*     box : mesh contained in the box
*-------------------------------------------------
*
x = coor 1 m;
sx = x POIN COMP x1 x2;
y = coor 2 sx;
sy = y POIN COMP y1 y2;
z = coor 3 sy;
sz = z POIN COMP z1 z2;
box = m ELEM APPU STRI sz NOVE;
*
finproc box;
*
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'sc3d25_mesh.ps';
opti sauv form 'sc3d25.msh';
lx = 7.0;
ly = 2.0;
lz = 1.0;
den = 0.5;
dens den;
p1 = 0 0 0;
p2 = lx 0 0;
p3 = lx ly 0;
p4 = 0 ly 0;
c1 = d p1 p2;
c2 = d p2 p3;
c3 = d p3 p4;
c4 = d p4 p1;
bas1 = dall c1 c2 c3 c4 plan;
base = bas1 volu tran (0 0 1.0);
bas2 = face 2 base;
trac cach qual base;
trac cach qual bas2;
p5 = 1.0 0.5 1.0;
p6 = 2.0 0.5 1.0;
p7 = 2.0 1.5 1.0;
p8 = 1.0 1.5 1.0;
c5 = d p5 p6;
c6 = d p6 p7;
c7 = d p7 p8;
c8 = d p8 p5;
cub1 = dall c5 c6 c7 c8 plan;
cube = cub1 volu tran (0 0 1.0);
ecub = enve cube;
cub5 = pxextr3d ecub 0.99 2.01 0.49 1.51 1.99 2.01;
cub6 = pxextr3d ecub 0.99 1.01 0.49 1.51 0.99 2.01;
ncub = chan poi1 cub1;
trac cach qual cube;
trac cach qual (base et cube);
trac cach qual (base et cub5 et cub6);
trac cach qual cub5;
trac cach qual cub6;
mesh = base et cube et ncub;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

sc3d25.epx

```
SC3D25
ECHO
 CONV WIN
CAST mesh
LAGR TRID
GEOM C82L base cube CL3D cub5 cub6 TERM
COMP GROU 1 'pbas' LECT base TERM COND APPU LARG LECT bas2 TERM
     COUL VERT LECT base TERM
          TURQ LECT cube TERM
          JAUN LECT cub5 cub6 TERM
MATE VM23 RO 8000.0 YOUN 2.0E12 NU 0.3 ELAS 2.0E10
          TRAC 2 2.0E10 0.01
                 2.0E10 1.0
          LECT base cube TERM
     IMPE PIMP RO 8000. PRES 2.0E7 PREF 0.0 FONC 1
          LECT cub5 TERM
     IMPE PIMP RO 8000. PRES 1.4E7 PREF 0.0 FONC 2
          LECT cub6 TERM
FONC NUM 1 TABL 2 0.0 1.0
                  1.0 1.0
     NUM 2 TABL 4 0.0     0.0
                  0.01    0.0
                  0.01001 1.0
                  1.0     1.0
OPTI LNKS STAT
LINK COUP SPLT NONE
     BLOQ 123 LECT bas1 TERM
     SLID 1 SURF FROT MUST 0.1 MUDY 0.1 GAMM 0.0
                 EMAS LECT base TERM
                 NMAS LECT bas2 TERM
                 NSLA LECT cub1 TERM
OPTI NOTE CSTA 0.5
     LOG 1
     PINS GRID DPIN 1.0001
     QUAS STAT 100. 1.0 UPTO 0.01
ECRI COOR DEPL VITE ACCE FLIA TFRE 0.01
          NOEL
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIT TFRE 1.E-5
          POIN LECT p5 p6 p7 p8 TERM
     FICH ALIC TFRE 1.E-3
CALC TINI 0 TEND 0.085
!fin
*=======================================================================
PLAY
CAME   1 EYE   9.67981E+00 -1.55461E+01  1.19133E+01
!        Q     8.47889E-01  4.83787E-01  5.59730E-02  2.09526E-01
         VIEW -2.97649E-01  7.96940E-01 -5.25635E-01
         RIGH  9.05932E-01  4.09467E-01  1.07814E-01
         UP   -3.01151E-01  4.44099E-01  8.43851E-01
         FOV   2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER :  3.50000E+00  1.00000E+00  1.00000E+00
!RSPHERE:  3.77492E+00
!RADIUS :  2.07620E+01
!ASPECT :  1.00000E+00
!NEAR   :  1.69871E+01
!FAR    :  2.83119E+01
SCEN GEOM NAVI FREE
     LIMA ON
SLER CAM1 1 NFRA 1
FREQ 0 TFRE 1.E-3
TRAC OFFS FICH AVI NOCL NFTO 86 FPS 10 KFRE 10 COMP -1
     OBJE LECT base cube ncub TERM REND
GOTR LOOP 84 OFFS FICH AVI CONT NOCL
     OBJE LECT base cube ncub TERM REND
GO
TRAC OFFS FICH AVI CONT
     OBJE LECT base cube ncub TERM REND
ENDP
*=======================================================================
SUITE
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP AXTE 1. 't [s]'
COUR 1 'dz_p5' DEPL COMP 3 NOEU LECT p5 TERM
COUR 2 'dz_p6' DEPL COMP 3 NOEU LECT p6 TERM
COUR 3 'dz_p7' DEPL COMP 3 NOEU LECT p7 TERM
COUR 4 'dz_p8' DEPL COMP 3 NOEU LECT p8 TERM
COUR 5 'dx_p5' DEPL COMP 1 NOEU LECT p5 TERM
COUR 6 'dx_p6' DEPL COMP 1 NOEU LECT p6 TERM
COUR 7 'dx_p7' DEPL COMP 1 NOEU LECT p7 TERM
COUR 8 'dx_p8' DEPL COMP 1 NOEU LECT p8 TERM
TRAC 1 2 3 4 AXES 1. 'Z-DISP (m)' YZER
TRAC 5 6 7 8 AXES 1. 'X-DISP (m)' YZER
LIST 1 2 3 4 AXES 1. 'Z-DISP (m)'
LIST 5 6 7 8 AXES 1. 'X-DISP (m)'
RCOU 15 'dx_p5' FICH 'sc3d21.pun' RENA 'dx_p5_21'
RCOU 25 'dx_p5' FICH 'sc3d08.pun' RENA 'dx_p5_08'
TRAC 5 15 25 AXES 1. 'X-DISP (m)' YZER
COLO NOIR ROUG VERT
FIN
```

test01.epx

```
TEST01
ECHO
!CONV WIN
LAGR DPLA
GEOM LIBR POIN 12 Q41L 4 TERM
  0 0  1 0  2 0  2 0  3 0  4 0
  0 1  1 1  2 1  2 1  3 1  4 1
  1 2 8 7
  2 3 9 8
  4 5 11 10
  5 6 12 11
COMP EPAI 1. LECT TOUS TERM
     GROU 2 'proj' LECT 1 2 TERM
            'targ' LECT 3 4 TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT TOUS TERM
OPTI ! DUMP
     PINS STAT ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     PINB BODY DIAM 1.0 LECT 2 TERM
          BODY DIAM 1.0 LECT 3 TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test02.epx

```
TEST02
ECHO
!CONV WIN
LAGR DPLA
GEOM LIBR POIN 12 Q41L 4 TERM
  0 0  1 0  2 0  2 0  3 0  4 0
  0 1  1 1  2 1  2 1  3 1  4 1
  1 2 8 7
  2 3 9 8
  4 5 11 10
  5 6 12 11
COMP EPAI 1. LECT TOUS TERM
     GROU 2 'proj' LECT 1 2 TERM
            'targ' LECT 3 4 TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT TOUS TERM
OPTI ! DUMP
     PINS STAT ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     PINB BODY FROT MUST 0.5 MUDY 0.3 GAMM 1.D0 DIAM 1.0 LECT 2 TERM
          BODY FROT MUST 0.5 MUDY 0.3 GAMM 1.D0 DIAM 1.0 LECT 3 TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test03.epx

```
TEST03
ECHO
!CONV WIN
LAGR TRID
GEOM LIBR POIN 9 CUB8 1 PMAT 1 TERM
  0 0 0  1 0 0  1 1 0  0 1 0
  0 0 1  1 0 1  1 1 1  0 1 1
  1 0.5 0.5
  1 2 3 4 5 6 7 8
  9
COMP EPAI 1. LECT 2 TERM
     GROU 2 'proj' LECT 1 TERM
            'targ' LECT 2 TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     GLIS 1 MAIT LECT proj TERM
            PESC LECT targ TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test04.epx

```
TEST04
ECHO
!CONV WIN
LAGR TRID
GEOM LIBR POIN 17 CUB8 3 PMAT 1 TERM
 0 0 0  1 0 0  0 1 0  1 1 0  0 2 0  1 2 0  0 3 0  1 3 0
 0 0 1  1 0 1  0 1 1  1 1 1  0 2 1  1 2 1  0 3 1  1 3 1
 1 1.5 0.5
 1 2 4 3 9 10 12 11
 3 4 6 5 11 12 14 13
 5 6 8 7 13 14 16 15
 17
COMP GROU 2 'proj' LECT 1 2 3 TERM
            'targ' LECT 4 TERM
     EPAI 1. LECT targ TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     GLIS 1 MAIT LECT proj TERM
            PESC LECT targ TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test05.dgibi

```
opti echo 0;
'DEBPROC' pxbox3d x0*'FLOTTANT' y0*'FLOTTANT' z0*'FLOTTANT'
                  lx*'FLOTTANT' ly*'FLOTTANT' lz*'FLOTTANT'
                  dd*'FLOTTANT';
*
*--------------------------------------------------
* Generates a parallelepiped mesh with origin in point
* (x0,y0,z0), sides of length (lx,ly,lz) and density (mesh size) dd.
* The mesh consists of CUB8 hexahedral elements and is oriented
* along the global axes.
*
* Input :
* -----
*     x0,y0,z0 : coordinates of 'origin' of the box
*     lx,ly,lz : length of the box sides
*     dd : "density" (size) of the mesh (the same in all directions)
* Output :
* ------
*     box : mesh consisting of CUB8 hexahedra
*--------------------------------------------------
*
dens dd;
p1 = x0 y0 z0;
p2 = (x0 + lx) y0 z0;
p3 = (x0 + lx) (y0 + ly) z0;
p4 = x0 (y0 + ly) z0;
*
c1 = p1 d p2;
c2 = p2 d p3;
c3 = p3 d p4;
c4 = p4 d p1;
base = dall c1 c2 c3 c4 plan;
*
box = base volu tran (0 0 lz);
*
finproc box;
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'test05_mesh.ps';
opti sauv form 'test05.msh';
x0 = 0.0;
y0 = 0.0;
z0 = 0.0;
lx = 2.0;
ly = 3.0;
lz = 3.0;
dd = 1.0;
proj = pxbox3d x0 y0 z0 lx ly lz dd;
ptar = lx (0.5*ly) (0.5*lz);
targ = manu poi1 ptar;
mesh = proj et targ;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

test05.epx

```
TEST05
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM CUB8 proj PMAT targ TERM
COMP EPAI 1. LECT targ TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     GLIS 1 MAIT LECT proj TERM
            PESC LECT targ TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test06.dgibi

```
opti echo 0;
'DEBPROC' pxbox3d x0*'FLOTTANT' y0*'FLOTTANT' z0*'FLOTTANT'
                  lx*'FLOTTANT' ly*'FLOTTANT' lz*'FLOTTANT'
                  dd*'FLOTTANT';
*
*--------------------------------------------------
* Generates a parallelepiped mesh with origin in point
* (x0,y0,z0), sides of length (lx,ly,lz) and density (mesh size) dd.
* The mesh consists of CUB8 hexahedral elements and is oriented
* along the global axes.
*
* Input :
* -----
*     x0,y0,z0 : coordinates of 'origin' of the box
*     lx,ly,lz : length of the box sides
*     dd : "density" (size) of the mesh (the same in all directions)
* Output :
* ------
*     box : mesh consisting of CUB8 hexahedra
*--------------------------------------------------
*
dens dd;
p1 = x0 y0 z0;
p2 = (x0 + lx) y0 z0;
p3 = (x0 + lx) (y0 + ly) z0;
p4 = x0 (y0 + ly) z0;
*
c1 = p1 d p2;
c2 = p2 d p3;
c3 = p3 d p4;
c4 = p4 d p1;
base = dall c1 c2 c3 c4 plan;
*
box = base volu tran (0 0 lz);
*
finproc box;
opti echo 1;
opti dime 3 elem cub8;
opti trac psc ftra 'test06_mesh.ps';
opti sauv form 'test06.msh';
x0 = 0.0;
y0 = 0.0;
z0 = 0.0;
lx = 2.0;
ly = 3.0;
lz = 3.0;
dd = 1.0;
proj = pxbox3d x0 y0 z0 lx ly lz dd;
ptar = lx (0.5*ly) (0.5*lz);
targ = manu poi1 ptar;
mesh = proj et targ;
trac cach qual mesh;
tass mesh noop;
sauv form mesh;
fin;
```

test06.epx

```
TEST06
ECHO
!CONV WIN
CAST mesh
LAGR TRID
GEOM CUB8 proj PMAT targ TERM
COMP EPAI 1. LECT targ TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     GLIS 1 FROT MUST 0.5 MUDY 0.3 GAMM 1.D0
            MAIT LECT proj TERM
```

```
          PESC LECT targ TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test07.epx

```
TEST07
ECHO
!CONV WIN
LAGR TRID
GEOM LIBR POIN 9 CUB8 1 PMAT 1 TERM
  0 0 0  1 0 0  1 1 0  0 1 0
  0 0 1  1 0 1  1 1 1  0 1 1
  1 0.5 0.5
  1 2 3 4 5 6 7 8
  9
COMP EPAI 1. LECT 2 TERM
     GROU 2 'proj' LECT 1 TERM
            'targ' LECT 2 TERM
     NGRO 1 'cibl' LECT 2 3 6 7 TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     IMPA DDL 1 COTE -1
          PROJ LECT targ TERM
          CIBL LECT cibl TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

test08.epx

```
TEST08
ECHO
!CONV WIN
LAGR TRID
GEOM LIBR POIN 9 CUB8 1 PMAT 1 TERM
  0 0 0  1 0 0  1 1 0  0 1 0
  0 0 1  1 0 1  1 1 1  0 1 1
  1 0.5 0.5
  1 2 3 4 5 6 7 8
  9
COMP EPAI 1. LECT 2 TERM
     GROU 2 'proj' LECT 1 TERM
            'targ' LECT 2 TERM
     NGRO 1 'cibl' LECT 2 3 6 7 TERM
     COUL TURQ LECT proj TERM
          VERT LECT targ TERM
MATE VM23 RO 8000. YOUN 1.D11 NU 0.0 ELAS 2.D8
          TRAC 3 2.D8 2.D-3 3.D8 1. 3.1D8 2.
          LECT proj TERM
     MASS 1000.0
          LECT targ TERM
OPTI ! DUMP
     LNKS STAT ! DUMP
LINK COUP SPLT NONE
     IMPA DDL 1 COTE -1
          FROT MUST 0.5 MUDY 0.3 GAMM 1.0
          PROJ LECT targ TERM
          CIBL LECT cibl TERM
INIT VITE 1  100 LECT proj TERM
     VITE 1 -100 LECT targ TERM
ECRI DEPL VITE ACCE FLIA FREQ 1
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 0.1D0 NMAX 10
FIN
```

List of input files

GETTING IN TOUCH WITH THE EU

In person

All over the European Union there are hundreds of Europe Direct centres. You can find the address of the centre nearest you online (european-union.europa.eu/contact-eu/meet-us_en).

On the phone or in writing

Europe Direct is a service that answers your questions about the European Union. You can contact this service:

— by freephone: 00 800 6 7 8 9 10 11 (certain operators may charge for these calls),

— at the following standard number: +32 22999696,

— via the following form: european-union.europa.eu/contact-eu/write-us_en.

FINDING INFORMATION ABOUT THE EU

Online

Information about the European Union in all the official languages of the EU is available on the Europa website (european-union.europa.eu).

EU publications

You can view or order EU publications at op.europa.eu/en/publications. Multiple copies of free publications can be obtained by contacting Europe Direct or your local documentation centre (european-union.europa.eu/contact-eu/meet-us_en).

EU law and related documents

For access to legal information from the EU, including all EU law since 1951 in all the official language versions, go to EUR-Lex (eur-lex.europa.eu).

Open data from the EU

The portal data.europa.eu provides access to open datasets from the EU institutions, bodies and agencies. These can be downloaded and reused for free, for both commercial and non-commercial purposes. The portal also provides access to a wealth of datasets from European countries.

Science for policy

The Joint Research Centre (JRC) provides independent, evidence-based knowledge and science, supporting EU policies to positively impact society

EU Science Hub
joint-research-centre.ec.europa.eu

@EU_ScienceHub

EU Science Hub - Joint Research Centre

EU Science, Research and Innovation

EU Science Hub

@eu_science

Publications Office
of the European Union