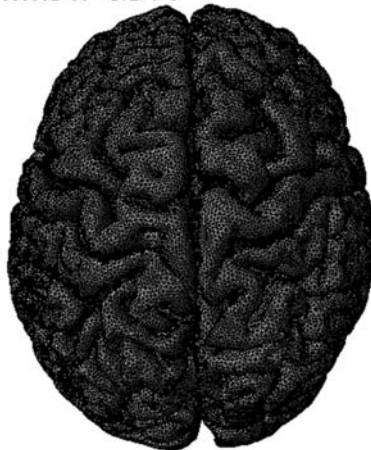


## JRC TECHNICAL REPORTS

# Simulation of rat head deceleration with EUROPLEXUS

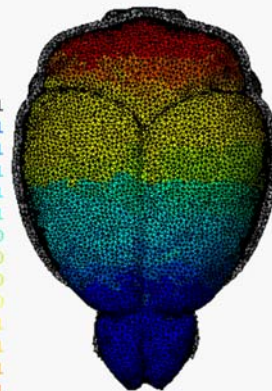
Casadei, F.  
Saez, P.  
Diez, P.  
Larcher, M.  
Valsamos, G.

HUMA01  
TIME: 0.00000E+00 STEP: 0



2018  
BRAT12  
TIME: 4.00034E-03 STEP: 7357

MAX: +7.59E+01  
---: +3.25E+01  
---: +2.75E+01  
---: +2.25E+01  
---: +1.75E+01  
---: +1.25E+01  
---: +7.50E+00  
---: +2.50E+00  
---: -2.50E+00  
---: -7.50E+00  
---: -1.25E+01  
---: -1.75E+01  
---: -2.25E+01  
---: -2.75E+01  
---: -3.25E+01  
MIN: -4.91E+01  
CONT 2 [Pa]



This publication is a Technical report by the Joint Research Centre (JRC), the European Commission's science and knowledge service. It aims to provide evidence-based scientific support to the European policymaking process. The scientific output expressed does not imply a policy position of the European Commission. Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use that might be made of this publication.

**Contact information**

Name: Georgios  
Address: Valsamos  
Email: [georgios.valsamos@ec.europa.eu](mailto:georgios.valsamos@ec.europa.eu)  
Tel.: +39-0332-789004

**JRC Science Hub**

<https://ec.europa.eu/jrc>

JRC106754

EUR 28643 EN

PDF ISBN 978-92-79-69552-0 ISSN 1831-9424 doi:10.2760/601191

Luxembourg: Publications Office of the European Union, 2018  
© European Union, 2018

The reuse of the document is authorised, provided the source is acknowledged and the original meaning or message of the texts are not distorted. The European Commission shall not be held liable for any consequences stemming from the reuse.

How to cite this report: Author(s), *Title*, EUR, doi

All images © European Union 2018

# Simulation of rat head deceleration with EUROPLEXUS

F. Casadei<sup>1</sup>, P. Saez<sup>2</sup>, P. Díez<sup>2</sup>, M. Larcher<sup>3</sup>, and G. Valsamos<sup>3</sup>

<sup>1</sup>*Retired from JRC ELSA*  
<sup>2</sup>*UPC Barcelona*  
<sup>3</sup>*JRC Ispra*

May 12, 2016

## Contents

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Introduction</b>   | <b>3</b>  |
| <b>2</b> | <b>Mesh generation</b>  | <b>3</b>  |
| 2.1      | Converting the mesh for use in EPX . . . . .  | 4         |
| 2.1.1    | Embedding the mesh in an EPX file with GEOM LIBR . . . . .                                | 6         |
| 2.1.2    | Reading the mesh from a separate file with a fixed format . . . . .                       | 7         |
| 2.1.3    | Passing through Cast3m . . . . .  | 8         |
| 2.1.4    | Manipulating the mesh by Cast3m . . . . .   | 10        |
| 2.2      | Rat head meshes . . . . .   | 11        |
| 2.3      | Human head mesh . . . . .   | 13        |
| <b>3</b> | <b>Numerical simulations of the rat head</b>  | <b>16</b> |
| 3.1      | Preliminary 3D simulations with the first volumetric mesh . . . . .                       | 16        |
| 3.1.1    | Materials . . . . .   | 17        |
| 3.1.2    | Mesh quality measurements . . . . .   | 17        |
| 3.1.3    | Eroding the bad elements . . . . .  | 18        |
| 3.2      | New material data . . . . .   | 19        |
| 3.2.1    | Mooney-Rivlin material . . . . .  | 19        |
| 3.2.2    | Ogden material set 1 . . . . .  | 19        |
| 3.2.3    | Ogden material set 2 . . . . .  | 20        |
| 3.3      | Testing the new material parameters on simplified geometries . . . . .                    | 20        |
| 3.3.1    | Mooney-Rivlin test <b>head46</b> . . . . .  | 20        |
| 3.3.2    | Mooney-Rivlin consistency tests <b>head56</b> , <b>head66</b> and <b>head55</b> . . . . . | 20        |
| 3.3.3    | Ogden set 1 test <b>head47</b> . . . . .  | 22        |
| 3.3.4    | Ogden set 2 test <b>head48</b> . . . . .  | 23        |
| 3.4      | Preliminary 3D simulations with the second volumetric mesh . . . . .                      | 24        |
| 3.4.1    | Cases <b>brat06</b> , <b>brat07</b> and <b>brat08</b> . . . . .                           | 24        |
| 3.5      | Stability step in hyperelastic materials . . . . .  | 27        |
| 3.5.1    | Stability of a single tetrahedron (cases <b>delt01</b> to <b>delt04</b> ) . . . . .       | 27        |
| 3.5.2    | Stability in a bar impact test (cases <b>hype01</b> and <b>hype02</b> ) . . . . .         | 28        |
| 3.5.3    | Behaviour of a single hexahedron (cases <b>hype03</b> to <b>hype05</b> ) . . . . .        | 32        |
| 3.6      | Actual 3D simulations with the second volumetric mesh . . . . .                           | 36        |
| 3.6.1    | Case <b>brat09</b> . . . . .  | 38        |
| 3.6.2    | Cases <b>brat10</b> and <b>brat11</b> . . . . .   | 40        |
| 3.6.3    | Cases <b>brat12</b> and <b>brat13</b> . . . . .   | 41        |
| 3.7      | Head slice simulations . . . . .  | 44        |

|       |  |    |
|-------|--|----|
| 3.7.1 | Circular 2D slice testing in case <code>br2d01</code> . . . . .                          | 45 |
| 3.7.2 | Malfunctioning of the HYPE material in 2D, case <code>br2d00</code> . . . . .            | 45 |
| 3.7.3 | Circular 3D slice testing in cases <code>br3d01</code> and <code>br3d02</code> . . . . . | 45 |
| 3.7.4 | Full 3D slice testing in cases <code>brat21</code> and <code>brat22</code> . . . . .     | 46 |
| 3.7.5 | Full 3D slice testing in case <code>brat23</code> . . . . .                              | 47 |

|                   |           |
|-------------------|-----------|
| <b>References</b> | <b>49</b> |
|-------------------|-----------|

|                               |           |
|-------------------------------|-----------|
| <b>Appendix — Input files</b> | <b>50</b> |
|-------------------------------|-----------|

|                            |           |
|----------------------------|-----------|
| <b>List of input files</b> | <b>77</b> |
|----------------------------|-----------|

## List of Tables

|    |  |    |
|----|--|----|
| 1  | Mesh data for the rat head. . . . .  | 4  |
| 2  | Summary of EPX input files for the visualization of rat head meshes. . . . .   | 7  |
| 3  | Raw mesh data for the second rat head volumetric mesh. . . . .   | 13 |
| 4  | Second volumetric mesh for the rat head (file <code>head.msh</code> ). . . . .   | 13 |
| 5  | Mesh data for the human head. . . . .  | 13 |
| 6  | Named element groups in the human head mesh. . . . .   | 13 |
| 7  | Named node groups in the human head mesh. . . . .  | 13 |
| 8  | Minimum and maximum element sizes ( $\mu\text{m}$ ) of the human head components in case <code>huma01</code> . . . . . | 15 |
| 9  | Summary of nodes and elements in the human head mesh. . . . .  | 16 |
| 10 | Minimum and maximum element sizes ( $\mu\text{m}$ ) of the head components in case <code>brat04</code> . . . . .       | 18 |
| 11 | Details of the second volumetric mesh for the rat head (file <code>head.msh</code> ). . . . .                          | 25 |
| 12 | Minimum and maximum element sizes ( $\mu\text{m}$ ) of the head components in case <code>brat06</code> . . . . .       | 26 |

## List of Figures

|    |   |    |
|----|---|----|
| 1  | First surface mesh of the rat brain (from <code>brain_stl2inp.inp</code> ), EPX case <code>brat01</code> . . . . .    | 6  |
| 2  | First volumetric mesh of the rat brain (from <code>brain_and_csf.inp</code> ), EPX case <code>brat02</code> . . . . . | 7  |
| 3  | Mesh in case <code>brainv</code> . . . . .  | 10 |
| 4  | Mesh in case <code>brat04</code> . . . . .  | 11 |
| 5  | Rat brain anatomy (from [5]). . . . .   | 12 |
| 6  | Human head mesh (complete). . . . .   | 14 |
| 7  | Human brain mesh: grey and white matter (with and without element outlines). . . . .                                  | 14 |
| 8  | Human brain mesh: white matter (with element outlines). . . . .   | 15 |
| 9  | Human brain mesh: white matter (without element outlines). . . . .  | 15 |
| 10 | Most critical elements in the brain in case <code>brat05</code> . . . . .   | 18 |
| 11 | Spherical head impact test: comparison of Ogden (top row) and Mooney-Rivlin (bottom row) materials. . . . .           | 21 |
| 12 | Forces in the brain and in the CSF in cases <code>head56</code> (left) and <code>head66</code> (right). . . . .       | 21 |
| 13 | Forces in nominally stress-free cubes made of Mooney-Rivlin material. . . . .   | 22 |
| 14 | Comparison of contacts for Ogden [3] (top row) and Ogden type 1 (bottom row) materials. . . . .                       | 23 |
| 15 | Comparison of brain pressures for Ogden [3] (top row) and Ogden type 1 (bottom row) materials. . . . .                | 23 |
| 16 | Comparison of brain pressures for Ogden type 1 (top row) and Ogden type 2 (bottom row) materials. . . . .             | 24 |
| 17 | Time evolution of the skull velocity. . . . .   | 25 |
| 18 | Local velocity and pressure in the CSF in case <code>brat06</code> . . . . .  | 26 |
| 19 | Single tetrahedron element used in test <code>delt01</code> . . . . .   | 27 |
| 20 | Displacements and stresses vs. time in case <code>hype01</code> . . . . .   | 29 |

|    |  |    |
|----|--|----|
| 21 | Stress along the bar at $t = 50$ ms in case <b>hype02</b> . . . . .  | 30 |
| 22 | Stresses and velocities along the bar every 10 ms in case <b>hype02</b> . . . . .                          | 30 |
| 23 | Deformations near the impact zone (last 10 mm) in case <b>hype02</b> . . . . .                             | 31 |
| 24 | Deformations near the impact zone (last 50 mm) in case <b>hype02</b> . . . . .                             | 31 |
| 25 | Single hexahedron element used in test <b>hype03</b> . . . . .   | 32 |
| 26 | Time evolution of the upper face velocity and displacement. . . . .  | 33 |
| 27 | Initial and deformed configurations in test case <b>hype03</b> . . . . .                                   | 33 |
| 28 | Normal stresses and normal strains in test case <b>hype03</b> . . . . .                                    | 34 |
| 29 | Further results of test case <b>hype03</b> . . . . .   | 35 |
| 30 | Initial and deformed configurations in test case <b>hype04</b> . . . . .                                   | 35 |
| 31 | Normal stresses and normal strains in test case <b>hype04</b> . . . . .                                    | 36 |
| 32 | Further results of test case <b>hype04</b> . . . . .   | 36 |
| 33 | Normal stresses and normal strains in test case <b>hype05</b> . . . . .                                    | 37 |
| 34 | Further results of test case <b>hype05</b> . . . . .   | 37 |
| 35 | Stress $\sigma_{yy}$ in the rat head in case <b>brat09</b> . . . . .                                       | 39 |
| 36 | Iso-surfaces of stress $\sigma_{yy}$ in the rat head in case <b>brat09</b> . . . . .                       | 39 |
| 37 | Results at the centroid of the rat head in case <b>brat09</b> . . . . .                                    | 41 |
| 38 | Location of dummy material elements in case <b>brat11</b> . . . . .  | 41 |
| 39 | Comparison of stress distributions in the head at 20 ms in cases <b>brat09</b> and <b>brat11</b> . . . . . | 42 |
| 40 | Comparison of stress at the head centroid in cases <b>brat09</b> and <b>brat11</b> . . . . .               | 42 |
| 41 | Distribution of stress $\sigma_{yy}$ in the rat brain in case <b>brat12</b> . . . . .                      | 43 |
| 42 | Time evolution of the skull velocity in case <b>brat13</b> . . . . .                                       | 43 |
| 43 | Distribution of stress $\sigma_{yy}$ in the rat brain in case <b>brat13</b> . . . . .                      | 44 |
| 44 | Distribution of stress $\sigma_{xx}$ in a circular 2D slice in case <b>br2d01</b> . . . . .                | 45 |
| 45 | Distribution of stress $\sigma_{xx}$ in a circular 3D slice (hexahedra) in case <b>br3d01</b> . . . . .    | 46 |
| 46 | Distribution of stress $\sigma_{xx}$ in a circular 3D slice (prisms) in case <b>br3d02</b> . . . . .       | 46 |
| 47 | Distribution of stress $\sigma_{yy}$ in a 3D slice (tetrahedra) in case <b>brat21</b> . . . . .            | 48 |
| 48 | Distribution of stress $\sigma_{yy}$ in a 3D slice (tetrahedra) in case <b>brat23</b> . . . . .            | 48 |

## 1 Introduction

This report presents some preliminary calculations for the simulation of rat, monkey and human head behaviour (in particular of the brain) under imposed deceleration conditions by the EUROPLEXUS code.

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 (or to manipulate) complex meshes. If not otherwise stated, the latest (2016) version of the Cast3m software, obtainable from the site [2], is used in the following mesh manipulations.

The present report is the sequel to [3], which presented preliminary human head *impact* simulations on a rigid obstacle by EPX, using a simplified (3D spherical) geometry.

## 2 Mesh generation

The computational mesh is generated by an external pre-processor (PointWise<sup>®</sup> [4] or other commercial products). The geometrical data (nodal coordinates and elements connectivity) are contained in a text (ASCII) file using the so-called STL2 format. Such files have a .INP extension. Table 1 lists all the available .INP files for the rat head. The type column indicates the geometric elements used (with Cast3m names). The TRI3 are 3-node triangles used to represent surfaces while the TET4 are 4-node tetrahedra used to represent volumes.

For example, file `brain_stl2inp.inp` is as follows:

| File                    | N. of lines | Type | Nodes   | Elements | Description                      |
|-------------------------|-------------|------|---------|----------|----------------------------------|
| brain.stl2inp.inp       | 31 809      | TRI3 | 10 604  | 21 204   | First surface mesh (brain only)  |
| brain_alone_fromCAE.inp | 490 856     | TET4 | 75 922  | 414 932  | First volume mesh of the brain   |
| cf_d_alone_fromCAE.inp  | 200 348     | TET4 | 39 200  | 161 147  | First volume mesh of the CSF     |
| brain_and_csf.inp       | 691 203     | TET4 | 115 122 | 576 079  | Union of the previous two meshes |
| brain.inp               | 370 695     | TET4 | 57 938  | 312 755  | Second volume mesh of the brain  |
| csf.inp                 | 181 330     | TET4 | 35 582  | 145 746  | Second volume mesh of the CSF    |

Table 1: Mesh data for the rat head.

```

*Node
  1, -8.94110394, -9.3942976, 1.82074296
  2, -9.05773067, -9.39597225, 1.81760705
  . . .
 63552, -6.81360817, -7.89493799, 8.84523964
*Element, type=S3
  1, 1, 2, 3
  2, 1, 5, 6
  . . .
21203, 48274, 48522, 49482
21204, 48270, 48274, 49482

```

That is, the file starts with a text line `*Node` and then the list of nodes consisting of the node index followed by its coordinates, comma-separated (one node per line). The node indexes might be non-consecutive, i.e. there might be “holes” in the numbering. This must be fixed so that EPX can use the data.

The total number of nodes is not known a priori but it can be found by reading the list until a line starting with `*Element, type=` is encountered. Following this line, there is the list of elements connectivity. One element per line is listed and consists of the element index followed by the list of the element’s nodes (comma separated). The number of nodes of the element depends upon the element type, which can be read in the `*Element, type=` introductory line. In the above example, the `S3` type indicates a 3-noded triangle. Alternatively, `C3D4` indicates a 4-node tetrahedron. Like in the case of nodes, the numbering of elements is not necessarily consecutive. This has to be fixed, since EPX requires *consecutive* numbering starting at 1 for both the nodes and the elements.

The total number of elements is not known a priori but it can be found by reading the file until the end. It is not known how this format would deal with the presence of elements of more than one type, since we have no example thereof. Presumably, the elements would be grouped by type and each group would be preceded by an `*Element, type=` line.

## 2.1 Converting the mesh for use in EPX

A simple Fortran utility program `readstl2` is written, which reads the STL2 file and writes the mesh data in a form that can be used by EPX. This program (`readstl2.f`) is listed in the Appendix, together with all input files used in the present report.

In its present form, the utility can only recognize two element types in the STL2 format: the `S3` type (3-noded triangle) and the `C3D4` type (4-noded tetrahedron). The utility reads the STL2 file from standard input and writes the data to standard output according to the following format, which can be used by EPX:

```

*NPTL= 10604 NELEM= 21204
-0.89411E+01 -0.93943E+01 0.18207E+01
-0.90577E+01 -0.93960E+01 0.18176E+01
. . .
-0.70939E+01 -0.81860E+01 0.88329E+01
-0.68136E+01 -0.78949E+01 0.88452E+01
  1 2 3
  1 4 5
. . .
 8336 8369 8517
 8335 8336 8517

```

The first line contains the total number of nodes (`NPTL`) and the total number of elements (`NELEM`). This information is only for human checking. EPX does not need this line so it is commented out by starting it with a `*`, according to EPX's input syntax conventions.

Then comes the list of the nodal coordinates. Each line contains the coordinates of a node. The node index is *not* given because nodes are consecutively numbered in EPX. Also, no comma is inserted to separate the values. The Fortran reading format for this is (`3E13.5`).

Then comes the list of the element connectivity. Each line contains the list of the nodes of an element. The element index is *not* given because elements are consecutively numbered in EPX. Also, no comma is inserted to separate the values. The Fortran reading format for this is (`8I8`).

Any "holes" in the numbering of nodes and/or elements are removed by the `readstl2` utility, so that the nodes and elements are guaranteed to be consecutively numbered from 1 on in the output file.

The command to convert the data is, for example:

```
readstl2 <brain_stl2inp.inp >brainsur.txt
```

The above example concerned a surface mesh (the rat's brain surface), composed of triangles. The next example concerns a volumetric mesh (the rat's brain and CSF), made of tetrahedra. The brain mesh and the CSF mesh are conforming but distinct. In other words, to each node on the brain surface there corresponds a node with the same coordinates, but a *different index*, on the CSF internal surface. This allows to treat FSI by using the classical FSA algorithm available in EPX (see [1]).

The input file is `brain_and_csf.inp`:

```
*Node
  1, -11.1735859, -9.84484959,  4.82526159
  2, -11.1675835, -9.89240742,  4.67690754
  . . .
115121, -8.47570229, -4.81912327,  2.75857115
115122, -6.90700054, -5.70823336,  8.12640762
*Element, type=C3D4
  1, 13852, 13853, 13855, 13854
  2, 13856, 13857, 13852, 13858
  . . .
576078, 101966,  83697, 105331, 101967
576079, 101846, 101848, 101845,  94534
```

The filtering command is:

```
readstl2 <brain_and_csf.inp >brainvol.txt
```

The resulting mesh data (file `brainvol.txt`) is:

```
*NPTL= 115122 NELEM= 576079
-0.11174E+02 -0.98448E+01  0.48253E+01
-0.11168E+02 -0.98924E+01  0.46769E+01
. . .
-0.84757E+01 -0.48191E+01  0.27586E+01
-0.69070E+01 -0.57082E+01  0.81264E+01
  13852  13853  13855  13854
  13856  13857  13852  13858
. . .
101966  83697 105331 101967
101846 101848 101845  94534
```

In this case, the same element type (tetrahedron) is used in the two parts of the volumetric mesh (brain and CSF). In order to distinguish the two mesh zones (to assign the corresponding materials in EPX), we use the *verbal* information (from the person who provided the mesh data) that the brain consists of the first 414 932 elements (from element 1 to element 414 932), while the CSF consists of the remaining 161 147 elements (from element 414 933 to element 576 079). These indications are *not* contained in the STL2 file format.

### 2.1.1 Embedding the mesh in an EPX file with GEOM LIBR

The mesh file resulting from the `readstl2` filter can be used in several ways in EPX. The first possibility is to just embed the file within an EPX input file by using the `GEOM LIBR` (free-format) geometry definition. For example, in the case of the surface mesh initially considered above (files `brain_stl2inp.inp` and `brainsur.txt`), a simple EPX input file in order to just check that the mesh is correctly interpreted by EPX would be as follows (see file `brat01.epx` in Appendix):

```
BRAT01
ECHO
  CONV WIN
LAGR TRID
GEOM LIBR POIN 10604 T3GS 21204 TERM
*NPTL= 10604 NELEM= 21204
-0.89411E+01 -0.93943E+01 0.18207E+01
-0.90577E+01 -0.93960E+01 0.18176E+01
. . . (lines skipped for brevity)
-0.70939E+01 -0.81860E+01 0.88329E+01
-0.68136E+01 -0.78949E+01 0.88452E+01
  1 2 3
  1 4 5
. . . (lines skipped for brevity)
 8336 8369 8517
 8335 8336 8517
COMP GROU 1 'surf' LECT 1 PAS 1 21204 TERM
  EPAI 1. LECT surf TERM
  COUL GR50 LECT surf TERM
MATE LINE RO 8000. YOUN 1.D11 NU 0.3
  LECT SURF TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN
```

In the input file, the `GEOM LIBR` directive requires the definition of the total number of nodes (10 604 in this case) and of the element zones. Here we have used the `T3GS` triangular shell element to discretize the surface. There are exactly 21 204 triangular elements. The `brainsur.txt` file is inserted in the EPX file just after the `GEOM LIBR ... TERM` directive.

Then, the `COMP GROU` directive is used to assign a name (`surf`) to the entire mesh in this case, and the `COUL` sub-directive gives it a color (`GR50`, i.e. 50 % grey). A material *must* be assigned to the mesh. In this case we simply use the linear elastic material (`LINE`).

Now the mesh is ready for visualization directly from EPX. By typing interactively `trac rend` and then by adjusting the view parameters one obtains, for example, the views shown in Figure 1. The dorsal region of the mesh visible in Figure 1(d) contains very large and irregular triangles, so this mesh is not suitable for numerical simulations.

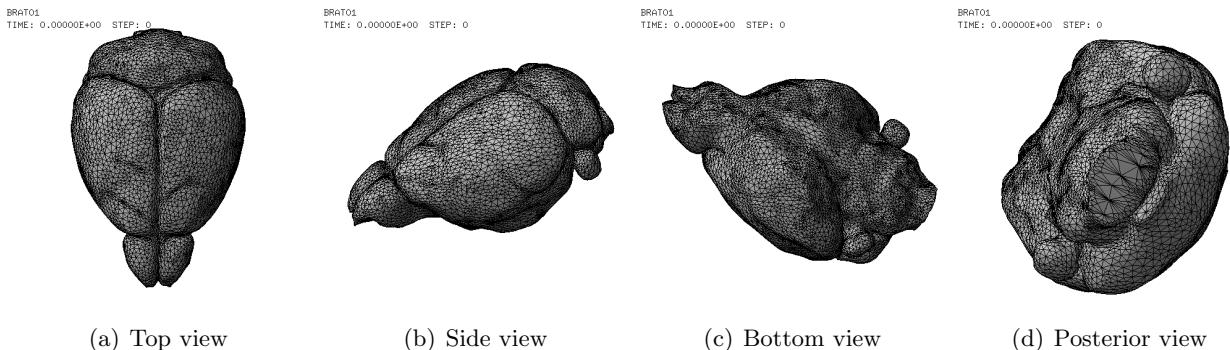


Figure 1: First surface mesh of the rat brain (from `brain_stl2inp.inp`), EPX case `brat01`.

A similar input file to check the first volumetric mesh of the rat's brain *and CSF* (from file `brain_and_csf.inp`) is presented in the Appendix as file `brat02.epx`. The mesh is visible in Figure 2 and now the grid is much more regular in the posterior region.



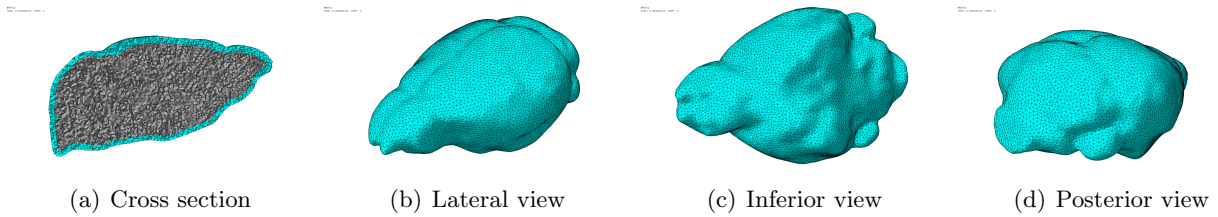


Figure 2: First volumetric mesh of the rat brain (from `brain_and_csf.inp`), EPX case `brat02`.

Table 2 lists the EPX input files used to check the rat brain meshes. No transient simulations were performed, only (interactive) visualizations of the mesh.

| Case                | Mesh coming from               | Description  |
|---------------------|--------------------------------|--|
| <code>brat01</code> | <code>brain_stl2inp.inp</code> | First surface mesh (brain only)  |
| <code>brat02</code> | <code>brain_and_csf.inp</code> | First volume mesh (brain and CSF), mesh data embedded in the EPX file        |
| <code>brat03</code> | <code>brain_and_csf.inp</code> | Same as <code>brat02</code> but mesh read from external file                 |
| <code>brainv</code> | <code>brain_and_csf.inp</code> | Same as <code>brat02</code> but mesh read from Cast3m <code>.MSH</code> file |
| <code>brat04</code> | <code>brain_and_csf.inp</code> | Same as <code>brainv</code> but skull surface model is added in Cast3m       |

Table 2: Summary of EPX input files for the visualization of rat head meshes.

### 2.1.2 Reading the mesh from a separate file with a fixed format

When the mesh contains *many* nodes and elements, including it directly in the EPX file can be unpractical. In such a case it is better to read the mesh from a separate file. This can be achieved by the so-called *fixed* (or COCO-like) format, in EPX's jargon (see [1] for full details).

An example is given in the following input file `brat03.epx`:

```
BRAT03
  CONV WIN
  LAGR TRID
  GEOM 9 '(3E13.5)' '(4I8)' POIN 115122 TETR 576079 TERM
  COMP GROU 9 'brain' LECT 1 PAS 1 414932 TERM
  'csf' LECT 414933 PAS 1 576079 TERM
  'head' LECT brain csf TERM
  'braix1' LECT brain TERM COND XB LT -6.111965
  'braix2' LECT brain DIFF braix1 TERM
  'csfx1' LECT csf TERM COND XB LT -6.111965
  'csfx2' LECT csf DIFF csfx1 TERM
  'headx1' LECT braix1 csfx1 TERM
  'headx2' LECT braix2 csfx2 TERM
  COUL GR50 LECT brain TERM
  TURQ LECT csf TERM
  MATE LINE RO 8000. YOUN 1.D11 NU 0.3
  LECT head TERM
  ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
  OPTI PAS AUTO NOTE LOG 1
  CALC TINI 0. TEND 1.DO NMAX 0
  FIN
```

Instead of `GEOM LIBR` we use the `GEOM n` form of the geometry directive. Here `n` designates the Fortran logical unit number from which the mesh data will be read. The value 9 used above is the default value for the EPX mesh file. The EPX launching procedure associates the mesh file with Fortran logical unit 9. For example, under Windows a file `fort.9` is created by the launching procedure, which copies into it the contents of the `.MSH` file associated with the input being executed.

The two strings `'(3E13.5)'` and `'(4I8)'` which follow are the Fortran formats that will be used to read the data: the first one refers to the nodal coordinates and the second one to the elements connectivity. Therefore in this case the format of the data is fixed and must be respected. Only one node, or one element, is listed in each line of this mesh file.

Beware of an important *limitation* with this type of input. Since the format for reading the element connectivity is fixed (4I8 in this example) the code will read the same number of nodes (4)

on each line. If the mesh has various types of elements, with different numbers of nodes (say, 4-node quadrangles and 3-node triangles), listing the nodes of each element in a separate line would not be appropriate. In that case one should pack the connectivity data so that each line contains the same amount of node indexes, in accordance with the given input format.

It is important to note that the fixed-format mesh file *must* start with (exactly) one line of text which is typically used to assign a title or a description to the following mesh data. This line may contain any characters: these are interpreted as the title of the mesh. Since the mesh file produced by `readst12` has an initial line containing `NPTL` and `NELEM`, it can be used directly as a (fixed-format, external) mesh file from EPX. To this end, just call (or rename) this file `<name>.msh`, where `<name>` stands for the base name of the EPX input file `<name>.epx`.

### 2.1.3 Passing through Cast3m

A third way of using the mesh file obtained from the `readst12` filter is to import it into the Cast3m mesh generator (see [2]). This may be useful for various purposes, e.g. in order to merge different meshes, to modify a mesh before reading it with EPX, to identify (and to name) parts of the mesh, etc.

In order to read the mesh data into Cast3m, there are several possibilities. Here we limit ourselves to describing the simplest one, which is using the `READ` operator of Cast3m. Note, however, that this operator is an add-on which is only available (together with the companion `WRITE` operator) in JRC's own local version of Cast3m. These operators are *not* available in the standard version distributed via the official Cast3m site [2].

Another possibility would be to modify the `readst12` filter so as to write directly a formatted SAUV file, which is the native format of Cast3m. This would not be too difficult at least for a single type and a single zone of elements as in this case.

The `READ` format of Cast3m of interest here is as follows.

1. `IDIM`, `NNOD`. A free-form line containing two integers, the first one is the space dimension and the second one is the total number of nodes in the mesh.
2. `x`, `y`, `z`. A line containing the coordinates of a node. The reading format is fixed and equal to `3(1X,E12.5)`. The `z` coordinate is read only if `IDIM` is equal to 3 (3-D geometry). This line is repeated as many times as necessary in order to define all the `NNOD` nodes.
3. `NBSOMM`, `NBELEM`, `NOMELM`. A free-form line containing two integers, the first one indicating the number of nodes per element, the second one indicating the total number of elements, followed by a string indicating the type of element used. Note that the element type *must* be given in *uppercase* characters. The element types available in Cast3m are `TRI3` for a 3-node triangle, `TET4` for a 4-node tetrahedron, etc. (see [2] for a complete list of element types).
4. `n_1`, `n_2`, ..., `n_m`, where `m` equals `NBSOMM`. A line containing the indexes of the nodes belonging to the first element. These numbers are read in *free* format. This line is repeated as many times as necessary in order to define all the `NBELEM` elements.

In order to obtain this file, the recommended procedure is as follows.

- Copy (or rename) the text file `brainvol.txt` produced by the `readst12` filter to `brainvol.read`. This is not really necessary since Cast3m's `READ` operator can read from any file (with any extension), but it helps keeping order in the process.
- Edit the file and replace the first line by a line containing only the space dimension (3 in this case) and the total number of nodes (115 122) in this case.
- Leave the following lines containing the nodal coordinates untouched, since they are already written in the correct format for the `READ` operator.

- Locate the end of the nodal coordinates and the beginning of the elements connectivity. Insert there a line containing the number of nodes per element (4 in this case), the total number of elements (576 079 in this case) and the element type (TET4 in this case).
- Leave all the remaining lines, containing the elements connectivity, untouched since they are already in an acceptable format.

The resulting `brainvol.read` file will be as follows:

```
3 115122
-0.11174E+02 -0.98448E+01 0.48253E+01
-0.11168E+02 -0.98924E+01 0.46769E+01
. . .
-0.84757E+01 -0.48191E+01 0.27586E+01
-0.69070E+01 -0.57082E+01 0.81264E+01
4 576079 TET4
 13852 13853 13855 13854
 13856 13857 13852 13858
. . .
101966 83697 105331 101967
101846 101848 101845 94534
```

The Cast3m (Gibiane) commands in order to read the mesh data are shown in the following Cast3m input file (`readbrainv.dgibi` in Appendix):

```
opti echo 1;
opti dime 3 elem cub8;
vol = 'READ' 'brainvol.read' 'MESH' 'ELEM';
opti sauv form 'brainv.msh';
opti trac psc ftra 'brainv_mesh.ps';
sauv form vol;
trac cach vol;
fin;
```

This reads in the mesh from the `brainvol.read` file, then saves it in the native Cast3m format on an (ASCII) file `brainv.msh`, in this simple case without manipulating it at all. Then EPX can read the `brainv.msh` file directly, like if it would have been created in Cast3m.

An example is given in the following EPX input file (`brainv.epx`):

```
BRAINV
ECHO
  CONV WIN
LAGR TRID
CAST vol
GEOM TETR vol TERM
COMP GROU 11 'brain' LECT 1 PAS 1 414932 TERM
          'csf' LECT 414933 PAS 1 576079 TERM
          'head' LECT brain csf TERM
          'braix1' LECT brain TERM COND XB LT -6.111965
          'braix2' LECT brain DIFF braix1 TERM
          'csfx1' LECT csf TERM COND XB LT -6.111965
          'csfx2' LECT csf DIFF csfx1 TERM
          'headx1' LECT braix1 csfx1 TERM
          'headx2' LECT braix2 csfx2 TERM
          'emin' LECT 3501 TERM
          'emax' LECT 288434 TERM
  COUL GR50 LECT brain TERM
  TURQ LECT csf TERM
  ROUG LECT emin emax TERM
MATE LINE RO 8000. YOUN 1.D11 NU 0.3
  LECT head TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN
```

The `CAST vol` directive reads the mesh object named `vol` from the Cast3m mesh file, by default named `brainv.msh` in this case. By visualizing the mesh with EPX, the images shown in Figure 3 are obtained.

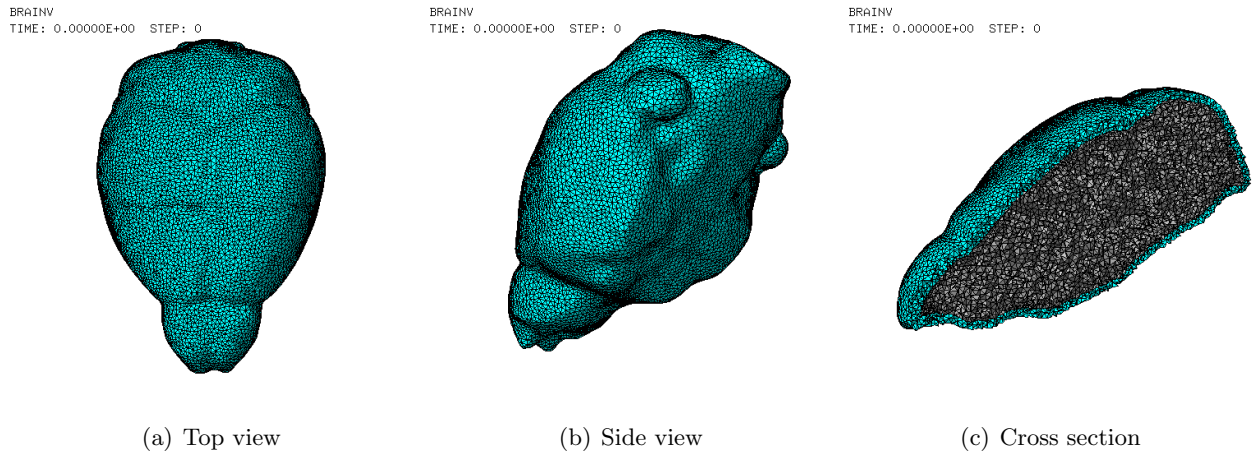


Figure 3: Mesh in case `brainv`.

### 2.1.4 Manipulating the mesh by Cast3m

As an example of mesh manipulation by Cast3m, we add a skull model to the brain and CSF (volumetric) model considered in the previous examples. Since the skull is very stiff compared with the brain and CSF, it can sometimes be represented as a rigid body. We will therefore generate a (rigid) surface made of triangles to represent the skull.

Since we want to treat FSI between the skull and the CSF (in addition to FSI between the CSF and the brain), the simplest possibility is to use the classical FSA algorithm of EPX [1]. Therefore, we need a *conforming* mesh. In other words, the skull is just a duplication (but with *distinct* nodes) of the external surface of the CSF.

To generate this surface with Cast3m we may proceed as follows (see file `genmesh.dgibi`):

```
opti echo 1;
opti dime 3 elem tet4;
opti trac psc ftra 'genmesh.ps';
opti rest form 'brainv.msh';
rest form;
list (nbel vol);
list (nbno vol);
```

We start by reading back the volumetric mesh for the brain and CSF from the `brainv.msh` file generated previously (alternatively, one might read this via the `READ` operator from the `brainv.txt` mesh file). This is done by the `REST` operator and makes available the complete volumetric mesh as the object named `vol`.

```
lbra = lect 1 pas 1 414932;
lcsf = lect 414933 pas 1 576079;
brain = vol elem lbra;
csf = vol elem lcsf;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
```

Next, we want to identify (name) the two parts of this object, that is the brain and the CSF. We know the element numbers corresponding to each part, as already commented previously. We use the `LECT` operator to define the two lists of elements and then the `ELEM` operator to extract the two (sub-)objects `brain` and `csf` from the `vol` mesh.

```
p0 = 0 0 0;
vol12 = vol plus p0;
```

Next, we duplicate the complete volume into a new object `vol12` by adding a zero displacement to it (by the `PLUS` operator). This is in order not to modify the original object by the operations we are going to execute next.

```

tol = 1.E-7;
elim tol vol2;
list (nbel vol2);
list (nbno vol2);

```

We use the ELIM operator with a small tolerance in order to *merge* together the (duplicated) brain and skull. Recall in fact that the (external) surface of the brain and the internal surface of the skull are *distinct* in the original mesh. In this way, we obtain a single volumetric object whose surface coincides with the *external* surface of the CSF (in practice, the internal surface of the CSF is eliminated in the duplicated mesh since it becomes an internal surface to the mesh).

```

skull = enve vol2;
list (nbel skull);
list (nbno skull);
oubl vol2;

```

Next, the ENVE operator extracts the external surface of `vol2` and names it `skull`. We can now *forget* (OUBL) the duplicated `vol2` object, which is no longer needed. Note that failing to do so would leave the nodes of this object in the final mesh file to be passed to EPX, which would complain about the presence of (a lot of) extra nodes in the mesh, not belonging to any element. This is not an error, but it would waste quite some memory in the following transient numerical simulations with EPX.

```

fsan = enve csf;
head = brain et csf et skull et fsan;
opti sauv form 'head.msh';
tass head noop;
sauv form head;
trac cach head;
fin;

```

We give a name `fsan` to the surfaces (both internal and external) of the CSF, since their nodes will be subjected to the FSA condition in EPX. Finally, we pack all our objects by the ET (union) operator into a global object called `head` and we save it in a file `head.msh`, ready to be read in by EPX. This file is then used by EPX as a complete mesh file, see the input `brat04.epx` in Appendix. The resulting model is shown in Figure 4.

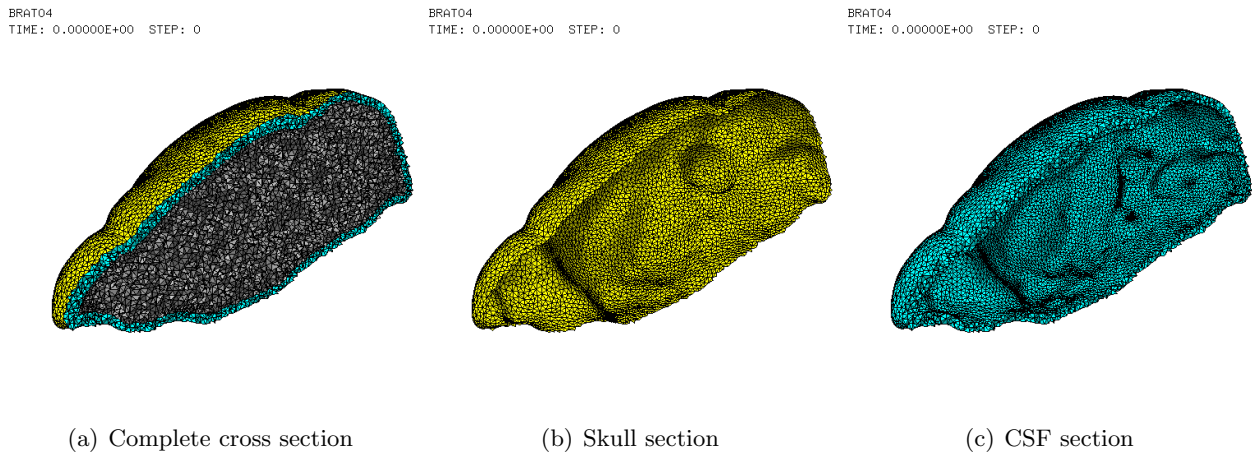


Figure 4: Mesh in case `brat04`.

## 2.2 Rat head meshes

A view of the “real” rat’s brain anatomy is given in Figure 5, from [5]. Here we summarize the various rat head meshes used in the present study:

1. A first surface mesh of the rat’s brain (file `brain_stl2inp.inp`). The mesh is shown in Figure 1. This mesh is too coarse and irregular in the posterior region to be used for numerical simulations. However, it was useful in order to set up the various mesh conversion utilities described above.

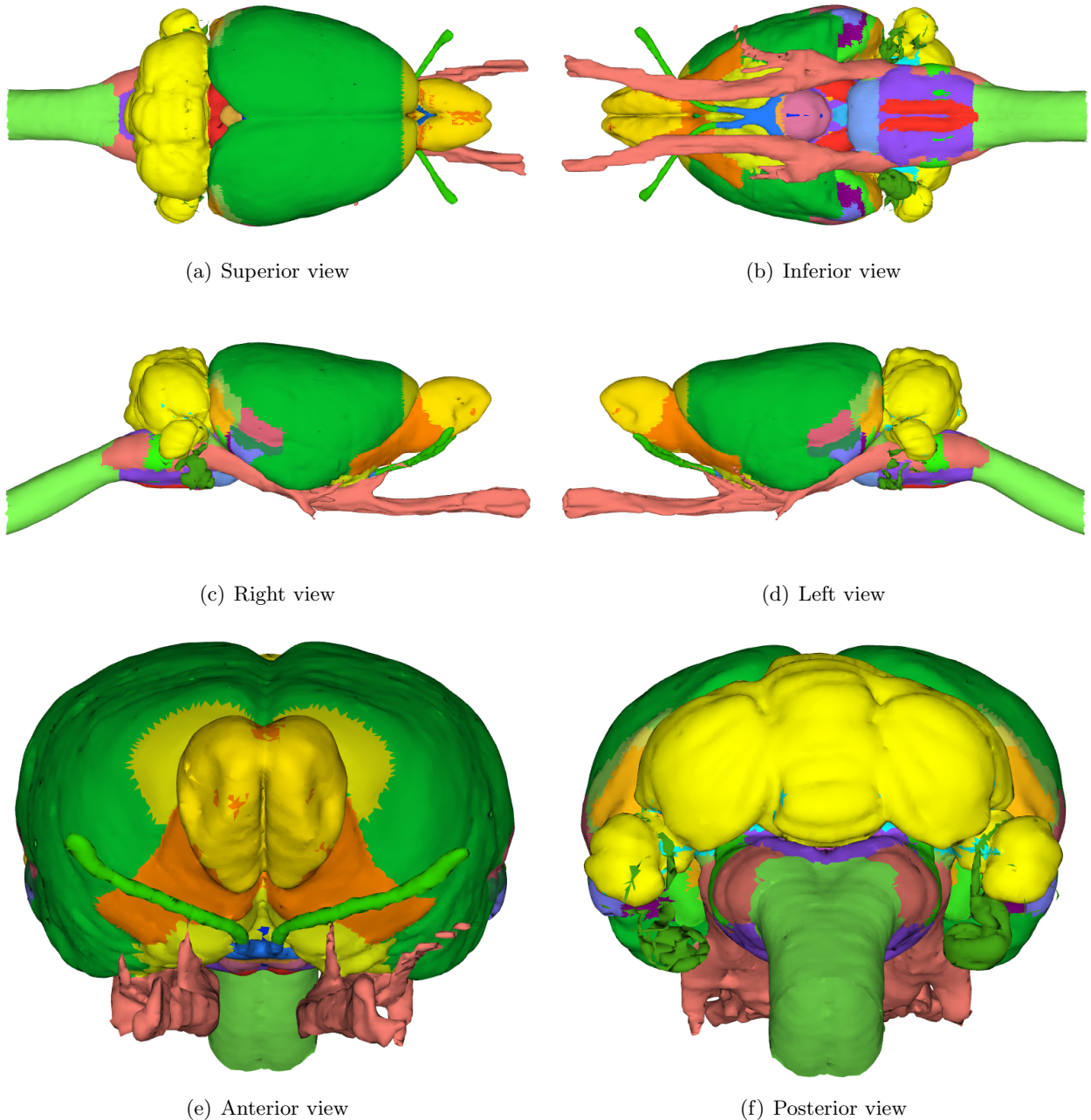


Figure 5: Rat brain anatomy (from [5]).

2. A *first volume mesh* of the rat's brain and CSF (file `brain_and_csf.inp`, see Figures 2 and 3. The mesh was passed through Cast3m and a surface mesh for the skull was added to it, see Figure 4(b). Then it was used for a few preliminary numerical simulations (see cases `brat04` and `brat05` in Section 3.1) which, however, were immediately stopped because they revealed that this mesh contains very badly shaped elements and so is inappropriate for simulations.
3. A *second volume mesh* of the rat's brain and CSF (files `brain.inp` and `csf.inp`. This mesh was used for the actual 3D numerical simulations, described in Section 3.4.

The second volumetric mesh of the rat's brain was obtained in order to have a more uniform grid in the interior of the brain and CSF, and thus facilitate the numerical simulations. Four data files were used in order to assemble this mesh, two for the brain (listing the nodes and the elements, respectively) and two similar ones for the CSF. They are summarized in Table 3.

These raw data were imported in Cast3m by the technique discussed in Section 2.1.3 (`READ` operator), see input files `readbrain.dgibi` and `readcsf.dgibi` in the Appendix. This generated two

| File                     | N. of lines (nodes or elements) |
|--------------------------|---------------------------------|
| <code>nodes_brain</code> | 57 938                          |
| <code>elem_brain</code>  | 312 755                         |
| <code>nodes_csf</code>   | 35 582                          |
| <code>elem_csf</code>    | 145 746                         |

Table 3: Raw mesh data for the second rat head volumetric mesh.

Cast3m mesh files named `brain.msh` and `csf.msh`, respectively. Then, a Cast3m command file `genmesh.dgibi` similar (but not identical) to the one described in Section 2.1.4 (see file renamed `genmesh_new.dgibi` in Appendix) was used in order to add the skull surface mesh and to save the complete (brain, CSF and skull) mesh in a file `head.msh`. Again, this name is the same as the one already used in Section 2.1.4, but here the new (second) rat head volumetric mesh is used and not the first one.

The complete rat head volumetric mesh (second moel) consists of the nodes and elements summarized in Table 4.

| Component | Nodes   | Elements | Element type |
|-----------|---------|----------|--------------|
| Brain     | 57 938  | 312 755  | TET4         |
| CSF       | 35 582  | 145 746  | TET4         |
| Skull     | 10 865  | 21 726   | TRI3         |
| Total     | 104 385 | 480 227  |              |

Table 4: Second volumetric mesh for the rat head (file `head.msh`).

## 2.3 Human head mesh

A mesh of the human head was provided in four STL2-format (`.INP`) input files, as listed in Table 5.

| File                        | N. of lines | Contents   |
|-----------------------------|-------------|--|
| <code>nodes.inp</code>      | 905 237     | Coordinates of the 905 237 nodes                   |
| <code>elem_brain.inp</code> | 5 300 984   | Connectivity of the 5 300 984 tetrahedral elements |
| <code>groups_geo.inp</code> | 331 329     | Named element groups                               |
| <code>groups_bc.inp</code>  | 15 613      | Named node groups                                  |

Table 5: Mesh data for the human head.

The list of the named element groups is shown in Table 6 and the list of the named node groups is given in Table 7. The original names have been shortened so that the univocally fit into a Fortran `CHARACTER*8` variable.

| Group name           | N. of elements | N. of nodes |
|----------------------|----------------|-------------|
| <code>fluid</code>   | 2 062 404      | 433 682     |
| <code>grey_l</code>  | 833 854        | 194 833     |
| <code>white_l</code> | 782 160        | 157 833     |
| <code>grey_r</code>  | 844 628        | 196 335     |
| <code>white_r</code> | 777 938        | 156 936     |

Table 6: Named element groups in the human head mesh.

| Group name            | Contents                              |
|-----------------------|---------------------------------------|
| <code>nwhite_r</code> | Nodes of the white matter, right part |
| <code>nfluid_f</code> | Outer nodes of the CSF fluid          |
| <code>nfluid_l</code> | Nodes of the CSF, left part           |
| <code>nfluid_2</code> | Unknown                               |
| <code>nfluid_d</code> | Unknown                               |

Table 7: Named node groups in the human head mesh.



The four `.inp` files were combined by hand into a single file `human.inp` by adding the necessary lines introducing each type of data. This was filtered by the `readst12` utility, producing file `human.txt` which contains the whole human head mesh (brain and CSF) in a format that can be read in by EPX. As part of the conversion process, the nodes and elements were renumbered consecutively starting from 1, since the original numberings in the `.inp` files were non-consecutive.

In order to check the mesh data, an EPX input file `huma01.epx` was prepared (see the Appendix). The file `human.txt` was copied to a file `huma01.msh`. The element and node group data were moved from this file to an appropriate location within file `huma01.epx`.

By running EPX on file `huma01.epx` the following views of the mesh were obtained. Figure 6 shows the complete mesh. Of course, only the CSF (colored in cyan) is visible. Figure 7 shows various views of the grey and white brain matter, colored in a darker and lighter shade of grey, respectively. Finally, Figures 8 and 9 show some further details of the white matter mesh.

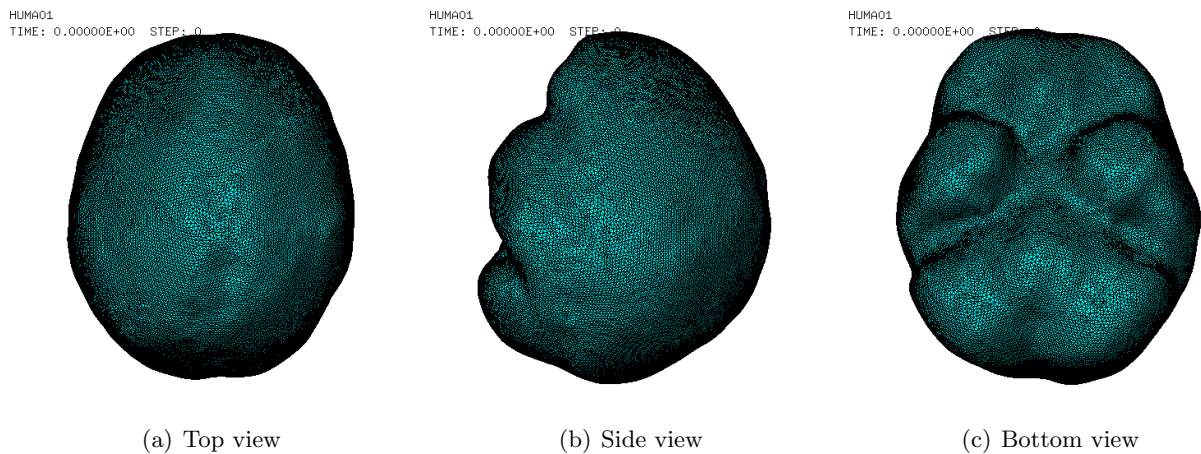


Figure 6: Human head mesh (complete).

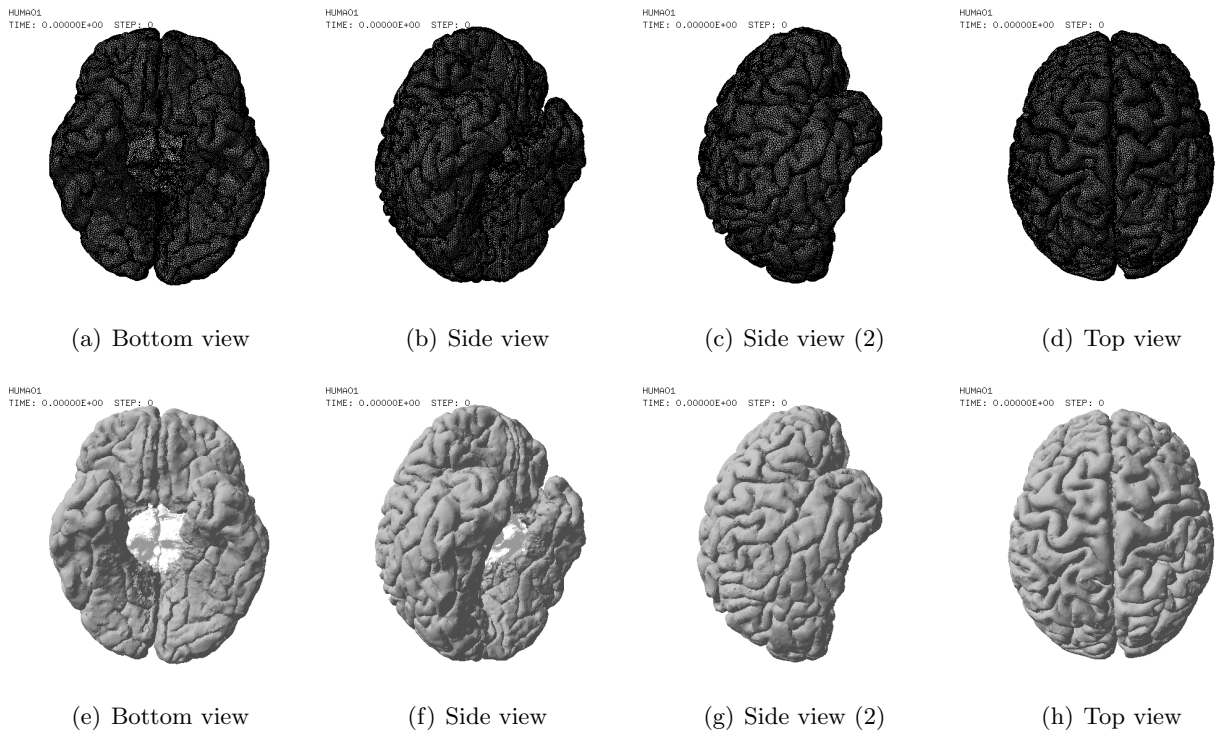


Figure 7: Human brain mesh: grey and white matter (with and without element outlines).



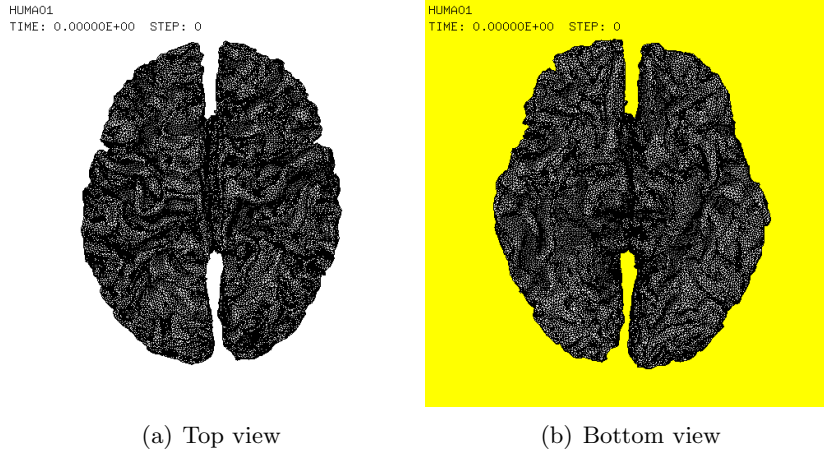


Figure 8: Human brain mesh: white matter (with element outlines).

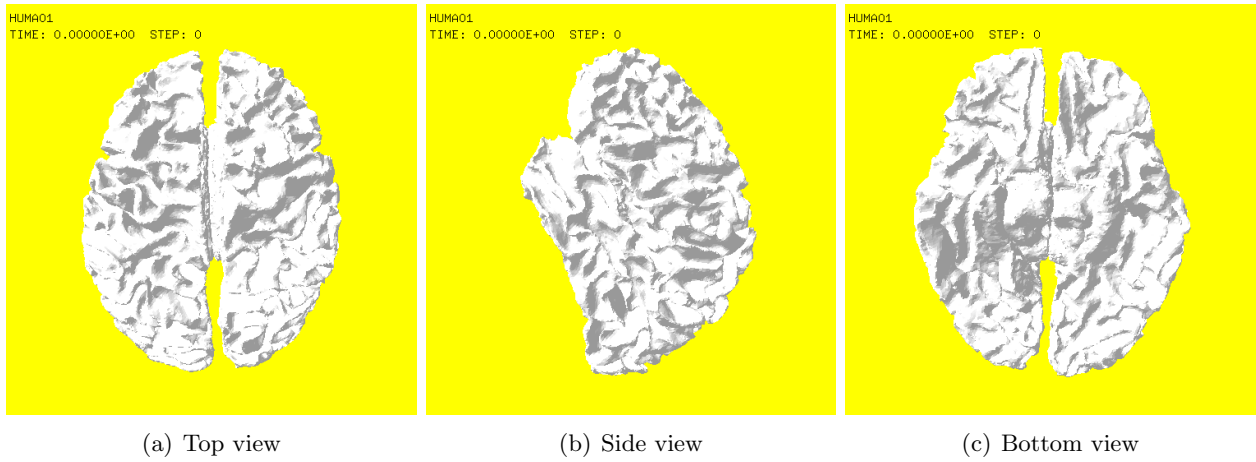


Figure 9: Human brain mesh: white matter (without element outlines).

The mesh dimensions in the original `.inp` files are in mm. They are converted into standard units (m) when reading the mesh with EPX by means of the command `GEOM SCAL FACT 0.001`.

Some measurements of the main geometrical characteristics of the human head mesh were performed interactively by using the input file `huma01.epx`. They are summarized in Table 8 (after the above mentioned units conversion). The overall extensions of the human head model are:  $L_x = 140.048$  mm,  $L_y = 167.914$  mm and  $L_z = 134.711$  mm.

| Component | $d_{\min}$ | $d_{\max}$ | Ratio |
|-----------|------------|------------|-------|
| Brain     | 54.6       | 4168.4     | 76    |
| CSF       | 103.1      | 4168.4     | 40    |

Table 8: Minimum and maximum element sizes ( $\mu\text{m}$ ) of the human head components in case `huma01`.

The total number of elements and nodes in the various components are summarized in Table 9. The `brain` object is the union of the grey and white matter, i.e. the union of the `grey_l`, `grey_r`, `white_l` and `white_r` groups defined above. The `head` object is the complete model, i.e. the union of the `fluid` (CSF) and `brain` objects.

Note that the total number of nodes in the `head` is less than the sum of the nodes in the `fluid` and in the `brain`. The difference amounts to 130 060 nodes. This means that in this mesh the brain and the CSF are *merged* together, i.e. single instead of double nodes are provided along the interface between the brain and the CSF. This might need to be corrected if a full FSI simulation has to be performed.

An estimate of the stability step in this model was obtained by assigning material properties in the

| Component | N. of elements | N. of nodes |
|-----------|----------------|-------------|
| fluid     | 2 062 404      | 433 682     |
| brain     | 3 238 580      | 601 615     |
| head      | 5 300 984      | 905 237     |

Table 9: Summary of nodes and elements in the human head mesh.

file `huma01.epx`. By using the Ogden set 1 material constants (9) to be described below in Section 3.2.2 for the brain matter (same material for the grey and white matter as a first approximation) and a fluid water-like material for the CSF, the following input data are obtained:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      C01 -10.8   ! This is alpha_1
      C02  19.7   ! This is alpha_2
      C05 -0.29   ! This is mu_1
      C06  9.6E-3 ! This is mu_2
      LECT grey_l white_l grey_r white_r TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT fluid TERM
```

This results in the following stability steps, with a safety factor `CSTA` equal to 0.5. The most critical element is element 2357453 located in the CSF with a value  $\Delta t_{\min} = 2.60 \times 10^{-8}$ , while the least critical element is element 1180305, located in the brain (grey matter, right region), and having  $\Delta t_{\max} = 2.69 \times 10^{-2}$ . Therefore, a ratio of over one million exists between the maximum and minimum stability steps, with the chosen materials. The critical element is a sliver, i.e. it has a very large aspect ratio.

### 3 Numerical simulations of the rat head

Numerical simulations of the effects of deceleration were started on the rat head model, which is by far simpler than the human head model.

#### 3.1 Preliminary 3D simulations with the first volumetric mesh

The first attempted simulation was case `BRAT04` which has already been described in Section 2.1.4 (see Figure 4) as concerns the geometrical data.

It should be noted that, by inspecting the mesh data, it was found that the coordinates of the mesh are expressed in non-standard units, i.e. in mm instead of m. Since it is always preferable to use standard units, a scaling command is applied in EPX in order to convert all coordinates to metres:

```
GEOM SCAL FACT 0.001
      TETR brain csf T3GS skull TERM
```

The above `SCAL FACT` command multiplies all the nodal coordinates read from the mesh file by the chosen factor (0.001), thus converting them from mm to m. All other dimensioned quantities in the EPX input file will then be expressed in standard units (in particular, metres will be used for the lengths).

The model is a full 3D model containing all the components of the rat head: the brain, the CSF and the skull. The latter is modelled by shells and it is assumed to behave as a rigid body, with an imposed velocity varying linearly in time (constant deceleration) from an initial value (uniform in the whole head), until the skull is at rest, and then it is kept indefinitely at rest.

Recall that the three regions of the head are modelled as distinct (but conforming) meshes. The scope is to perform a Fluid-Structure Interaction (FSI) analysis by using the classical FSA algorithm implemented in EPX (see [1] for details). The CSF is tentatively represented by a compressible fluid which can freely slide with respect to the brain and the skull.

### 3.1.1 Materials

The brain is tentatively modelled as a hyperelastic material. Like in reference [3], the material model tentatively used (HYPE TYPE 4) is a new Ogden material model (replacing the former HYPE TYPE 3 Ogden material implementation) currently in development at JRC. The implementation at the moment is quite heavy as concerns CPU time but this might be optimized in the future if this model has to be used in many parametric calculations.

For the Ogden material (new formulation) the constitutive law is expressed by the following equation:

$$W = \sum_{p=1}^N \frac{\mu_p}{\alpha_p} (\lambda_1^{*\alpha_p} + \lambda_2^{*\alpha_p} + \lambda_3^{*\alpha_p} - 3) + K(J - 1 - \ln J) \quad (1)$$

with  $N$  the number of terms retained,  $\lambda_i^* = \lambda_i J^{-\frac{1}{3}}$ ,  $\lambda_i$ ,  $i = 1, \dots, 3$  the principal strains (or principal stretch ratios  $\lambda_i = L_i/L_{0i}$ ) and  $J = V/V_0 = \lambda_1 \lambda_2 \lambda_3$  is the ratio between the current and the initial volume, which should be very close to 1.0 for a nearly incompressible material. Note that it is also  $J = \det \mathbf{F}$  where  $\mathbf{F}$  is the deformation gradient. Here  $W$  is the strain energy density,  $K$  is the bulk modulus,  $\mu_p$  and  $\alpha_p$  are the material parameters (to be determined).

The values of the coefficients assumed in this first model are the same as the ‘‘corrected’’ values adopted in reference [3], namely:

$$\begin{aligned} N = 2 \quad \alpha_1 = -14.058 \quad \alpha_2 = 0.38793 \quad \mu_1 = 8.8365 \times 10^5 \quad \mu_2 = 5.0888 \times 10^8 \\ \rho = 1040 \text{ kg/m}^3 \quad K = 128 \times 10^6 \text{ Pa} \end{aligned} \quad (2)$$

In the above equations,  $\rho$  is the density and  $K$  is the bulk modulus assumed for the brain material.

The CSF is tentatively modelled by a (water-like) fluid with the same characteristics assumed in reference [3], that is a FLUI material with density  $\rho = 1040 \text{ kg/m}^3$ , sound speed  $c = 1451 \text{ m/s}$  and dynamic viscosity  $\mu = 1.0 \times 10^{-3} \text{ Pa}\cdot\text{s}$ .

The material assumed for the skull is a linear elastic material (LINE) of density  $\rho = 1800$ , Young’s modulus  $E = 1.5 \times 10^9$  and Poisson’s coefficient  $\nu = 0.21$ , but these values are irrelevant since the skull is treated as rigid (all its displacements are imposed).

The materials definition becomes:

```
MATE HYPE TYPE 4 RO 1040
      BULK 128E+06
      CO1 -0.14058E+02
      CO2 0.38793E+00
      CO5 0.88365E+06
      CO6 0.50888E+09
      LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
      LECT skull TERM
```

### 3.1.2 Mesh quality measurements

By reading the `brat04.epx` input file with EPX and performing some measurements in order to estimate the cost of the numerical simulation, it was found that the stability step was extremely low. Table 10 lists the minimum and maximum intra-nodal distances (in  $\mu\text{m}$ ) in the elements of the various head components. Both in the brain and in the skull, there is a very large ratio between the maximum and minimum distances, thus indicating that some of the elements have an extremely flattened aspect. Tetrahedra of this type are sometimes called *slivers* in the literature. The situation is better in the skull, but this is irrelevant since the skull is treated as rigid so its elements do not contribute to global stability of the model.

Since the stability step is dictated by the minimum intra-nodal distance, the presence of slivers risks to heavily penalize the computation. As a matter of fact, the minimum stability step occurs in the CSF and is as low as  $\Delta t_{\min} = 3.66 \times 10^{-10}$ . In these conditions, 2.7 million time steps would

| Component | $d_{\min}$ | $d_{\max}$ | Ratio |
|-----------|------------|------------|-------|
| Brain     | 1.5        | 454.7      | 303   |
| CSF       | 1.5        | 393.0      | 262   |
| Skull     | 60.8       | 336.8      | 6     |

Table 10: Minimum and maximum element sizes ( $\mu\text{m}$ ) of the head components in case `brat04`.

be required to reach just 1 ms of physical time (which is a small fraction of the intended final time), making this computation totally unrealistic in practice.

This was the main reason for preparing a better mesh of the rat head (second volumetric mesh), as described in Section 2.2.

### 3.1.3 Eroding the bad elements

A subsequent EPX calculation `brat05` was used in order to explore the possibility of using *element erosion* in order to get rid of the most critical elements in the mesh. The optional `TFAI` keyword in the `CALC` directive of EPX can be used in order to erode (i.e. to exclude from the calculation) any elements whose stability step drops below a user-defined value. Of course, this applies only to elements using an *erodible* material, and provided erosion has been activated at the global level in the calculation by means of the `EROS` directive. The `HYPE` material is erodible, but the `FLUI` material is not.

By activating the `STEL` option, the stability steps of all elements are printed out on the listing at every chosen printout time station (including at the initial time). Additionally, an histogram of the distribution of stability steps in the mesh is printed.

The minimum time step (with a safety factor `CSTA` equal to 0.5 over the estimated value) for this test case is as low as  $3.66 \times 10^{-10}$  s and occurs in element 417537. Unfortunately, this element is part of the CSF (fluid material) and cannot be eroded. The largest time step is  $3.14 \times 10^{-7}$  s and occurs in element 248730 which is part of the brain.

From the stability histogram one sees that there are 15 elements in the lowest 1/100 of the stability range, which goes from  $3.66 \times 10^{-10}$  to  $3.50 \times 10^{-9}$ . However, only 5 of these are in the brain, while the rest (including the most critical one as already mentioned) are in the CSF fluid.

By activating erosion with `TFAI 3.51E-9` the five most critical brain elements are eroded, but the time increment remains the same since the most critical element is in the CSF and cannot be eroded. The erosion technique therefore is ineffective in this particular case.

By inspecting the mesh, one sees that the eroded elements are in a small region in the lower part of the brain, near the interface with the CSF, see Figure 10.

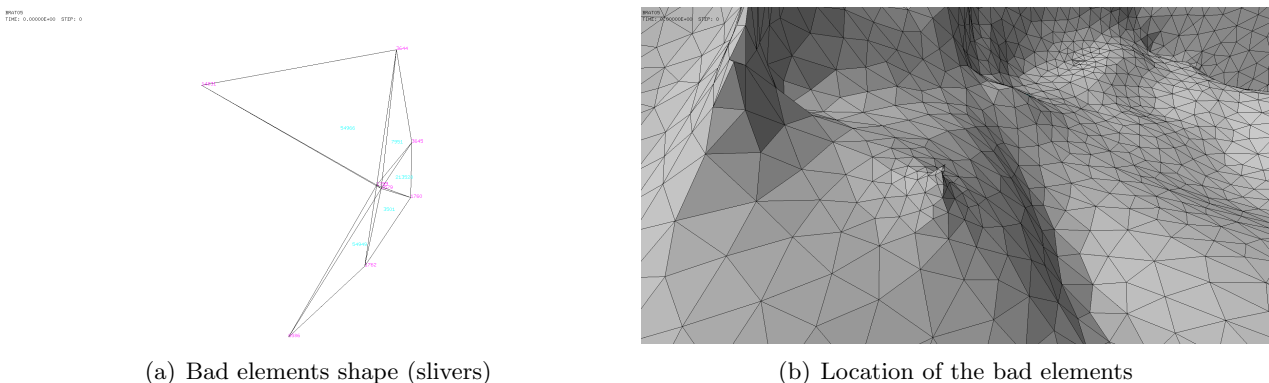


Figure 10: Most critical elements in the brain in case `brat05`.

## 3.2 New material data

The material models used in the simulation were also inspected. It turned out that several authors in the literature do not simulate the CSF as a fluid, but as a hyperelastic material having different properties from the brain matter (stiffer behaviour).

The following alternative material models were considered (both for the brain and for the CSF).

### 3.2.1 Mooney-Rivlin material

An *incompressible* Mooney-Rivlin hyperelastic material is described by:

$$W = C_1(\bar{I}_1 - 3) + C_2(\bar{I}_2 - 3) \quad (3)$$

where  $W$  is the strain energy density function,  $C_1$  and  $C_2$  are empirically determined material constants and:

$$\bar{I}_1 = J^{-2/3} I_1 \quad I_1 = \lambda_1^2 + \lambda_2^2 + \lambda_3^2 \quad (4)$$

$$\bar{I}_2 = J^{-4/3} I_2 \quad I_2 = \lambda_1^2 \lambda_2^2 + \lambda_2^2 \lambda_3^2 + \lambda_3^2 \lambda_1^2 \quad (5)$$

Here  $I_1$  and  $I_2$  are the first and second invariants of the unimodular component of the left Cauchy-Green deformation tensor and:

$$J = \det \mathbf{F} = \lambda_1 \lambda_2 \lambda_3 \quad (6)$$

with  $\mathbf{F}$  the deformation gradient. For an incompressible material  $J = 1$ .

For a *compressible* Mooney-Rivlin material eq. (3) becomes:

$$W = C_1(\bar{I}_1 - 3) + C_2(\bar{I}_2 - 3) + K(\ln I_3)^2 \quad (7)$$

with  $K$  the bulk modulus and  $I_3$  is the third invariant.

The formulation of a Mooney-Rivlin material implemented in EPX uses a more general expression of the strain energy (potential) function than (7), which includes also *higher-order* (quadratic, cubic etc.) terms in the invariants via additional material coefficients  $C_3, \dots, C_{14}$ .

The following set of parameters is proposed:

$$\begin{aligned} C_1 = 0.185 \text{ kPa} \quad C_2 = 0.0463 \text{ kPa} \quad C_3, \dots, C_{14} = 0 \\ \rho = 1040 \text{ kg/m}^3 \quad K = 128 \times 10^6 \text{ Pa} \end{aligned} \quad (8)$$

The same value for the bulk modulus  $K$  as for the Ogden material used in the preliminary simulations of reference [3] (see also eqs. (2)) is tentatively assumed. The *units should be checked*, since the values of  $C_1$  and  $C_2$  are said to be valid for lengths expressed in mm (not in m).

### 3.2.2 Ogden material set 1

Two sets of material parameters for a Ogden material characterized by eq. (1), both quite different from those that had been used in the preliminary calculations of [3] with the spherical geometry, are also proposed.

The first set, referred to as material *Ogden set 1* in the following, is:

$$\begin{aligned} N = 2 \quad \alpha_1 = -10.8 \quad \alpha_2 = 19.7 \quad \mu_1 = -0.29 \quad \mu_2 = 9.6 \times 10^{-3} \\ \rho = 1040 \text{ kg/m}^3 \quad K = 128 \times 10^6 \text{ Pa} \end{aligned} \quad (9)$$

Also here the *units should be checked*, like in the case of the Mooney-Rivlin material. In fact, by looking at eq. (1) it seems that the  $\alpha$  and  $\mu$  parameters are *dimensional*. Since  $J$  is non-dimensional, the  $W$  (strain energy density function?) in (1) has the same units as the bulk modulus  $K$ , that is a stress, expressed in Pa in the SI units system. Then, since the  $\lambda_i$  are also non-dimensional, it turns out that the ratio  $\mu_p/\alpha_p$  should also have the dimensions of a stress (Pa).

### 3.2.3 Ogden material set 2

The second set, referred to as material *Ogden set 2* in the following, is:

$$\begin{aligned} N = 2 \quad \alpha_1 = 23.5 \quad \alpha_2 = -4.0 \quad \mu_1 = 8.32 \times 10^{-4} \quad \mu_2 = -0.41 \\ \rho = 1040 \text{ kg/m}^3 \quad K = 128 \times 10^6 \text{ Pa} \end{aligned} \quad (10)$$

## 3.3 Testing the new material parameters on simplified geometries

Some tests of the new material parameters were performed on relatively simple geometries before applying them to full 3D rat head calculations.

### 3.3.1 Mooney-Rivlin test head46

The first test was performed in order to check the Mooney-Rivlin material of Section 3.2.1 in the spherical head impact test described in reference [3]. This is a repetition of case **head46** of [3] where the Mooney-Rivlin material parameters (8) are used instead of the Ogden parameters (2) in the brain material. The material input data become, therefore:

```
MATE HYPE TYPE 1 RO 1040      ! Mooney-Rivlin
      BULK 128E+06 ! This is in Pa
      CO1 185.0   ! These are in Pa
      CO2 46.3
      LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
      LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM
```

Note that the  $C_1$  and  $C_2$  coefficients of the Mooney-Rivlin material have been tentatively converted to SI (Pa) in order to be consistent with the SI units system that is adopted here.

This calculation was intentionally stopped after 1918 time steps, having reached  $t = 0.764$  ms in 5056 s of CPU, because the results were not convincing. The calculation times were comparable to (slightly faster than) those of the previous case with the Ogden material. However, it was noticed that starting at the initial time, i.e. even *before* head impact occurs, a strange parasitic spherical stress wave is developed starting at the external surface of the brain material, as illustrated in Figure 11 which compares results from the two solutions (**head36** in the top row and **head46** in the bottom row).

### 3.3.2 Mooney-Rivlin consistency tests head56, head66 and head55

In order to investigate the origin of the spurious spherical wave observed in the Mooney-Rivlin brain material, test **head56** was conducted as a repetition of solution **head46** but with the following differences: first, no initial velocity is set in the head; second, no pinballs are used. The pinballs become useless since in the absence of any initial velocity the head should remain steady and stress-free forever (no impact).

At step 0 of the calculation (initial time  $t = 0$ ), some spurious internal forces are observed on the whole surface of the brain (Mooney-Rivlin material), see Figure 12(a). This is strange because in the conditions assumed here the brain material should be stress-free. The CSF liquid does have an initial (*absolute*) pressure of 1 bar, but the *reference* pressure is also set to 1 bar in the liquid, so that its *relative* pressure is zero and the liquid exerts no force on the brain (nor on the skull). The spurious force on the external brain nodes is transferred to the CSF due to the FSI algorithm (FSA model), as can be seen in Figure 12(b).

For confirmation, the same test is repeated but with the Ogden material (2) of reference [3] instead of the Mooney-Rivlin material, see case **head66**. In this case, the internal (and link) forces are rigorously zero, as it should be expected, see Figures 12(c) and 12(d). This proves that the behaviour of the Mooney-Rivlin material is faulty, and will have to be corrected.

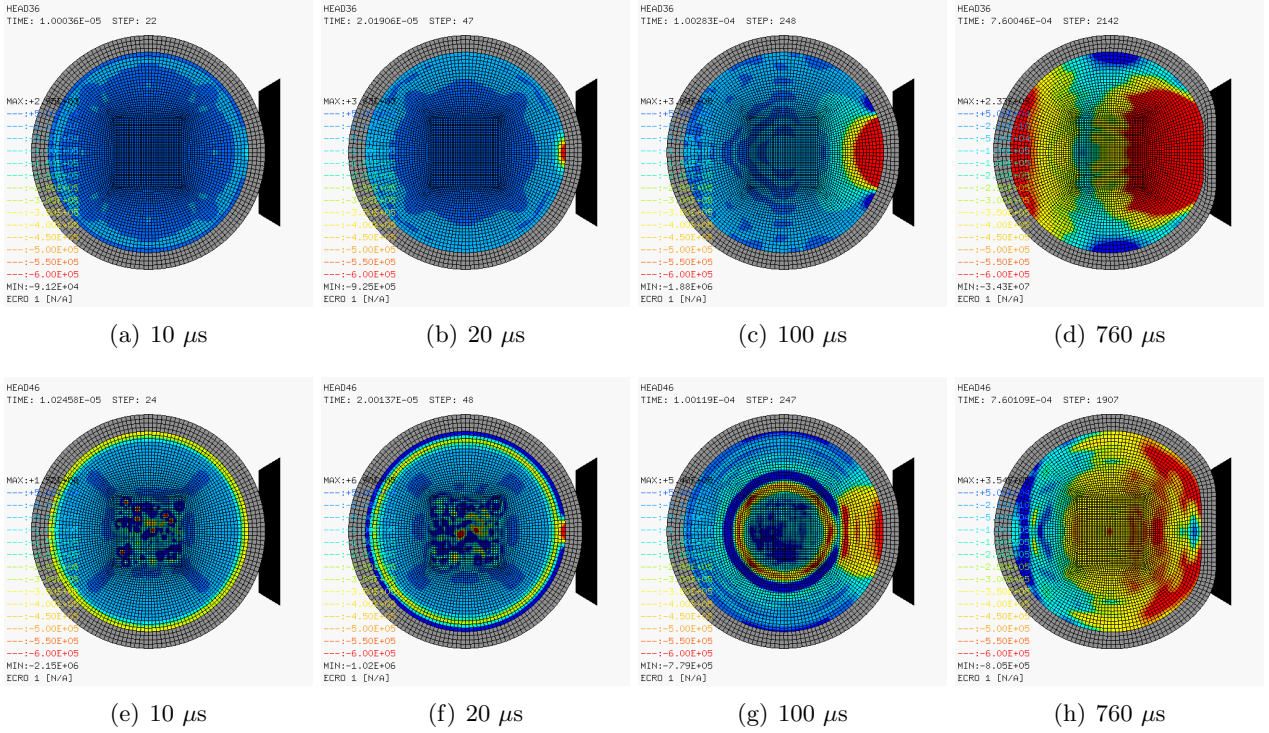


Figure 11: Spherical head impact test: comparison of Ogden (top row) and Mooney-Rivlin (bottom row) materials.

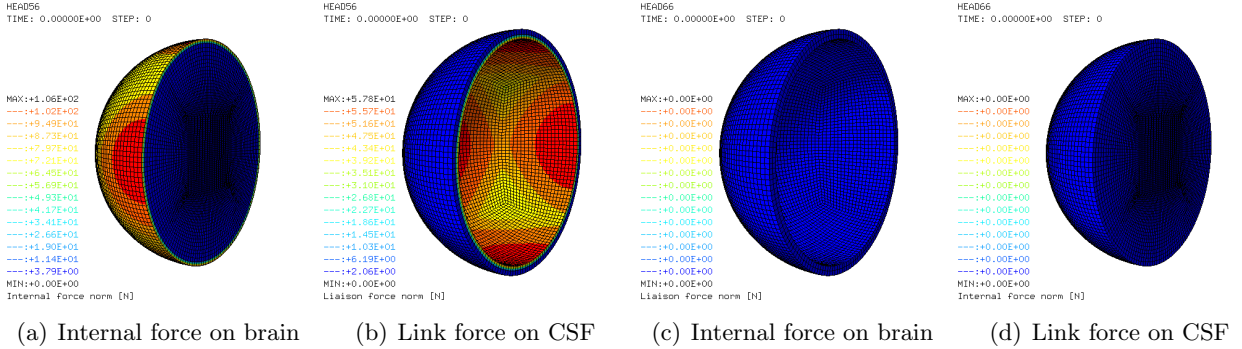


Figure 12: Forces in the brain and in the CSF in cases head56 (left) and head66 (right).

In order to highlight even more precisely the fault in the Mooney-Rivlin material behaviour (in view of its correction), a much simpler test is performed, case `head55`. A cube of Mooney-Rivlin material is discretized by  $3 \times 3 \times 3 = 27$  regular hexahedra and is subjected to no external forces and no initial velocity. The stress should clearly be zero in the material, and also the internal forces should vanish.

Two cubes are modelled in the test, with the same Mooney-Rivlin material parameters but with different values of the bulk modulus  $K$ . The first cube has  $K = 128 \times 10^6$  Pa like in the head calculations performed so far, while the second cube has  $K = 0$ . Figure 13 shows the result of the calculation. At the initial time spurious (very large) forces are observed in the material with  $K > 0$  (left cube) while they are comparatively zero in the material with  $K = 0$  (right cube). This will hopefully provide a useful hint on how to correct the behaviour of the Mooney-Rivlin material in EPX.

Another fault in the Mooney-Rivlin material that has emerged in this simple test is the fact that the stability in an element using this material is apparently *not* evaluated, probably due to the fact that an expression giving the sound speed in the material is not implemented. Consequently, the



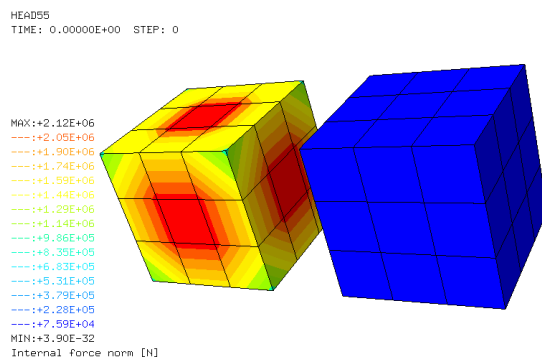


Figure 13: Forces in nominally stress-free cubes made of Mooney-Rivlin material.

stability limit is set to a huge value in these elements ( $1.0 \times 10^{12}$ ). In the presence of other materials, like in the head impact tests presented above, this fact is not noticed because the other materials do have a stability limit. However, in a test like the present one with only the Mooney-Rivlin material, the user is obliged to explicitly choose a step for marching in time (e.g. a constant value) in order to run a simulation.

This fault will also have to be corrected in the Mooney-Rivlin material implementation if this material has to be used in realistic numerical simulations.

### 3.3.3 Ogden set 1 test head47

This calculation is a repetition of case `head46` presented above but by using the Ogden set 1 (9) material instead of the Mooney-Rivlin material in the brain. The materials input data become:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new formulation)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8   ! This is alpha_1
      CO2  19.7   ! This is alpha_2
      CO5 -0.29   ! This is mu_1
      CO6  9.6E-3 ! This is mu_2
      LECT brain TERM
      FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
      LINE RO 1800 YOUN 1.500E9 NU 0.21
      LECT skull TERM
      FANT 1.E+3 LECT obst TERM
      FANT 1.E-3 LECT pscon TERM
```

Note that the Ogden material parameters (9) have been used without any conversion. This has to be verified according to the doubts expressed above about the dimensionality of the  $\alpha$  and  $\mu$  coefficients.

This calculation arrived at the final time of 4.0 ms in 10 185 time steps and 26 644 s (7.4 hours) of CPU time. The average  $\Delta t$  was  $\Delta t_{ave} = 0.4 \mu s$ . It was therefore significantly (almost twice) faster than solution `head36` of reference [3] with the previous set of Ogden material parameters, which had used 13 716 steps and 13.7 hours of CPU for 4.0 ms.

Figure 14 compares the brain/skull contacts in the two solutions. With the new Ogden material parameters (set 1) proposed in this study no contact occurs, probably due to the fact that the brain material is much softer than in the previous solution, with respect to the CSF. Consequently, the CSF fluid layer is less deformed during the impact, no contacts occur and the critical time increment (which usually occurs in the CSF) stays larger. This explains why the numerical solution is twice faster with the new set of material parameters.

Figure 15 compares the brain pressure in the two solutions. The solutions are qualitatively similar in the initial part of the impact until rebound occurs. In the rebound phase the new solution looks more convincing than the previous one. There is no *back contact* since the posterior gap remains quite open, and no significant pressure remains in the brain after rebound.



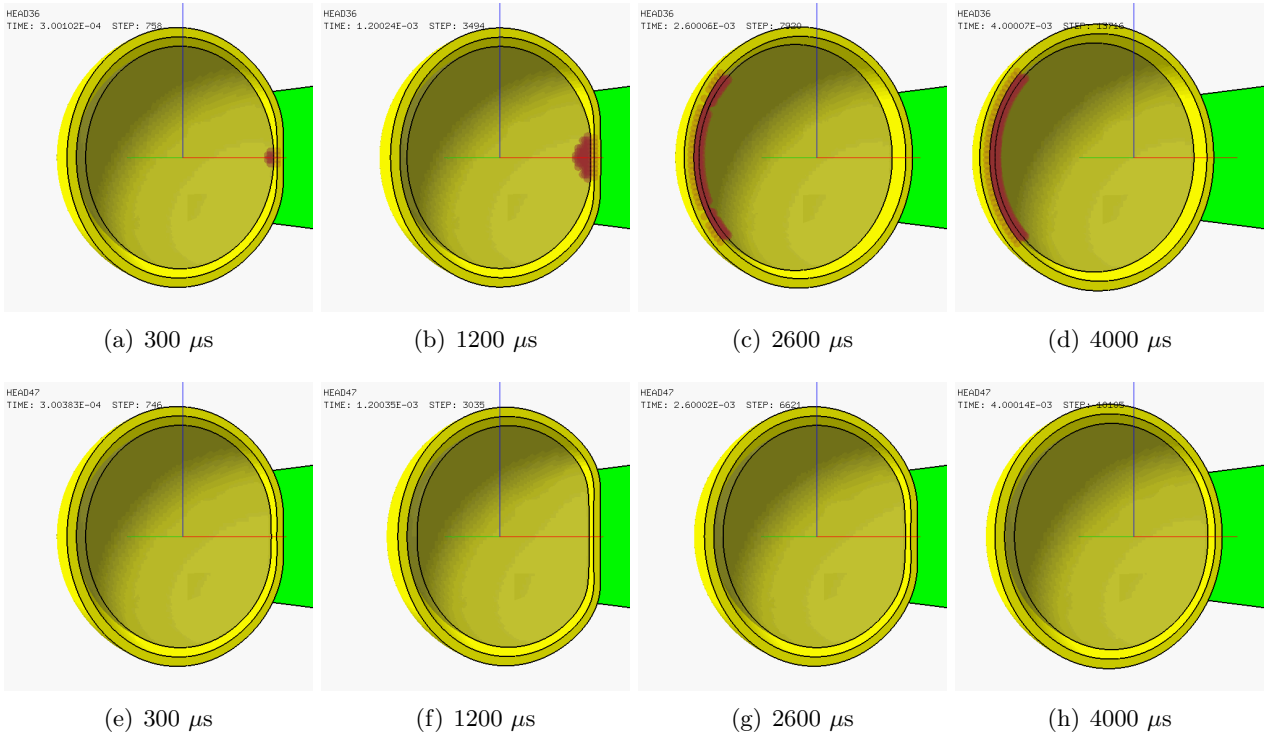


Figure 14: Comparison of contacts for Ogden [3] (top row) and Ogden type 1 (bottom row) materials.

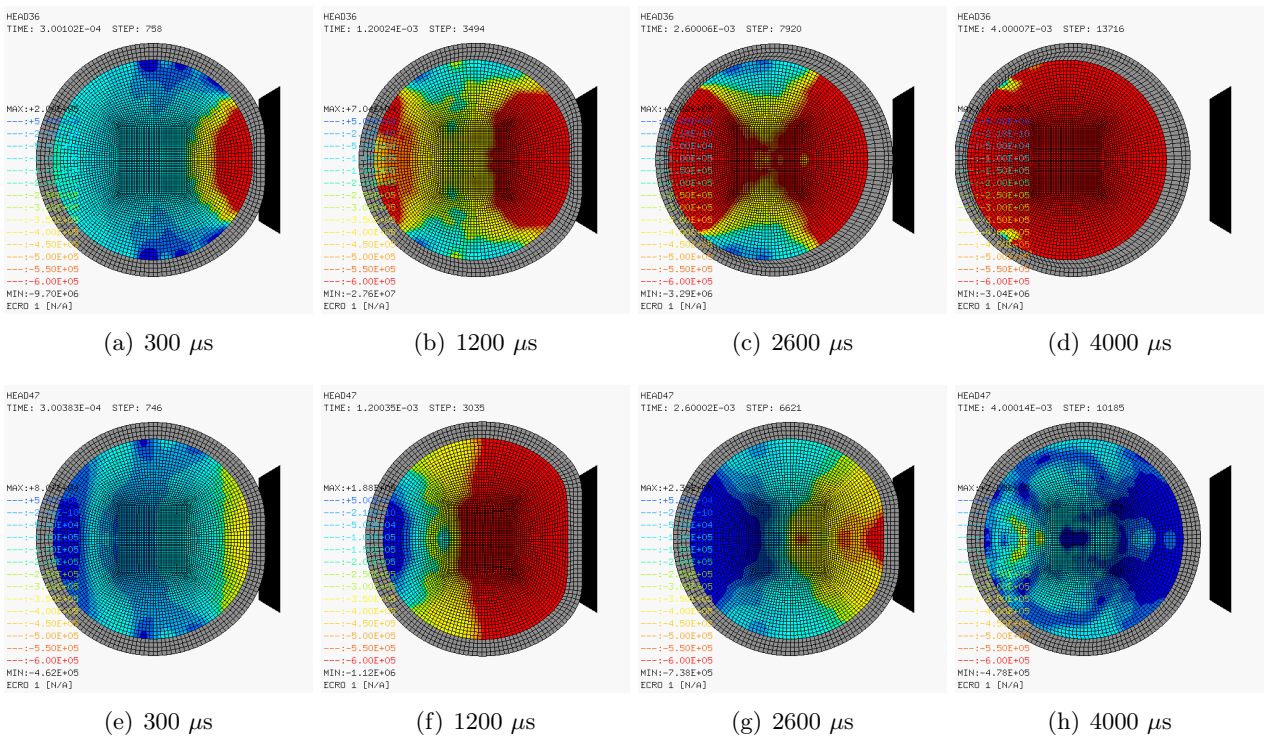


Figure 15: Comparison of brain pressures for Ogden [3] (top row) and Ogden type 1 (bottom row) materials.

### 3.3.4 Ogden set 2 test head48

This calculation is a repetition of case head47 presented above but by using the Ogden set 2 (10) material instead of the Ogden set 1 material in the brain. The materials input data become:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new formulation)
      BULK 128E+06             ! This is in Pa
```

```

C01 23.5 ! This is alpha_1
C02 -4.0 ! This is alpha_2
C05 8.32E-4 ! This is mu_1
C06 -0.41 ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM

```

This calculation arrived at the final time of 4.0 ms in 10 183 time steps and 26 706 s (7.4 hours) of CPU time. The average  $\Delta t$  was  $\Delta t_{ave} = 0.4 \mu s$ . The calculation time and also the results were almost identical to those obtained in the previous solution `head47` with the Ogden set 1 material. Figure 15 compares the brain pressure in the two solutions.

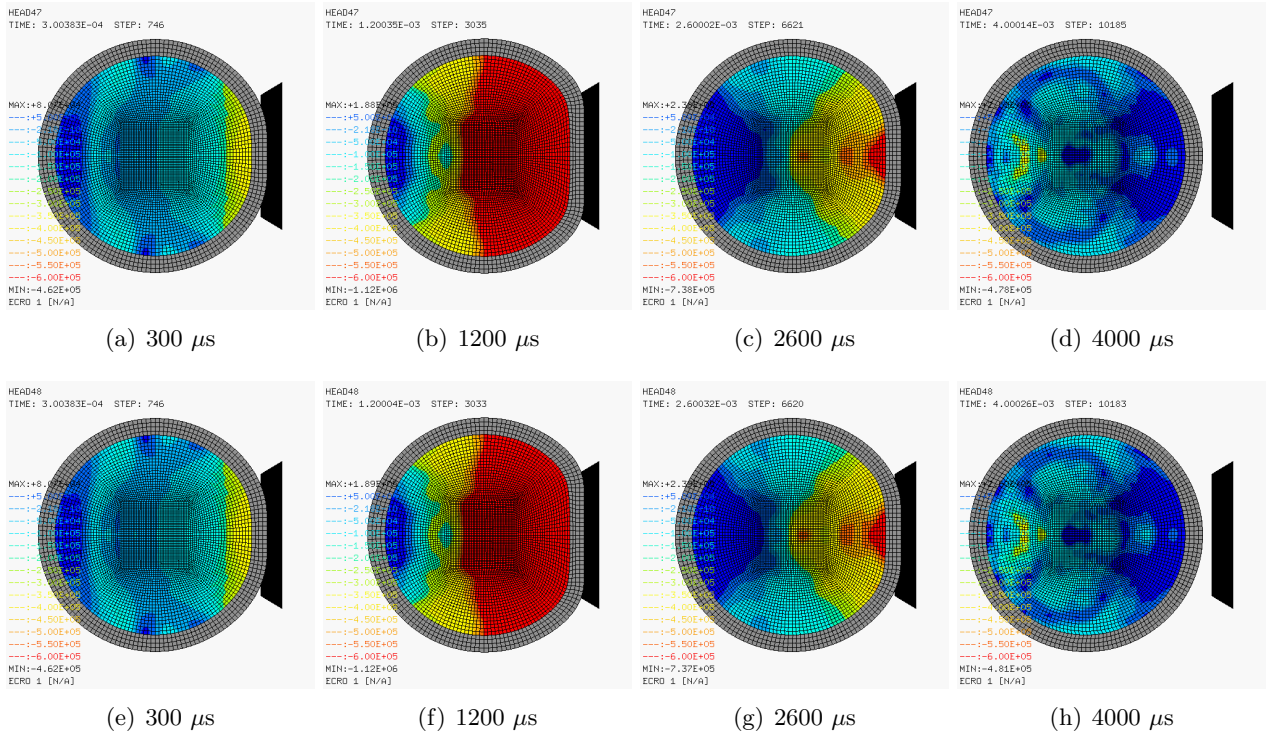


Figure 16: Comparison of brain pressures for Ogden type 1 (top row) and Ogden type 2 (bottom row) materials.

From this comparison it is tentatively concluded that the Ogden set 1 and Ogden set 2 material give almost identical results so that only Ogden set 1 will be used in the forthcoming calculations.

### 3.4 Preliminary 3D simulations with the second volumetric mesh

Some preliminary 3D simulations of the rat head were performed by using the second volumetric mesh described in Section 2.2 and the improved material parameters described in Section 3.2.

#### 3.4.1 Cases `brat06`, `brat07` and `brat08`

The test `brat06` uses the second volumetric mesh of the rat head, from file `head.msh`. The mesh name is the same as the one containing the first volumetric mesh, which can be quite confusing. However, the sizes of the two files and the number of elements and nodes in the two meshes are different, and this can be used to distinguish them in case of doubts.

The mesh components have been already summarized in Table 4 of Section 2.2. Table 11 below adds some more detailed information about nodes and elements numbering, which will be useful in some following discussions.

As already noted for the first volumetric mesh of the rat head (see Section 3.1), all dimensions in the mesh file are in non-standard units i.e. in mm instead of m. In order to convert them to m, a scaling is applied by means of command `SCAL FACT 0.001` in the EPX input file. After scaling, the overall extensions of the mesh are as follows:  $L_x = 11.2921$  mm,  $L_y = 16.6593$  mm and  $L_z = 7.73530$  mm.

| Component | Nodes   | From   | To      | Elements | From    | To      | Element type |
|-----------|---------|--------|---------|----------|---------|---------|--------------|
| Brain     | 57 938  | 1      | 57 938  | 312 755  | 1       | 312 755 | TET4         |
| CSF       | 35 582  | 57 939 | 93 520  | 145 746  | 312 756 | 458 501 | TET4         |
| Skull     | 10 865  | 93 521 | 104 385 | 21 726   | 458 502 | 480 227 | TRI3         |
| Total     | 104 385 |        |         | 480 227  |         |         |              |

Table 11: Details of the second volumetric mesh for the rat head (file `head.msh`).

As concerns materials, the Ogden set 1 material (9) is used for the brain. The CSF is modelled by the FLUI fluid material as in previous simulations. Finally, the linear elastic (LINE) material is used for the skull, but this is irrelevant since all motions of the skull are prescribed. The materials definition in the EPX input file is therefore:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8   ! This is alpha_1
      CO2  19.7   ! This is alpha_2
      CO5 -0.29   ! This is mu_1
      CO6  9.6E-3 ! This is mu_2
      LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
      LECT skull TERM
```

An initial velocity  $v_0 = 0.1$  m/s in the  $y$  direction is given to the whole model. The grid velocity is set to the same value ( $w_0 = v_0$ ). In order to simulate a uniform deceleration of the skull, treated as rigid, all degrees of freedom of the skull nodes are blocked except translation along the  $y$  direction. The latter is imposed to follow the time function shown in Figure 17, that is a constant deceleration from 0.1 m/s at the initial time  $t_0 = 0$  down to 0.0 m/s at  $t_1 = 25$  ms, followed by an imposed zero velocity until the chosen final time  $t_f = 50$  ms.

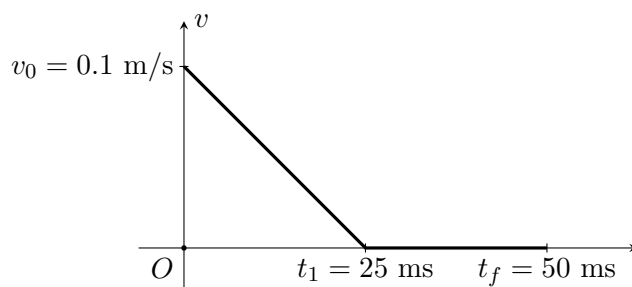


Figure 17: Time evolution of the skull velocity.

The EPX input commands for this are:

```
FONC 1 TABL 3 0.0 1.0
      25.E-3 0.0
      50.E-3 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
      BLOQ 13456 LECT skull TERM
      VITE 2 0.1 FONC 1 LECT skull TERM
      FSA LECT fsan TERM
INIT VITE 2 0.1 LECT brain csf skull TERM
      VITG 2 0.1 LECT brain csf skull TERM
```

The minimum and maximum intra-nodal distances in the elements of the various components are summarized in Table 12, which could be compared with the analogous Table 10 for the first volumetric mesh of the rat head.

| Component | $d_{\min}$ | $d_{\max}$ | Ratio |
|-----------|------------|------------|-------|
| Brain     | 28.0       | 550.0      | 20    |
| CSF       | 20.0       | 394.1      | 14    |
| Skull     | 60.8       | 336.8      | 6     |

Table 12: Minimum and maximum element sizes ( $\mu\text{m}$ ) of the head components in case **brat06**.

The overall minimum intra-nodal distance passes from  $1.5 \mu\text{m}$  of the first mesh to  $28 \mu\text{m}$  and the spread of element sizes is much lower (from 303 to 20). Inspection of the elements' stability by means of the **STEL** optional keyword in **EPX** shows that the critical step (multiplied by a safety factor **CSTA** equal to 0.5) is now  $3.36 \times 10^{-9}$  s, a value larger than the  $3.66 \times 10^{-10}$  s of the first mesh but still very low, while the maximum step has grown up to  $2.61 \times 10^{-3}$  s (it was  $3.14 \times 10^{-7}$  s with the first mesh, see case **brat04**).

The main reason for the large increase in the maximum stability step is the use of a different set of material parameters in the brain, i.e. Ogden set 1, eqs. (9), instead of the the old Ogden parameters from the study [3], eqs. (2). Unfortunately, the *minimum* stability step remains very low, since the most critical element occurs in the CSF, which is modelled by the **FLUI** material with the same parameters in both cases.

The stability histogram of the elements shows that all the 145 746 elements of the CSF fall within the first 1 % of the histogram ( $3.36 \times 10^{-9} \leq \Delta t \leq 2.61 \times 10^{-5}$ ), i.e. they have much smaller stability steps than the brain or skull elements.

Despite the very unfavorable stability step, the calculation was tentatively run until a time of  $5.82 \mu\text{s}$ , which was reached in 1733 steps and 4734 s (1.3 hours) of CPU time. Very high (unphysical) velocities (424 m/s) developed in a CSF fluid node along the F-S interface, located in a thin groove of the brain, see Figure 18(a). Also, very large local pressures were observed at the same location, as shown in Figure 18(b), which might indicate a local instability.

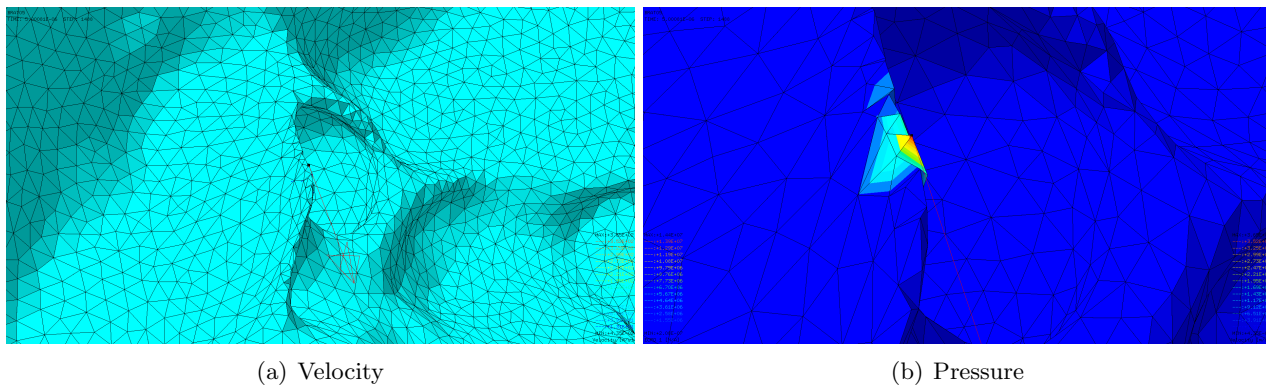


Figure 18: Local velocity and pressure in the CSF in case **brat06**.

Test case **brat07** was a repetition of case **brat06** by using spatial time step partitioning (**PART**) instead of an automatic but uniform time increment over the whole mesh (**PAS AUTO**). Since the ratio between the maximum and minimum stability step in this model is very high (of the order of 1 million), in theory the partitioning algorithm could bring a benefit in the cost of the solution.

However, in the present case the fraction of elements in the lowest levels of the partition is too high. These are all the CSF elements. Therefore, no important speed-up was observed and the calculation was stopped.

Finally, test case **brat08** was a repetition of test **brat06** with the **STEL** option activated in order to obtain the time step distribution over the elements. These results have been already commented above for the case **brat06** and need not be repeated here.

### 3.5 Stability step in hyperelastic materials

A small set of ad-hoc tests was performed to verify the treatment of stability in hyperelastic materials, i.e. the evaluation of the sound speed. This means in particular the Ogden material since, as already mentioned in Section 3.3.2, the current Mooney-Rivlin implementation does not deal with stability at all (this will have to be corrected in the code).

#### 3.5.1 Stability of a single tetrahedron (cases delt01 to delt04)

A single tetrahedron element of side 1 m, shown in Figure 19, is associated with the Ogden set 1 material (9) in test delt01. The EPX input file reads:

```

DELTO1
ECHO
  CONV WIN
LAGR TRID
GEOM LIBR POIN 4 TETR 1 TERM
  0 0 0 1 0 0 0 1 0 0 0 1
  1 2 3 4
MATE HYPE TYPE 4 RO 1040      ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8   ! This is alpha_1
      CO2  19.7   ! This is alpha_2
      CO5 -0.29   ! This is mu_1
      CO6  9.6E-3 ! This is mu_2
      LECT 1 TERM
ECRI DEPL VITE ACCE FINT FEXT CONT EPST ECRO FREQ 1
OPTI PAS AUTO NOTE LOG 1
      STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

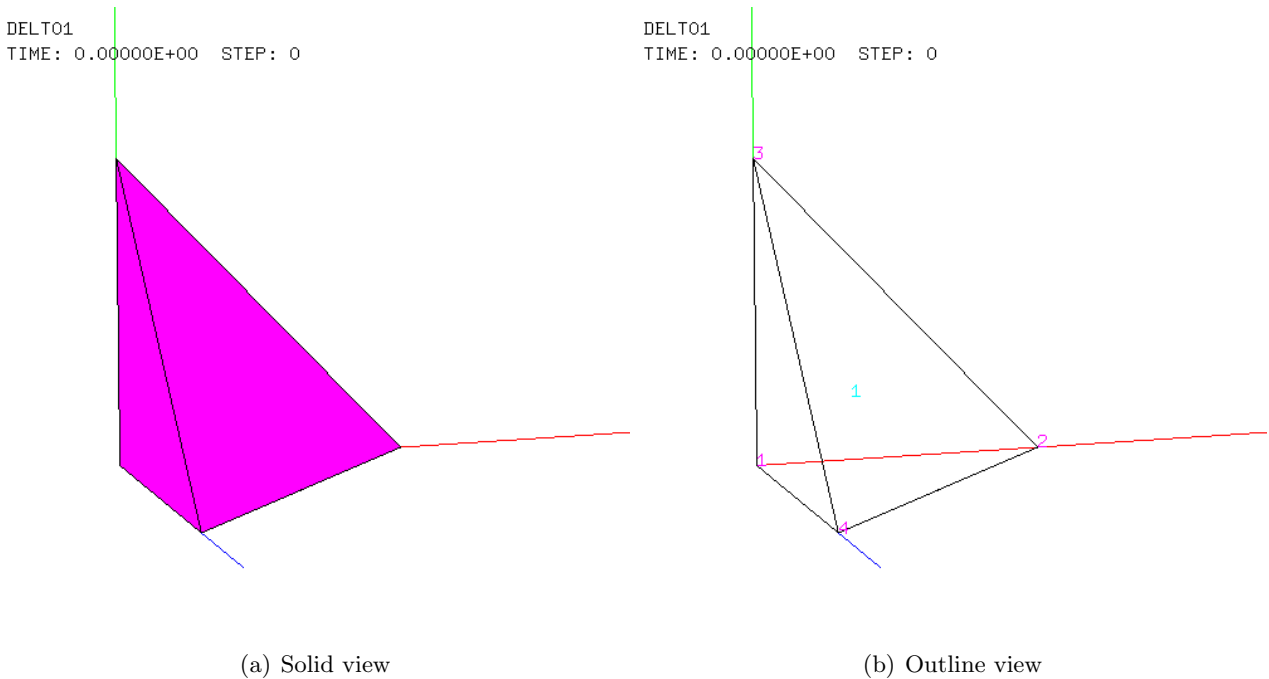


Figure 19: Single tetrahedron element used in test delt01.

The computed sound speed  $c$  in the material is not directly accessible in EPX for the HYPE material (this will have to be added), but one can find it indirectly from the stability step  $\Delta t_{\text{stab}}$  by knowing that:

$$\Delta t_{\text{stab}} = \phi \frac{L}{c} \quad (11)$$

where  $\phi$  is the safety coefficient (CSTA in EPX) and  $L$  is the element's characteristic length. In this test CSTA has not been set explicitly, so it has the default value  $\phi = 0.8$  in EPX.

The element's characteristic length depends on the type of element but in general it is computed as the *minimum length* of the element, taking into account both the element's sides (that is, the intranodal distances) and the element's *heights* (if any). In the particular case considered here it seems that  $L$  should be the height of the tetrahedron with respect to the face not parallel to the coordinate planes (face 234 in Figure 19(b)). Since the side length is 1.0 one can easily find that  $L = 0.577$ .

From EPX we obtain  $\Delta t_{\text{stab}} = 8.261$  s and by using (11):

$$c = \phi \frac{L}{\Delta t_{\text{stab}}} = 0.8 \frac{0.577}{8.261} = 5.588 \times 10^{-2} \quad (12)$$

that is the sound speed in the Ogden set 1 material seems to be as low as  $c_{\text{brain}} = 5.6$  cm/s. Comparing this with the sound speed assigned so far to the CSF fluid (FLUI) material ( $c_{\text{CSF}} = 1451$  m/s) explains why there are so dramatic differences between the estimated stability steps of the brain and of the CSF in the simulations performed so far:  $c_{\text{CSF}}/c_{\text{brain}} = 25\,966$ .

It remains to be verified whether this low value of sound speed is realistic, if it is correctly computed, and how it varies (if it varies) with the state of stress in the brain material.

A consistency check is run (case `delt02`) by using exactly the same geometry as `delt01` but with the FLUI material in the tetrahedron. The result is  $\Delta t_{\text{stab}} = 3.183 \times 10^{-4}$ , i.e.  $1/25\,966$  of the value with the Ogden set 1 material. This confirms the above analysis.

In order to exclude (for prudence) any scale effects in the estimation of the stability step, the two tests are repeated by using a tetrahedron of 1 mm side instead of 1 m, see cases `delt03` and `delt04` in Appendix. The results are  $\Delta t_{\text{stab}} = 8.261 \times 10^{-3}$  and  $\Delta t_{\text{stab}} = 3.183 \times 10^{-7}$ , respectively, i.e.  $1/1000$  of the previous values, as it should be.

### 3.5.2 Stability in a bar impact test (cases `hype01` and `hype02`)

Next, some simple tests are conducted in order to check the *variation* of sound speed (variation of stability) in the hyperelastic material when it is subjected to stress, i.e. when the associated elements get heavily deformed.

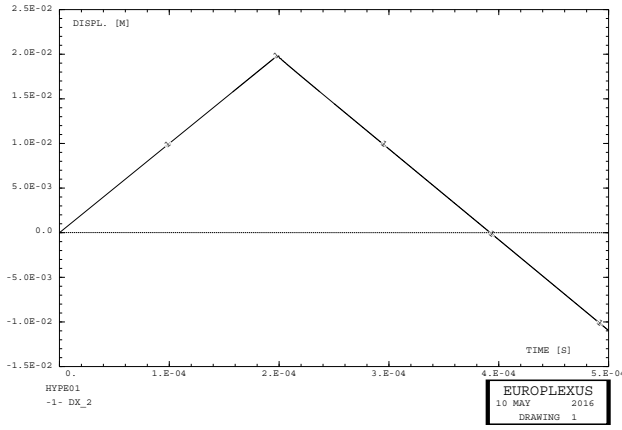
The tests simulate the impact of a prismatic bar of length  $L = 1$  m and square cross section of side  $s = 0.001$  m on a rigid wall. The bar has a uniform initial velocity  $v_0$ . The mesh consists of a row of 1000 CUBE elements discretizing the bar. The elements are regular cubes of size  $h = 1$  mm.

cross section of side  $s = 0.001$  m on a rigid wall. The bar has a uniform initial velocity  $v_0 = 100$  m/s.

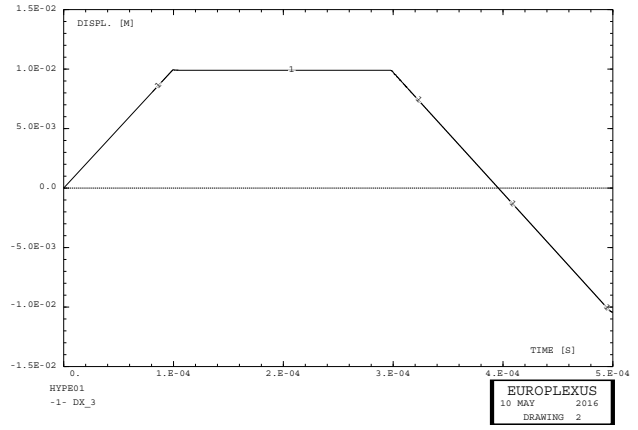
In the first solution (test `hype01`), to be used as a qualitative reference, the bar is made of a linear elastic (steel-like) material with density  $\rho = 8000$ , Young's modulus  $E = 2.0 \times 10^{11}$  Pa and Poisson's coefficient  $\nu = 0$  (which is unrealistic, but useful to avoid lateral effects). The initial velocity is set to  $v_0 = 100$  m/s. The solution of the problem up to  $t_f = 0.5$  ms is shown in Figure 20. The pictures 20(a) and 20(b) show the displacement of the free extremity  $p_1$  of the bar and of the mid-point  $p_m$ . The pictures 20(c) and 20(d) show the stress in the elements  $e_m^-$  and  $e_m^+$  immediately to the left and to the right of the mid-point, respectively. This solution coincides with the analytical solution of the problem if one neglects the stress oscillations due to the linear elastic material and to the assumed discretization.

The next solution (`hype02`) uses the same geometry but the elastic steel-like material is replaced by Ogden set 1 hyperelastic material (9). Since this material is much softer than steel, the impact velocity is tentatively reduced to just 1 m/s. Despite this, the calculation immediately stops at step 1 because the last element of the bar (the one in contact with the obstacle) gets entangled.

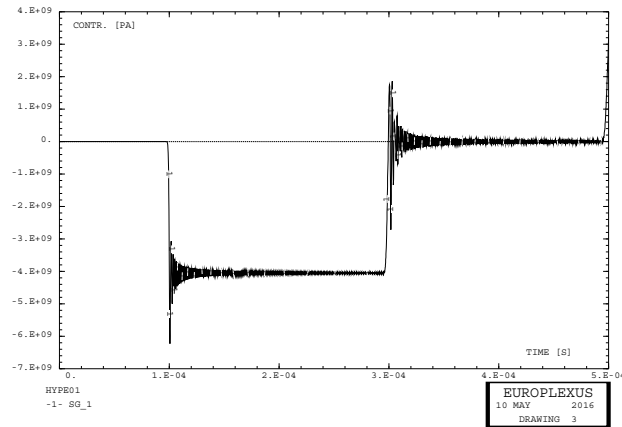
Inspection of results reveals that EPX estimates a stability step of  $1.239 \times 10^{-2}$  s, which is coherent with the speed of sound in the material which has been estimated in the previous tests. In fact, assuming  $c = 5.588 \times 10^{-2}$  and an element length  $L = 1.0 \times 10^{-3}$  one would obtain  $\Delta t_{\text{stab}} = \phi L/c = 1.432 \times 10^{-2}$ , a value which is close to the one reported by EPX (small differences may be due to the fact that the expressions used in this estimation are just an approximation of the real ones).



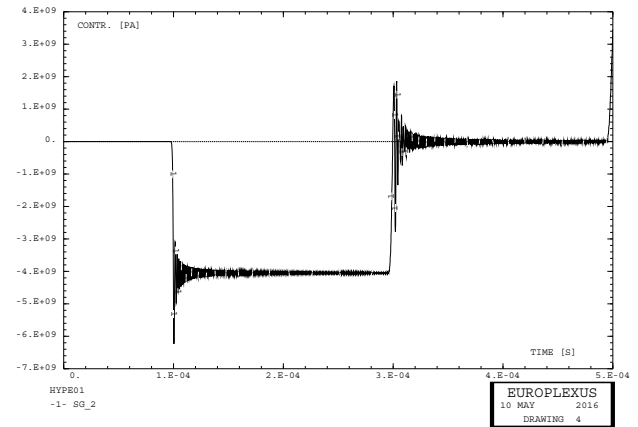
(a) Displacement of  $p_1$



(b) Displacement of  $p_m$



(c) Stress at  $e_m^-$



(d) Stress at  $e_m^+$

Figure 20: Displacements and stresses vs. time in case hype01.

However, with such a time increment and an impact velocity  $v_0 = 1$  m/s the bar moves by  $u = v_0 \Delta t = 1.0 \times 1.239 \times 10^{-2} = 1.239 \times 10^{-2}$  m in the first step, which is far *more* than one element length. This explains while the contacting element becomes entangled.

With the chosen impact velocity, which is very small for typical EPX applications but very large for this particular material, the impact is *supersonic*:  $v_0 > c$ . This simple simulation shows that EPX is unable to treat a supersonic impact in this case since the estimation of stability is based upon the sound speed and not on the maximum between the sound speed and the impact speed (or more precisely, the maximum relative velocity between two nodes of an element, which could be called the *element deforming velocity*).

This problem will have to be solved by a suitable development in EPX, for example a new optional keyword that instructs the code that a supersonic impact is expected so that the element stability must be estimated in the more general way outlined above.

For the sake of conducting the present simple test, it is sufficient to run the case with a user-imposed time increment (OPTI PAS UTIL) instead of letting EPX compute it automatically (OPTI PAS AUTO, the default). The actual critical step in this simple case can be estimated as:

$$\Delta t_{\text{crit}} = \frac{L}{\max(c, v_r)} = \frac{0.001}{\max(5.588 \times 10^{-2}, 1.0)} = 0.001 \quad (13)$$

where the initial length  $L = 1$  mm of an undeformed element has been used, and the impact velocity  $v_0 = 1$  m/s has been used as the relative (element deforming) velocity  $v_r$ . Since the elements will get squeezed during the impact, a value (much) smaller than (13) should be taken, to ensure stability. For example, by taking  $\Delta t_{\text{fixed}} = 1.0 \times 10^{-6}$  s the calculation is stable and runs until the final chosen time  $t_f = 50.0$  ms.



At a velocity of 1 m/s the free end of the bar moves over a distance of 50 mm in  $t_f = 50$  ms, so the length of the bar reduces to 0.95 m from the initial 1.0 m (this of course by assuming that the perturbation due to the impact has not yet reached the free end at  $t = t_f$ , which turns out to be the case here).

It is indeed quite surprising that one has to take such a small time increment ( $\Delta t_{\text{fixed}} = 1.0 \times 10^{-6}$  s) when the stability step estimated by the supposedly right relation (13) taking into account both the sound and the impact speed is  $\Delta t_{\text{crit}} = 0.001$ , i.e. 1000 times larger. (By the way, using e.g.  $\Delta t_{\text{fixed}} = 1.0 \times 10^{-5}$  s turns out to be unstable already during the first time steps.)

Some results of this simulation are presented next. Figure 21 shows the longitudinal stress distribution along the bar at the final time. The stress perturbation has propagated over the last 100 mm (1/10) of the original bar length. Note that the bar has shortened by about 50 mm at this time so that the maximum abscissa (which is measured from the left, unloaded end of the bar) has reduced to about 0.95 m from the initial 1.0 m. A sort of *precursor* wave of non-completely negligible amplitude (1/20 of the main wave or so) seems to precede the main wave, traveling at a much higher speed. Whether this phenomenon is physical or just numerical, it will have to be investigated.

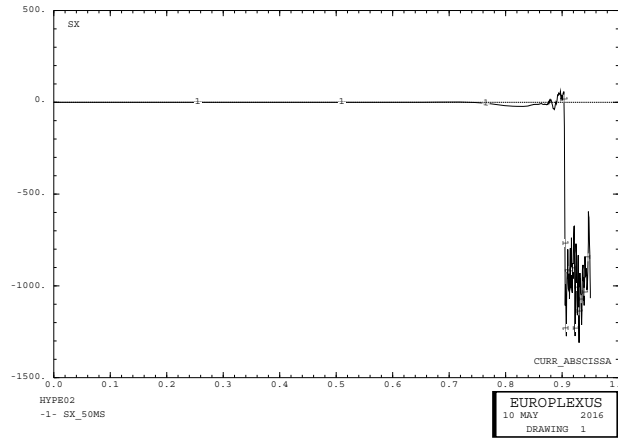
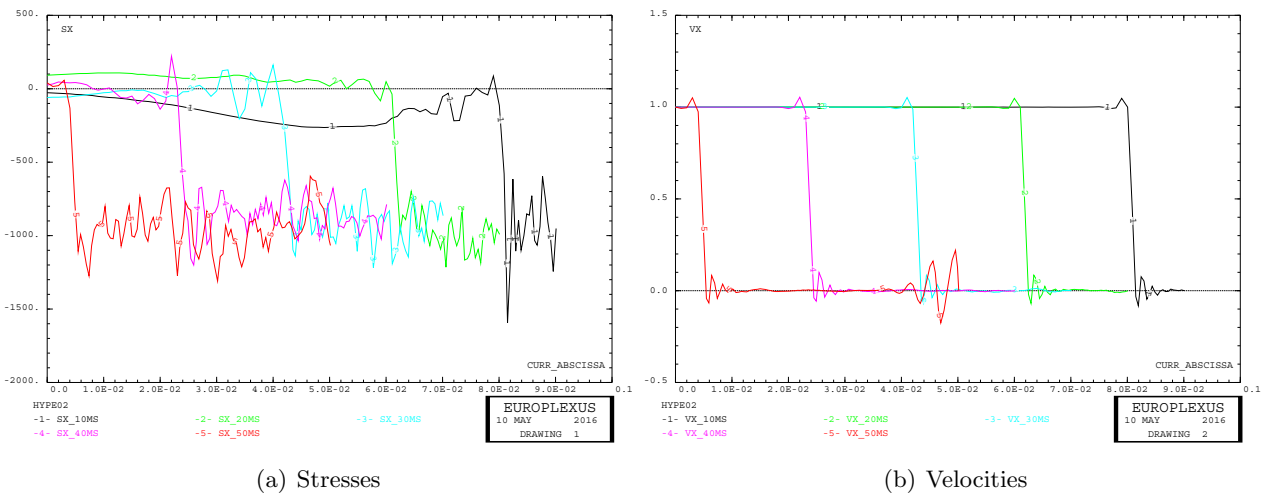


Figure 21: Stress along the bar at  $t = 50$  ms in case hype02.

Figure 22 shows the longitudinal stress and velocity distributions along the bar every 10 ms until the final time. The stress and velocity waves seem to propagate at constant speed.



(a) Stresses

(b) Velocities

Figure 22: Stresses and velocities along the bar every 10 ms in case hype02.

The deformation of the bar near the impact zone can be appreciated from Figures 23 (showing the last 10 elements, i.e. the last 10 mm, and Figure 24, showing the last 50 elements. The deformation of the elements is very large, as is typical of hyperelastic material applications. Near the final time



( $t_f = 50$  ms) an instability (lateral buckling) starts to develop near the loaded end of the bar. This is normal since the bar is very slender and no constraints are placed on lateral displacements.

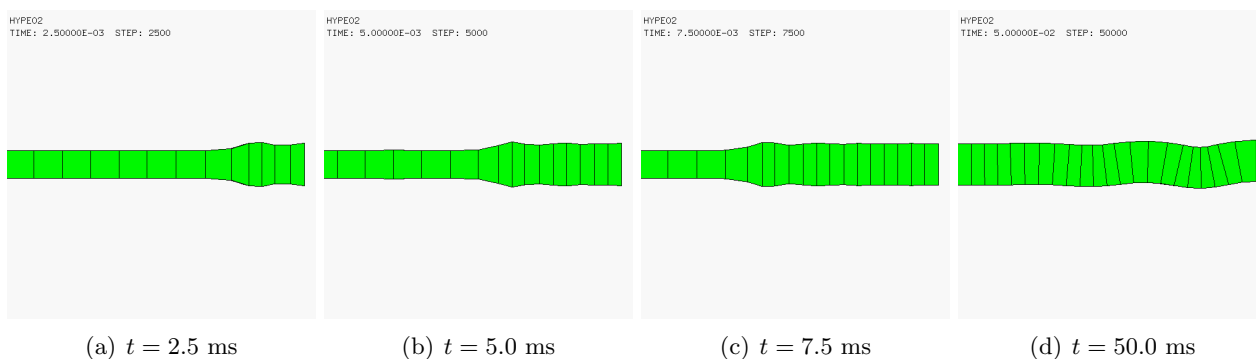


Figure 23: Deformations near the impact zone (last 10 mm) in case `hype02`.

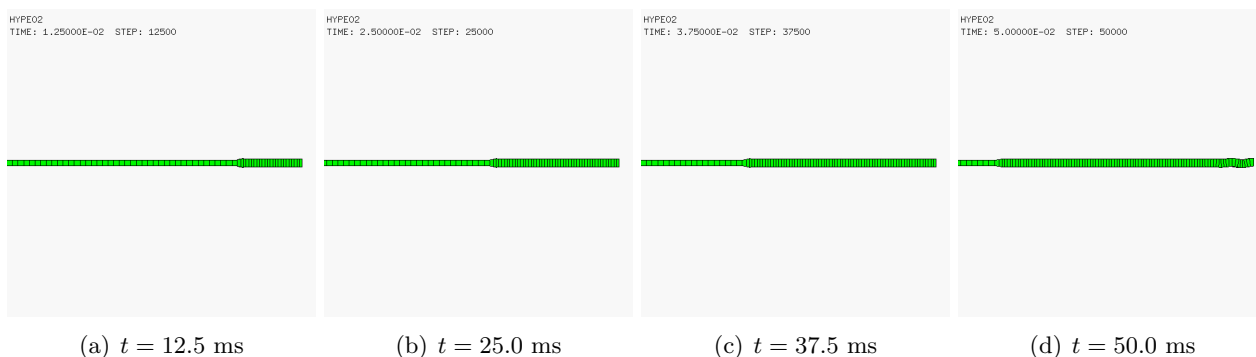


Figure 24: Deformations near the impact zone (last 50 mm) in case `hype02`.

If one looks at the results printed by EPX on the log file `hype02.log` one sees that the minimum stability step drops dramatically from  $1.239 \times 10^{-2}$  at step 0 (stress-free bar) to  $\mathcal{O}(1.0 \times 10^{-6})$  already at step 1, and then stays almost constant for the whole rest of the computation (while the maximum step remains constant at  $1.239 \times 10^{-2}$ , in all the elements not yet reached by the perturbation).

This seems to indicate two things:

- The code computes very different (by orders of magnitude) sound speeds in stress-free *vs.* loaded elements. This should be corrected since when the loading first arrives on an element it might be too late to reduce the step (the element gets entangled in just one step).
- The real sound speed in the material is the one resulting from the loaded elements, not from the stress-free ones.

Based on the second observation above, it would then result that a more realistic value of the sound speed in this material is:

$$c = \phi \frac{L}{\Delta t_{\text{stab}}} = 0.8 \frac{0.001}{1.2 \times 10^{-6}} = 670 \text{ m/s} \quad (14)$$

and not 5.6 cm/s as estimated previously by eq. (12). Under these circumstances, the impact would no longer be supersonic since  $v_0 < c$ .

An indirect estimation of the sound speed could perhaps be obtained from the propagation speed  $s$  of the compression waves observed in the bar. From Figure 22(b) one can estimate that the compression (or velocity) wave has traveled over a distance of 95 mm in 50 ms, which gives  $s \approx 19$  m/s. If the material would be linear elastic, one could use the well-known relation  $s/c = \sigma/E$ , where  $\sigma$  is the uniaxial stress, to estimate  $c$ . In this case from Figure 22(a) one sees that  $\sigma \approx 1000$  Pa. However,

it is not known whether the above relation can be applied for a hyperelastic material, nor what would be an equivalent value of  $E$  (the Young's modulus) to be used in this case.

### 3.5.3 Behaviour of a single hexahedron (cases hype03 to hype05)

As a last check of hyperelastic material behaviour, we would like to see the influence of the chosen material bulk modulus  $K$  and to this end we consider a single hexahedron of unit side subjected to imposed displacements, see Figure 25.

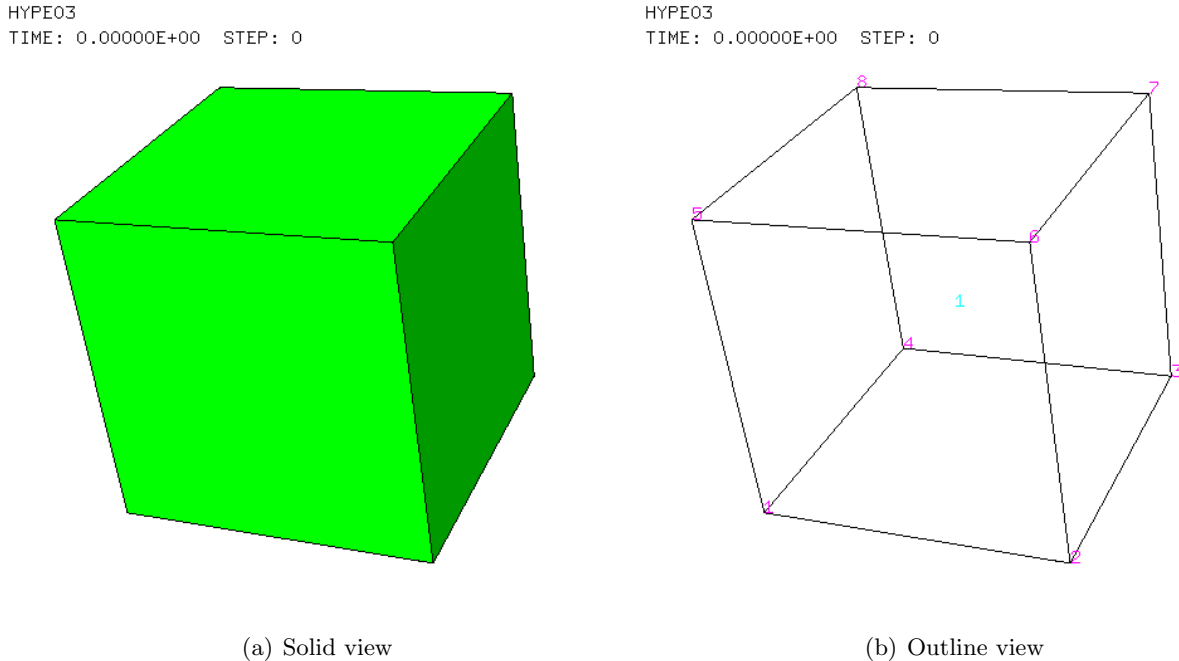


Figure 25: Single hexahedron element used in test `hype03`.

The EPX input file for the first simulation of this problem (case `hype03`) reads:

```

HYPE03
ECHO
  CONV win
TRID LAGR
GEOM LIBR POIN 8 CUB8 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 2 3 4 5 6 7 8
COMP COUL VERT LECT 1 TERM
MATE HYPE TYPE 4 RO 1040      ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8   ! This is alpha_1
      CO2  19.7   ! This is alpha_2
      CO5 -0.29   ! This is mu_1
      CO6  9.6E-3 ! This is mu_2
      LECT 1 TERM
FONC 1 TABL 6 0 1 0.8 1 0.8001 0 1.1901 0 1.2 -1 10.0 -1
LINK COUP
  BLOQ 3 LECT 1 2 3 4 TERM
  VITE 3 -1.0 FONC 1 LECT 5 6 7 8 TERM
INIT VITE 3 -1.0 LECT 5 6 7 8 TERM
ECRI DEPL VITE CONT ECRO EPST TFRE 0.1
  FICH ALIC FREQ 1
OPTI PAS UTIL NOTE
  LOG 1
  STEL
CALC TINI 0. TEND 0.88 PASF 1.E-3
FIN

```

An Ogden set 1 material is chosen with  $K = 128 \times 10^6$ , like in all simulations performed so far with this material,

The lower face of the hexahedron is blocked vertically while the upper face is subjected to an imposed vertical velocity following the time function shown in Figure 26(a). This results in the displacement time history shown in Figure 26(b), that is, the cube is compressed at constant speed down to 20 % of its initial height in 0.8 s, then it is kept fixed for 0.4 s (to let any elastic waves fade out), and finally it is uncompressed back to the initial height in 0.8 s. The total duration of the numerical experiment is 2.0 s. Since the material is (hyper-)elastic, it is expected that zero stress and zero strain states be recovered at the end of the numerical experiment.

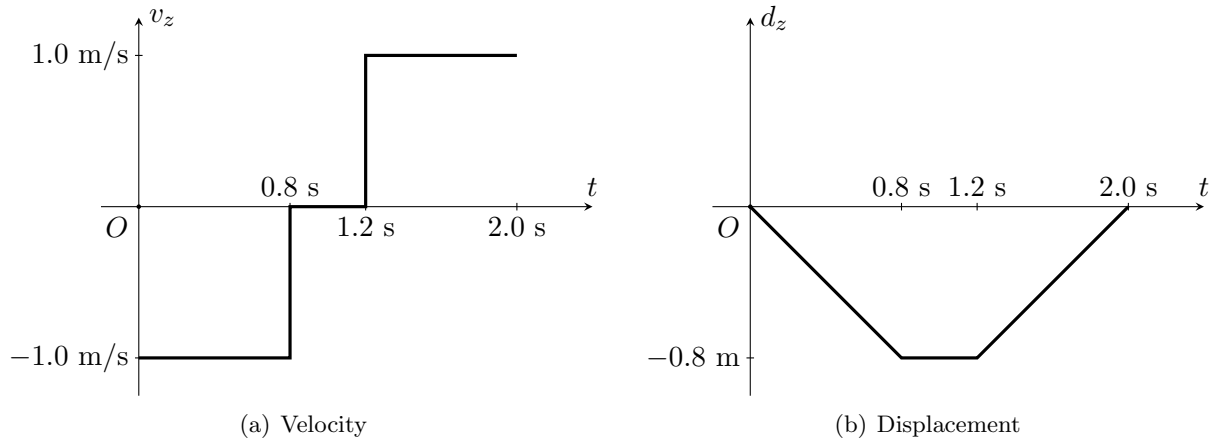


Figure 26: Time evolution of the upper face velocity and displacement.

A user-imposed time step of 1.0 ms is chosen which, according to the stability values printed out by EPX *after* step 0 (of the order of 2.0 ms at the least), should be on the safe side of stability. The value of stability printed out at step 0 (unloaded material) is unrealistically high, as already noted, Unfortunately the calculation shows some oscillations at the end of the compression phase and gets unstable at about 0.88 s in the holding phase.

Figure 27 shows some configurations of the body: the initial one, the one at the end of the compression phase, and the one just before instability. The small visual artifacts are due to the superposition of (nominally coinciding) initial and deformed configurations.

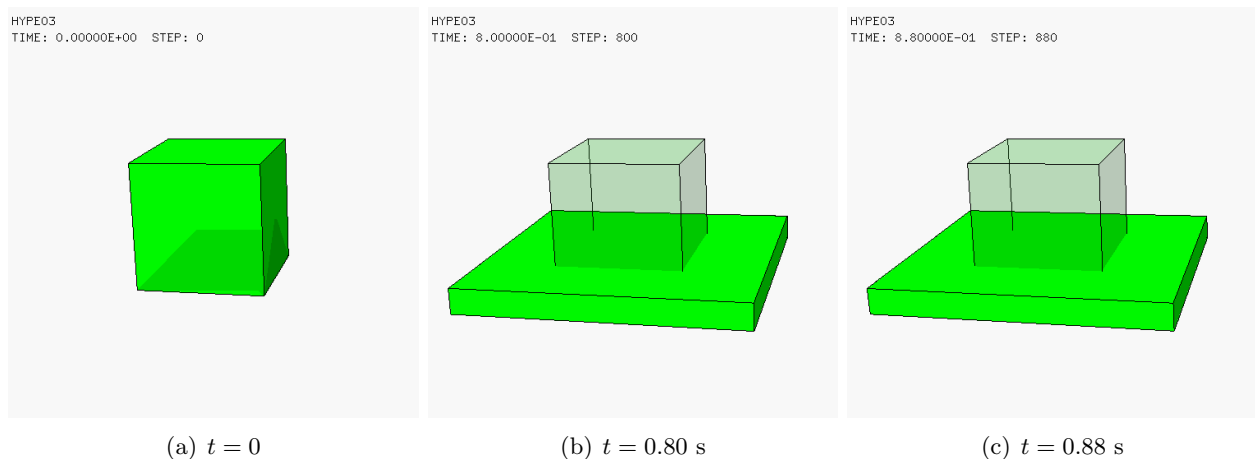


Figure 27: Initial and deformed configurations in test case hype03.

Figure 28 shows the normal stresses and the normal strains computed in this test. In red are indicated the components along the imposed motion ( $\sigma_{zz}$  and  $\epsilon_{zz}$ ). Some oscillations are observed during the compression phase, which is normal for a (hyper-)elastic material. However, their amplitude grows as soon as the compression is stopped and finally leads to instability. The strains are the *natural* values, as usual in EPX. The maximum value of  $\epsilon_{zz}$  results from the imposed displacement:  $\epsilon_{zz} = \ln(L_z/L_{z0}) = \ln(0.2/1.0) = -1.609$ . No oscillations appear in this component since this is along

the imposed motion. The lateral strains are equal ( $\epsilon_{xx} = \epsilon_{yy}$ , as it should be) and start exhibiting some oscillations at the end of the compression phase. Some *spurious* shear strains and stresses (not shown for brevity) start developing just before the calculation stops due to instability.

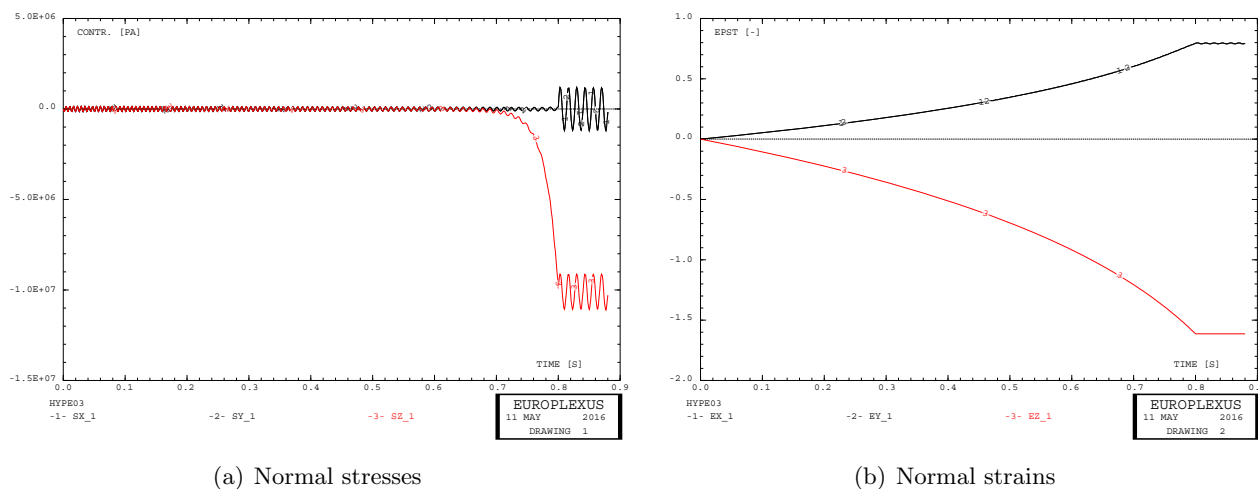


Figure 28: Normal stresses and normal strains in test case `hype03`.

Figure 29 shows some further results: the pressure (trace of the stress tensor), the von Mises equivalent stress, the energy potential used in the material law definition and the maximum time step for the element. The latter shows a weird pattern. It starts with the totally unrealistic value of 14.3 s at step 0, drops down to  $\mathcal{O}(1.0 \times 10^{-3})$  at step 1 and then stays relatively constant, but with some occasional spikes up to  $\mathcal{O}(1.0 \times 10^0)$ . This will have to be corrected in the code, as already mentioned in several occasions.

The next test case (`hype04`) is similar to `hype03` but the lateral motions of the hexahedron are completely blocked. As a consequence, the volume of the element is reduced to 20 % of the initial value at the end of the compression phase and returns to the initial value at the end of the test. This calculation runs without problems until the final time  $t_f = 2.0$  s. No oscillations are observed, but this is normal since all motions are imposed.

Figure 30 shows the initial, intermediate (fully compressed) and final (uncompressed) configurations of the body, with the usual small visual artifacts.

The normal stresses and strains are shown in Figure 31 and correspond to the expected values. Shear stresses and strains are rigorously zero.

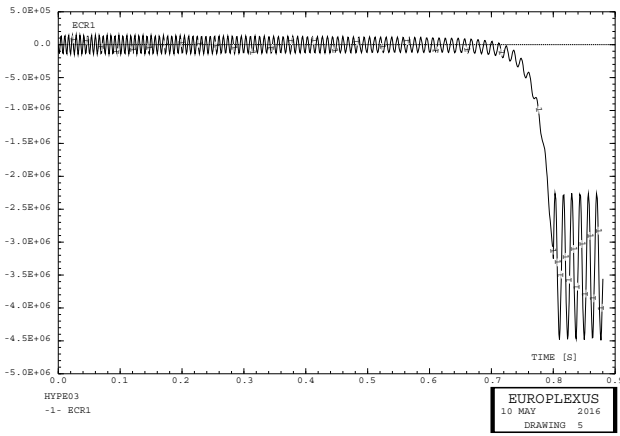
Figure 32 shows some further results: the pressure (trace of the stress tensor), the von Mises equivalent stress, the energy potential and the maximum time step for the element. The latter appears more regular than in the previous test case (but still wrong at step 0).

From the above results, it is not immediate to establish any connection between the assumed value of the bulk modulus ( $K = 128 \times 10^6$  Pa) and the stress or other quantities.

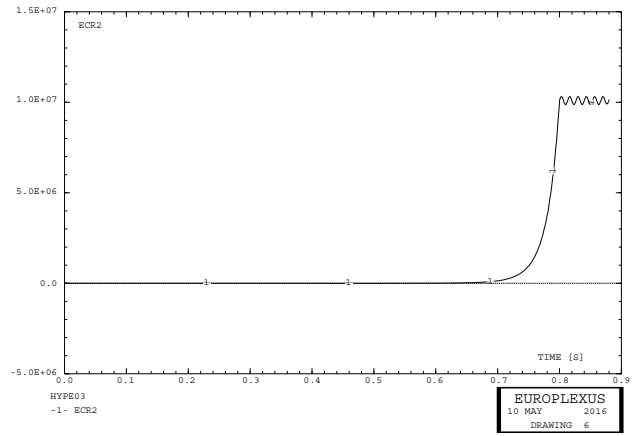
The next and final test case `hype05` is similar to `hype04` (lateral restraints) but the bulk modulus of the Ogden material is set to 0. According to an arbitrary interpretation of the User's manual (to be verified by the present test) this might indicate an *incompressible* material ( $K = \infty$ ).

The normal stresses and strains are shown in Figure 33. Shear stresses and strains are rigorously zero. The normal strains are identical to those of case `hype04`, as it should be. The stresses (see Figure 33(a)) look strange. The component  $\sigma_{zz}$  follows a similar pattern to the case with non-zero bulk modulus (see Figure 31(a)) but the values are 5000 times smaller. This seems to indicate that choosing  $K = 0$  produces a more (infinitely?) compressible rather than a fully incompressible material, thus contradicting the tentative interpretation of the Users' manual mentioned above. Strangely, positive (traction) stresses develop in the lateral directions, of magnitude comparable with the main (compressive) stress. These strange results will have to be checked in detail.

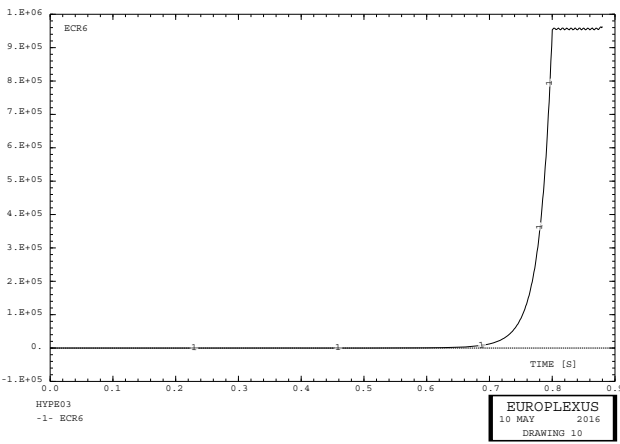
Figure 34 shows the pressure (trace of the stress tensor), the von Mises equivalent stress, the energy potential and the maximum time step for the element. The pressure is numerically zero, which



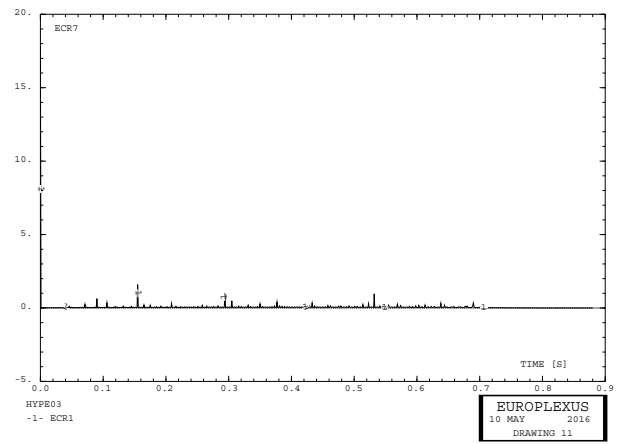
(a) Pressure



(b) von Mises stress

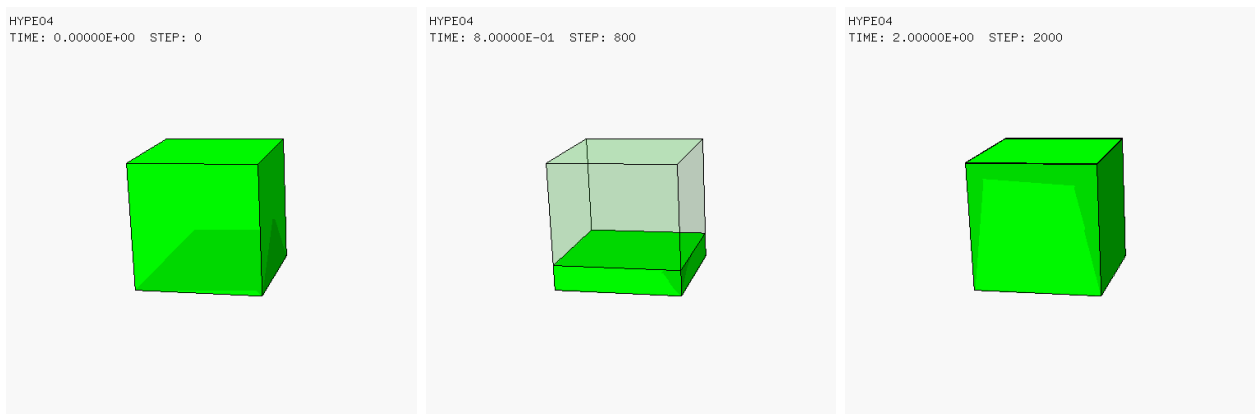


(c) Energy potential



(d) Maximum time step

Figure 29: Further results of test case hype03.



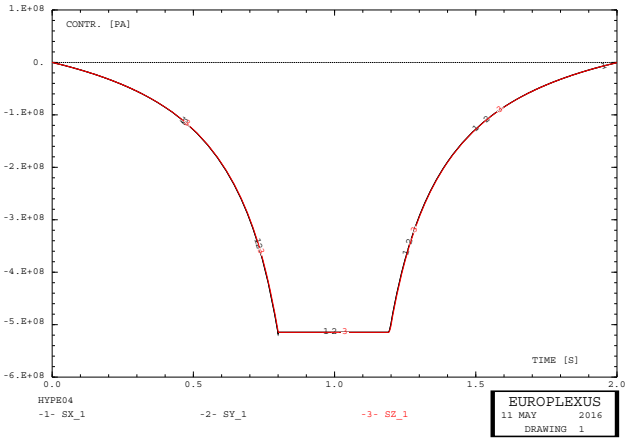
(a)  $t = 0$

(b)  $t = 0.80$  s

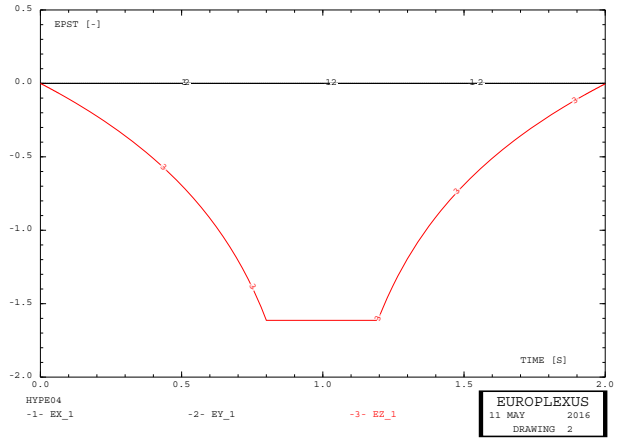
(c)  $t = 2.00$  s

Figure 30: Initial and deformed configurations in test case hype04.

confirms that the material is infinitely compressible (not infinitely incompressible), coherently with the chosen value  $K = 0$ . The stability time step (Figure 34(d)) follows a weird pattern, which will have to be explained.

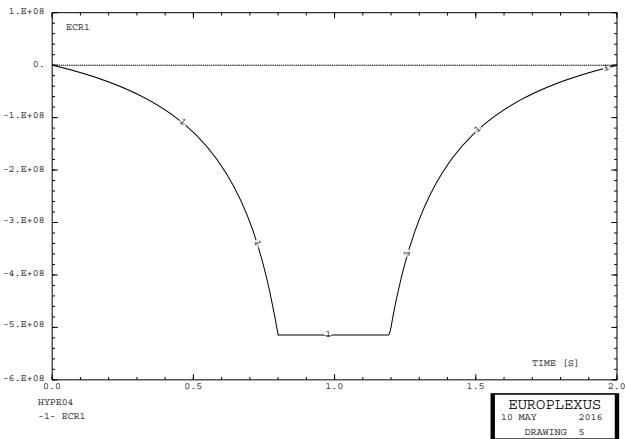


(a) Normal stresses

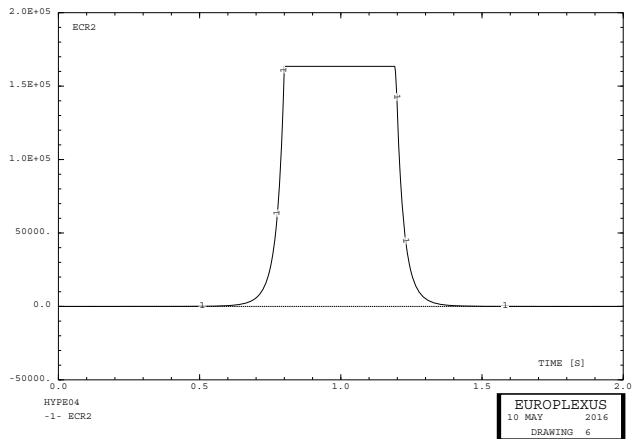


(b) Normal strains

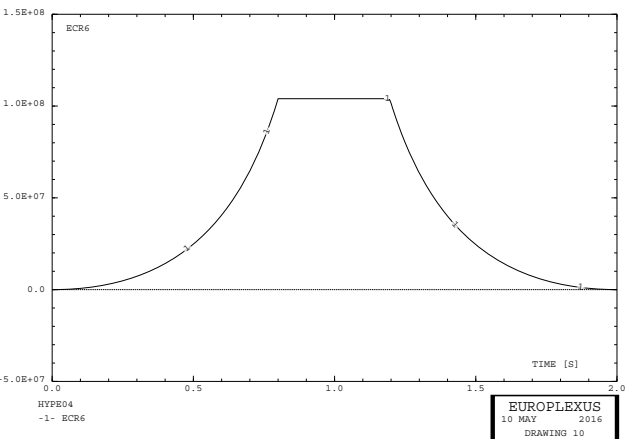
Figure 31: Normal stresses and normal strains in test case hype04.



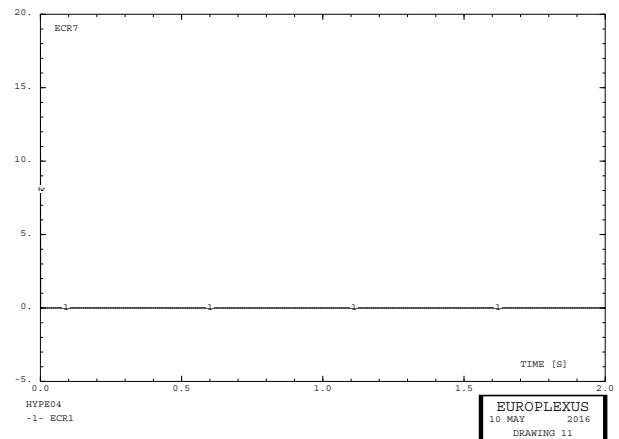
(a) Pressure



(b) von Mises stress



(c) Energy potential

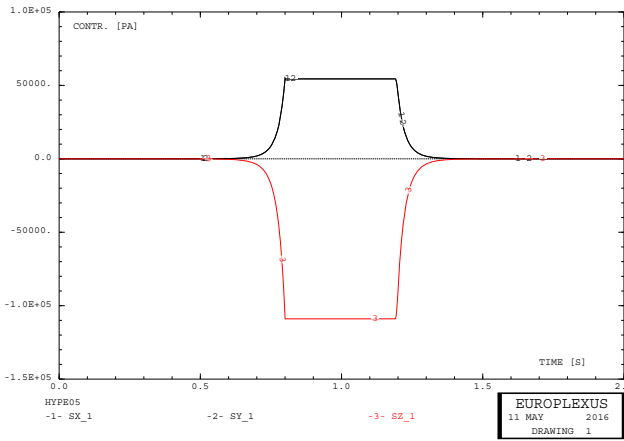


(d) Maximum time step

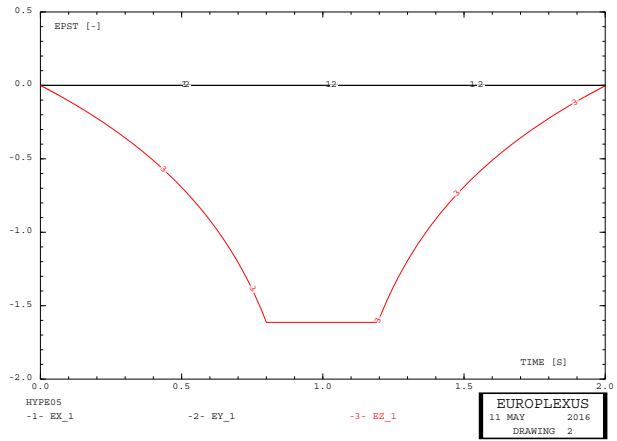
Figure 32: Further results of test case hype04.

### 3.6 Actual 3D simulations with the second volumetric mesh

The preliminary simulations of the rat head performed so far have shown that the major bottleneck is the use of a water-like (FLUI) material to model the CSF, because this material is much (orders of magnitude) stiffer than the brain material and thus penalizes the stability step for the entire

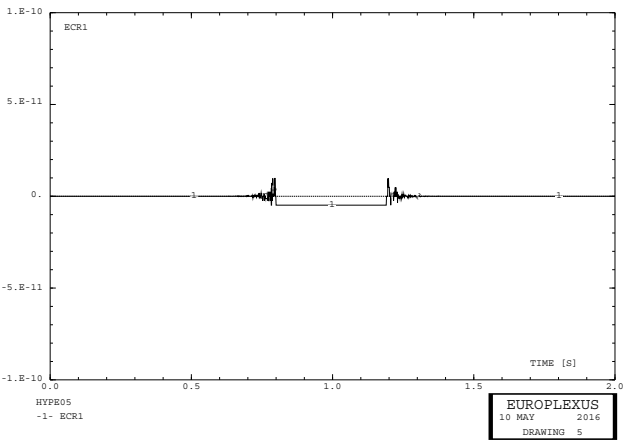


(a) Normal stresses

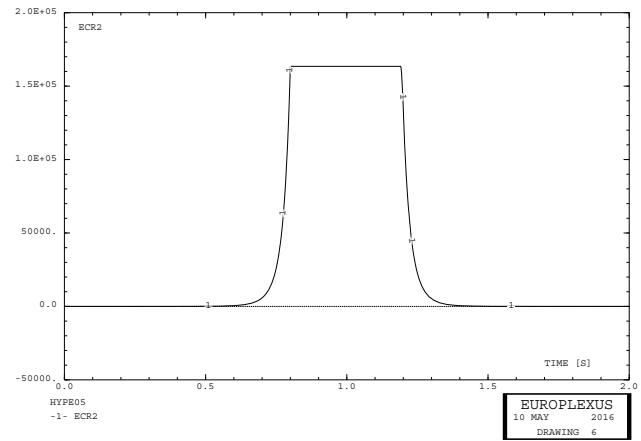


(b) Normal strains

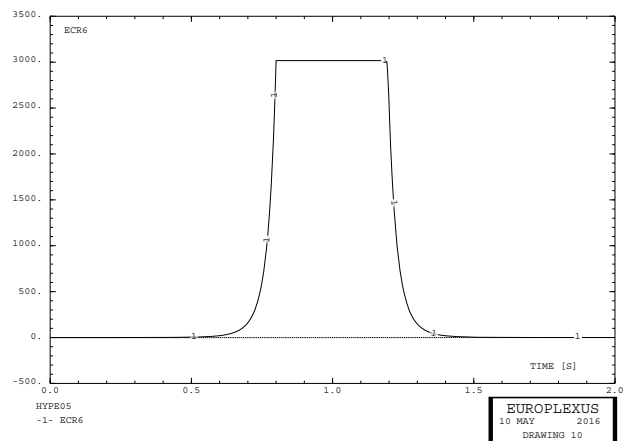
Figure 33: Normal stresses and normal strains in test case hype05.



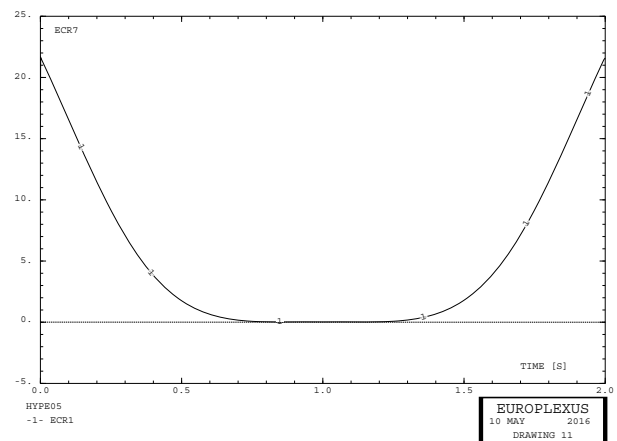
(a) Pressure



(b) von Mises stress



(c) Energy potential



(d) Maximum time step

Figure 34: Further results of test case hype05.

computation.

From the literature not much is known about the CSF material. It might have a much softer behaviour than almost-incompressible pure water, since it contains impurities. Also, some thin filaments are present in the cavity between the skull and the brain. Therefore the overall behaviour of

the CSF region might be much softer than assumed so far.

Several authors seem to adopt a material behaviour similar to that of the brain (but with different parameters) also in the CSF, i.e. they treat also the CSF as a (solid) hyperelastic material.

This has direct implications on the modelling. The ALE technique can no longer be used to represent the CSF so that the complete model becomes Lagrangian. Furthermore, there is no reason to keep the CSF nodes distinct from those of the (external surface of the) brain, and of the skull. Such nodes can therefore be merged at the mesh generator level. Moreover, since the skull is assumed to behave like a rigid body, it is no longer necessary to model it explicitly: the imposed motion can be applied directly to the external nodes of the (now solid) CSF.

A Cast3m procedure for producing such a mesh is prepared, see file `genmesh1.dgibi` in Appendix (the final 1 stays for Lagrangian). This writes the complete head mesh (now containing just the brain and the CSF) to a Cast3m mesh file called `head1.msh`.

### 3.6.1 Case brat09

A first attempt of using a hyperelastic material also for the CSF is shown in test case `brat09`. Two important changes are made on the chosen material parameters:

- For the brain, the Ogden set 1 parameters are used but the bulk modulus is reduced from  $K = 128 \times 10^6$  to as low as  $K = 10 \times 10^3$  (a reduction of four orders of magnitude).
- For the CSF, the same material as for the brain is used (with the new, low bulk modulus) but the  $\mu$  parameters (coefficients `C05` and `C06` in this case) are reduced by a factor 10. This should make the CSF material 10 times *softer* than the brain material.

These values of the material data are being used at UPC to perform preliminary simulations of the rat brain with the Abaqus<sup>®</sup> explicit code. The same values must be used in EPX in order to compare results from the two codes.

The material data for this case are as follows:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new) "Set 1" from P. Saez
                                ! (with reduced BULK)
                                BULK 10.E+03 ! This is in Pa
                                C01 -10.8    ! This is alpha_1
                                C02  19.7    ! This is alpha_2
                                C05  -0.29   ! This is mu_1
                                C06  9.6E-3  ! This is mu_2
                                LECT brain TERM
                                HYPE TYPE 4 RO 1040      ! Ogden (new) modified "Set 1"
                                                                ! from P. Saez for CSF
                                                                BULK 10.E+03 ! This is in Pa
                                                                C01 -10.8    ! This is alpha_1
                                                                C02  19.7    ! This is alpha_2
                                                                C05  -0.029  ! This is mu_1_brain / 10
                                                                C06  9.6E-4  ! This is mu_2_brain / 10
                                                                LECT csf TERM'
```

With the new materials the stability step changes dramatically with respect to the previous simulations. At step 0 EPX estimates a stability step of  $8.60 \times 10^{-5}$  s. This value is not reliable, as already noted previously, because now the model includes only HYPE materials. The reported stability drops to  $1.05 \times 10^{-6}$  at step 1 and reaches values of  $2.55 \times 10^{-6}$  some steps later.

In order to avoid stability problems, we impose an initial step value of  $1.0 \times 10^{-8}$  by means of the `PAS1` command of EPX. Since by default the code at most doubles the time increment from a step to the next one, this ensures a few initial steps with values certainly below the stability limit, until the stability estimation done by EPX becomes (hopefully) more reliable.

The technique worked well and the calculation proceeded without problems until it was stopped at step 8321 and time 29.0 ms, having used 22 283 s (6.2 hours) of CPU time.

The computed longitudinal stress in the head ( $\sigma_{yy}$ , where  $y$  is the direction of the initial velocity which is gradually reduced) is shown in Figure 35. recall that the imposed time evolution of the skull velocity is the one shown in Figure 17, i.e. the velocity is reduced from the initial 0.1 m/s to rest in



25 ms, then the skull is kept at rest indefinitely. Figure 36 shows the same four first results by using iso-surface representation which allows to see *inside* the head stress field instead of looking only at values on the head's external surface (i.e. the CSF's external surface).

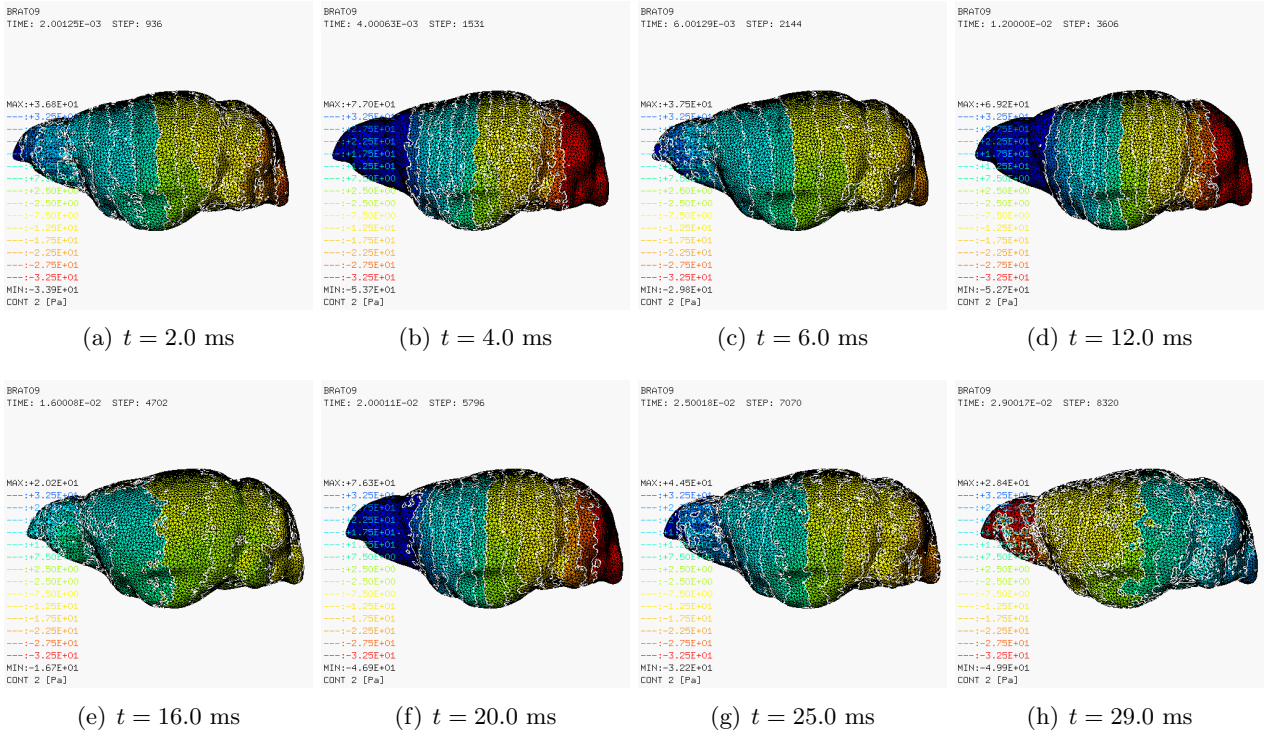


Figure 35: Stress  $\sigma_{yy}$  in the rat head in case `brat09`.

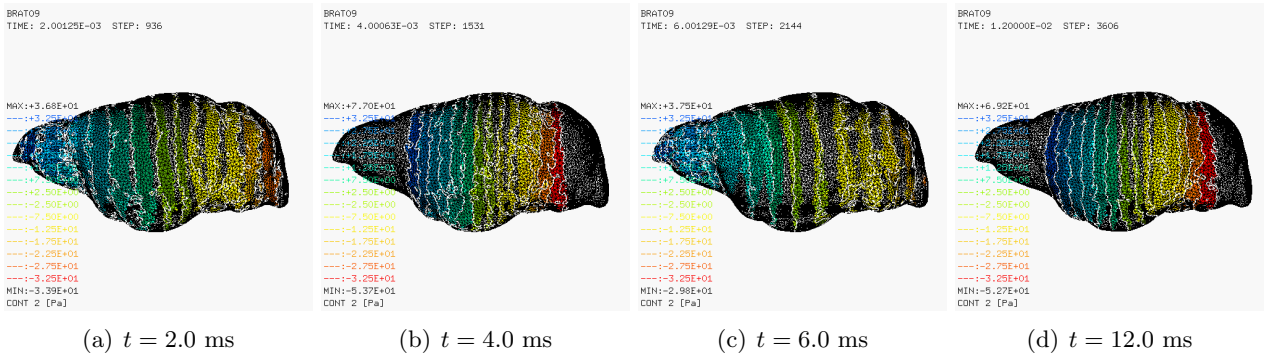


Figure 36: Iso-surfaces of stress  $\sigma_{yy}$  in the rat head in case `brat09`.

It should be noted that, due to a mis-understanding of the rat's head anatomy, the frontal and occipital regions were confused so that the initial velocity assumed here corresponds to a rat moving **backwards**, which is not what was intended. However, this is probably irrelevant given the preliminary nature of these simulations. The same misunderstanding affected also all other simulations in the present report.

In order to interpret the results in Figures 35 and 36 it should be kept in mind that the colors have been inverted with respect to a normal representation of iso-values, so that the full red color corresponds to the highest negative (i.e. **compression**) stress, the green color corresponds to a stress-free situation and the dark blue color corresponds to the highest positive (i.e. **traction**) stress.

One sees a series of qualitatively intuitive phenomena occurring in the head:

- As deceleration (towards the left) begins, compression starts to build up in the right part of the head (occipital region here) and a corresponding traction zone builds up at the opposite end.

- A sort of pulsating stress pattern (very evident from the results animations) is established which continues as long as the (constant) deceleration is applied. In the occipital region, the stress oscillates between zero and a maximum compression value while in the frontal region the opposite occurs i.e. the stress oscillates between zero and a maximum traction value. The central part of the head remains relatively stress-free during this phase.
- As the head stops completely and is kept blocked (no *counter-coup* is modelled here, although this would likely occur in practice), the soft brain and CSF matter starts oscillating within the supposedly rigid (and still) skull. As the brain matter bounces back a compression stress (red zone) is for the first time observed at  $t = 29$  ms in the frontal region, see Figure 35(h). A longer continuation of the calculation would allow to better appreciate this phenomenon (see subsequent simulations).

From Figure 36 one can see that the state of stress in the head at any given time is at least approximately *one-dimensional*, in the sense that  $\sigma \approx \sigma(y)$ , where  $y$  is the direction of the deceleration. In fact, the iso-surfaces of stress are in good approximation planes perpendicular to the  $y$  axis. This fact is very important because it could open the possibility of a reduced-model representation of the head, at least for some initial simulations, that is using a *2D slice* of the head instead of a full 3D model. This would of course dramatically reduce the cost of the simulations.

To conclude the presentation of this test case, Figure 37 shows the variation in time of the displacement, velocity and stress (in the direction of deceleration) at a point of the rat head near the centroid of the model (i.e. approximately in the middle of the brain). Note that, for a point located in the central part of the head as the one chosen here, the maximum stress occurs during the holding phase and not during the deceleration phase, as already discussed above.

### 3.6.2 Cases `brat10` and `brat11`

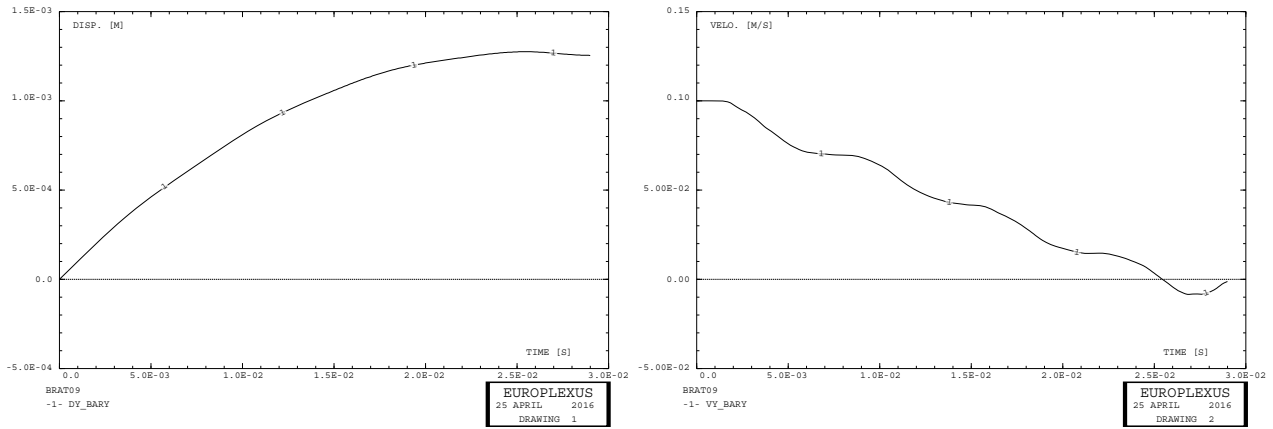
The use of a solid (though very soft) material in the CSF region opens the way to a technique of speeding up the computation based upon element erosion. The idea is to *erode* (i.e. to remove from the computation) the most critical elements, i.e. most probably the smallest ones in the model. Such elements may result from the difficulty of automatically meshing such a complex object as a brain. The erosion introduces a sort of small *holes* in the head mesh. Of course, it is essential that the presence of such holes does not appreciably modify the results, while at the same time speeding up the overall calculation

In order to explore this possibility, test case `brat10` is a repetition of case `brat09` discussed in the previous Section, but by adding erosion. The keyword `TFAI 4.E-4` activates erosion of all elements in the (initial) mesh whose critical step is below the chosen value (0.4 ms). The scope here is not to run the transient calculation, but simply to obtain the list of such elements from the EPX listing. In this case, 178 elements fall within the chosen threshold and are eroded.

Next, another calculation is prepared, case `brat11`. A new group of elements (`eros0` object) is added in the input file, which includes all the elements extracted from the previous test. Then, all elements belonging to the `eros0` group are assigned a dummy (`FANT`) material. This is equivalent to removing them from the calculation, i.e. it has the same effect as erosion. The advantage is that in this way erosion can be completely removed from the calculation, since we are not interested in eroding any elements during the transient simulation. The location of the dummy-material elements is shown in Figure 38.

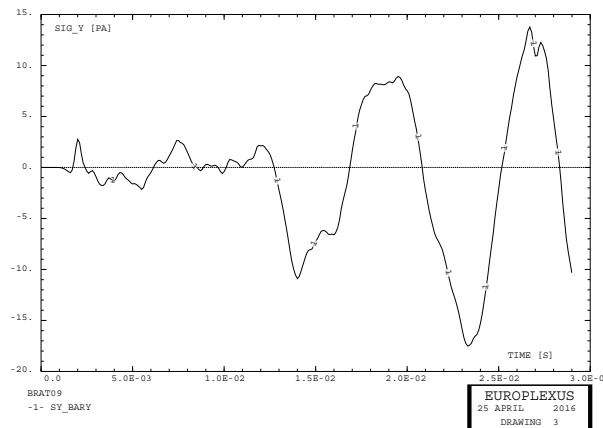
The calculation `brat11` was stopped at  $t = 28$  ms (i.e. 1 ms earlier than case `brat09`), after 7714 time steps and 20390 s (5.7 hours) of CPU time. The speed-up with respect to case `brat09` was of about 5 %, which is a modest amount for the effort (at least in this particular case).

Comparison of results shows that the two solutions without and with dummy material in the smallest elements are very close, see e.g. Figure 39 which compares the stress distributions at 20 ms, or Figure 40 which compares the stress *vs.* time at the head centroid.



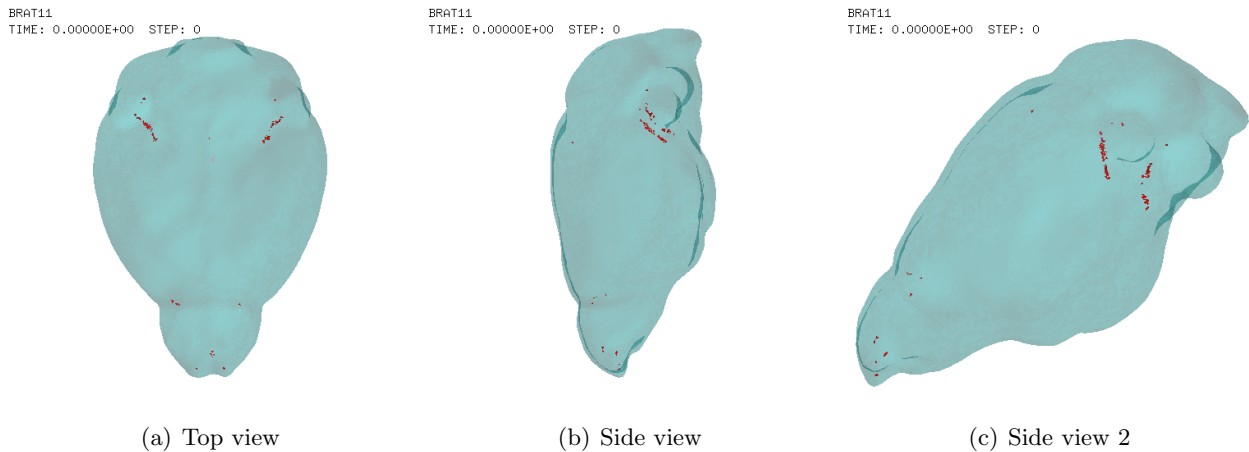
(a) Displacement

(b) Velocity



(c) Stress

Figure 37: Results at the centroid of the rat head in case **brat09**.



(a) Top view

(b) Side view

(c) Side view 2

Figure 38: Location of dummy material elements in case **brat11**.

### 3.6.3 Cases **brat12** and **brat13**

The next calculation, **brat12**, is similar to case **brat09** but uses different parameters in the **HYPE** material for the CSF. The bulk modulus of the CSF is increased by a factor 100 and the  $\mu$  parameters are reduced by a factor 10. The material for the brain and the velocity profile stay unchanged.

Thus in this simulation the brain and the CSF use the same **HYPE** material but with different bulk modulus ( $K = 1.0 \times 10^4$  in the brain and  $K = 1.0 \times 10^6$  in the CSF), and different  $\mu$  parameters (the

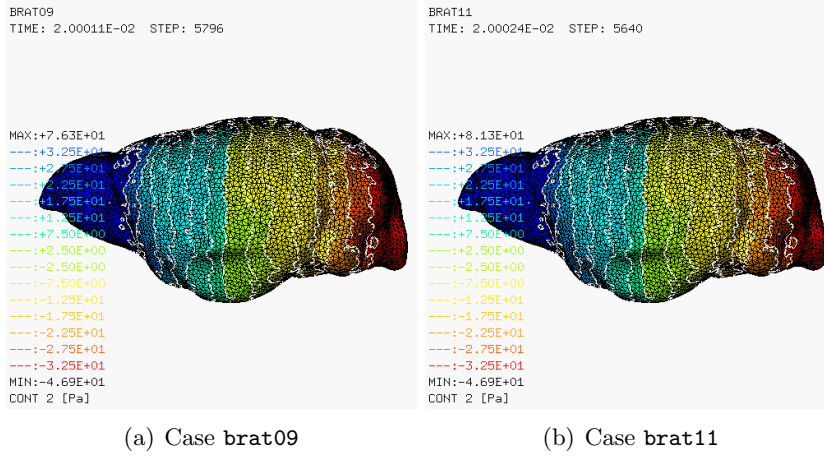


Figure 39: Comparison of stress distributions in the head at 20 ms in cases **brat09** and **brat11**.

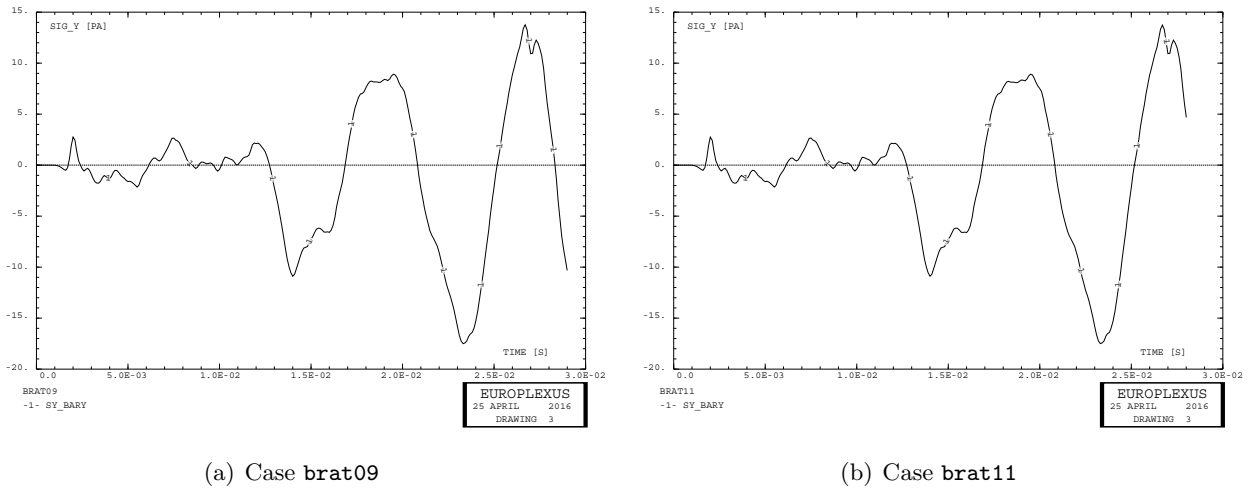


Figure 40: Comparison of stress at the head centroid in cases **brat09** and **brat11**.

$\mu$  in the CSF being now  $1/100$  of the  $\mu$  in the brain).

The calculation was stopped at  $t = 11.5$  ms, reached in 22 318 time steps and 57 471 s (16.0 hours) of CPU time. The initial estimated stability was  $8.6 \times 10^{-5}$ , dropping to  $2.7 \times 10^{-7}$  after just a few steps and the final step before interruption was  $4.13 \times 10^{-7}$ . The average step in the computed interval was  $5.1 \times 10^{-7}$  s, thus much lower than in simulation **brat09**, due to higher stiffness of the CSF material in the present case.

Some results are presented in Figure 41 which shows the stress  $\sigma_y$  in the brain seen from the top. The CSF is cross-sectioned in order to show its position but the stress is only visualized in the brain (therefore the CSF appears in grey). Hardly any motion of the brain with respect to the CSF (which gives the impression of being rather stiff) can be observed.

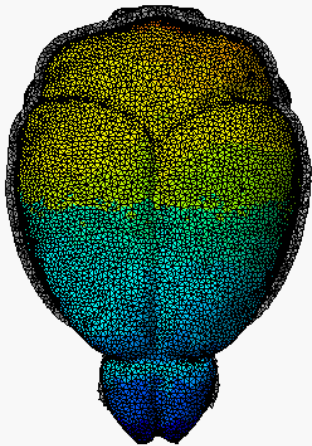
The next calculation (**brat13**) uses exactly the same materials as in case **brat12** but a 100 times larger initial velocity (10 m/s instead of 0.1 m/s), so that the deceleration (and the generated inertia force) is 100 times larger. The corresponding time function is shown in Figure 42.

This calculation stopped at  $t = 3.70$  ms after 5746 time steps and 15 364 s (4.3 hours) of CPU time. One of the elements (element 19 332, located in the brain) became too distorted and the stability step became too small for the calculation to continue.

Figure 43 shows the stress  $\sigma_y$  in the brain seen from the top. The CSF is cross-sectioned in order to show its position but the stress is only visualized in the brain (therefore the CSF appears in grey). In this case a very large motion of the brain with respect to the CSF is observed (or more precisely a very large deformation of the brain, since the CSF appears as comparatively much stiffer than the

BRAT12  
 TIME: 2.00044E-03 STEP: 3235

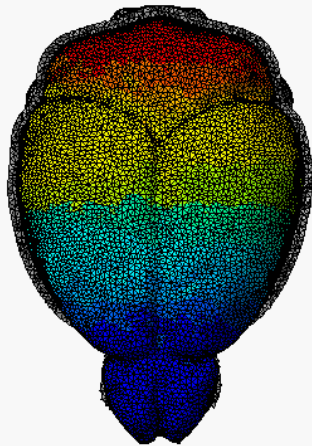
MAX:+4.24E+01  
 ---:+3.25E+01  
 ---:+2.75E+01  
 ---:+2.25E+01  
 ---:+1.75E+01  
 ---:+1.25E+01  
 ---:+7.50E+00  
 ---:+2.50E+00  
 ---:-2.50E+00  
 ---:-7.50E+00  
 ---:-1.25E+01  
 ---:-1.75E+01  
 ---:-2.25E+01  
 ---:-2.75E+01  
 ---:-3.25E+01  
 MIN:-3.32E+01  
 CONT 2 [Pa]



(a)  $t = 2.0$  ms

BRAT12  
 TIME: 4.00034E-03 STEP: 7357

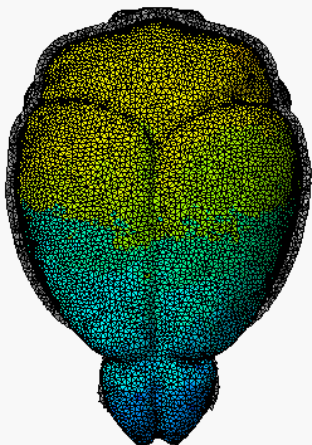
MAX:+7.59E+01  
 ---:+3.25E+01  
 ---:+2.75E+01  
 ---:+2.25E+01  
 ---:+1.75E+01  
 ---:+1.25E+01  
 ---:+7.50E+00  
 ---:+2.50E+00  
 ---:-2.50E+00  
 ---:-7.50E+00  
 ---:-1.25E+01  
 ---:-1.75E+01  
 ---:-2.25E+01  
 ---:-2.75E+01  
 ---:-3.25E+01  
 MIN:-4.91E+01  
 CONT 2 [Pa]



(b)  $t = 4.0$  ms

BRAT12  
 TIME: 6.00044E-03 STEP: 11282

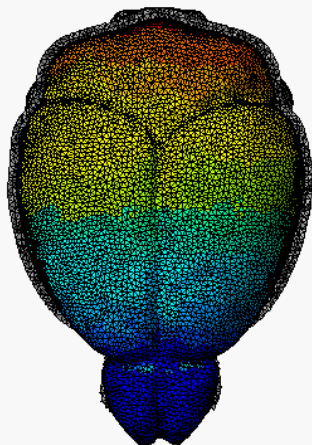
MAX:+3.40E+01  
 ---:+3.25E+01  
 ---:+2.75E+01  
 ---:+2.25E+01  
 ---:+1.75E+01  
 ---:+1.25E+01  
 ---:+7.50E+00  
 ---:+2.50E+00  
 ---:-2.50E+00  
 ---:-7.50E+00  
 ---:-1.25E+01  
 ---:-1.75E+01  
 ---:-2.25E+01  
 ---:-2.75E+01  
 ---:-3.25E+01  
 MIN:-2.72E+01  
 CONT 2 [Pa]



(c)  $t = 6.0$  ms

BRAT12  
 TIME: 1.10004E-02 STEP: 21062

MAX:+6.62E+01  
 ---:+3.25E+01  
 ---:+2.75E+01  
 ---:+2.25E+01  
 ---:+1.75E+01  
 ---:+1.25E+01  
 ---:+7.50E+00  
 ---:+2.50E+00  
 ---:-2.50E+00  
 ---:-7.50E+00  
 ---:-1.25E+01  
 ---:-1.75E+01  
 ---:-2.25E+01  
 ---:-2.75E+01  
 ---:-3.25E+01  
 MIN:-4.23E+01  
 CONT 2 [Pa]



(d)  $t = 11.0$  ms

Figure 41: Distribution of stress  $\sigma_{yy}$  in the rat brain in case brat12.

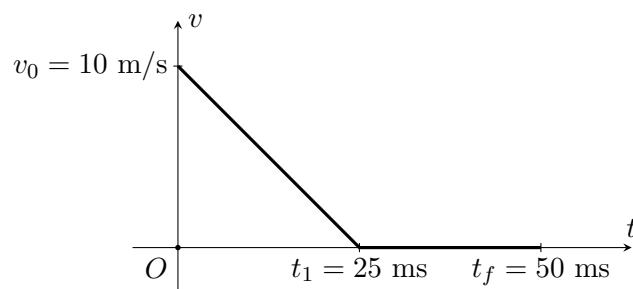


Figure 42: Time evolution of the skull velocity in case brat13.



brain), and this is probably the reason for the element entanglement.

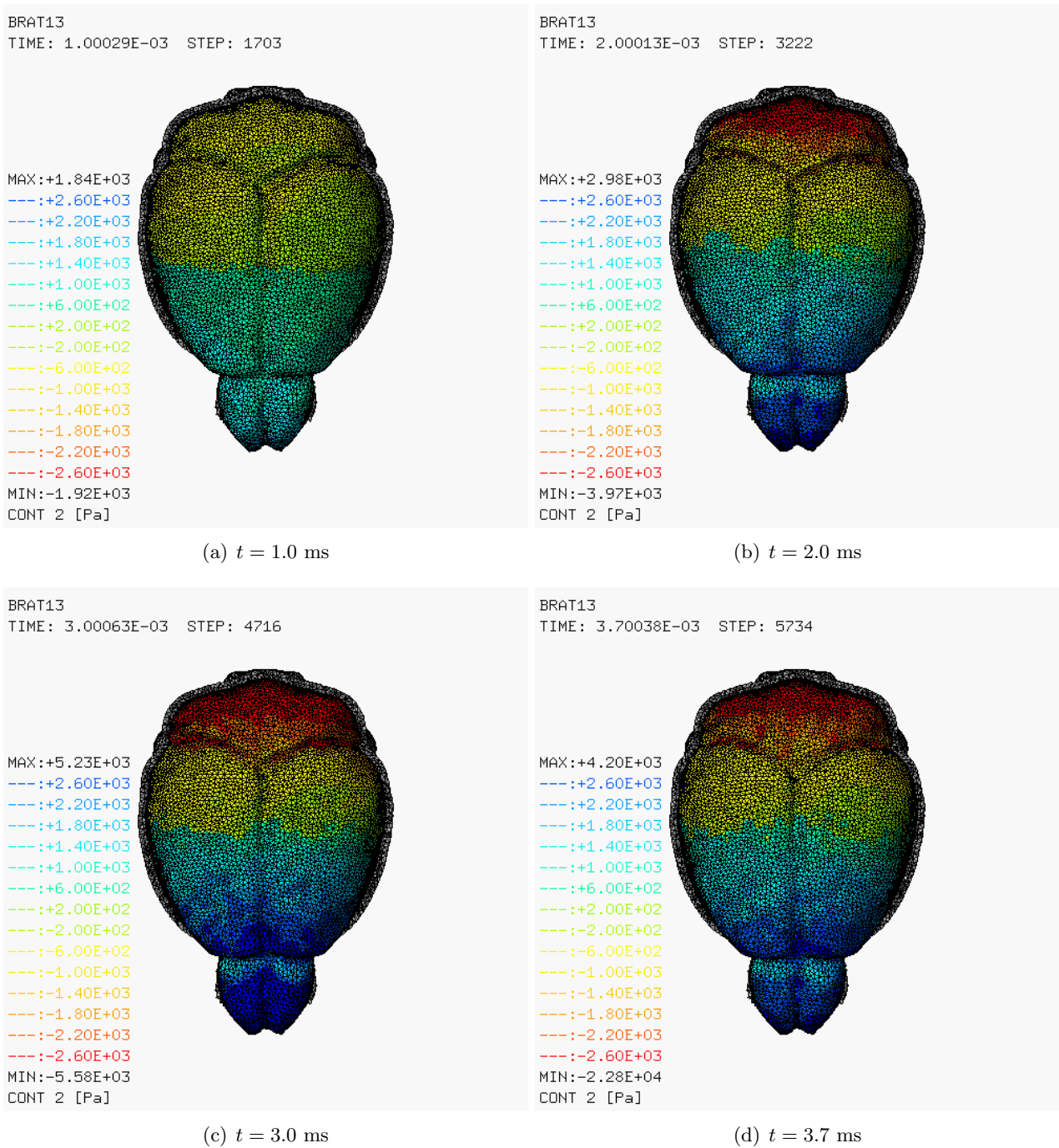


Figure 43: Distribution of stress  $\sigma_{yy}$  in the rat brain in case **brat13**.

### 3.7 Head slice simulations

Based upon the findings of full 3D simulation **brat09**, an attempt is made to simulate only one slice of the head. This should substantially reduce the computation cost and will allow to do many parametric simulations and model calibrations (in particular of the material properties) before continuing with more detailed full 3D analyses.

### 3.7.1 Circular 2D slice testing in case br2d01

We start by testing a 2D slice of the simplified spherical head geometry used in reference [3]. A 2D (circular) mesh is built containing the brain and the CSF and made of 1920 CAR4 4-node quadrilaterals. A Lagrangian, plane strain analysis is performed. This simulates a prismatic body of infinite length normal to the slice plane.

The initial velocity is  $v_0 = 0.1$  m/s and the deceleration shown in Figure 17 is imposed to all nodes on the external contour of the CSF. The materials used are the same as in case `brat12` of the previous Section.

Results of this calculation in terms of the stress  $\sigma_{xx}$  (the deceleration in this case acts along the  $x$  axis) are shown in Figure 44. These results are obviously wrong because the symmetry of the problem is not respected. Since this is the first time that this combination of element (2D CAR4 in plane strain) and material (HYPE) is used, a simpler one-element test is performed in the next Section to check the code.

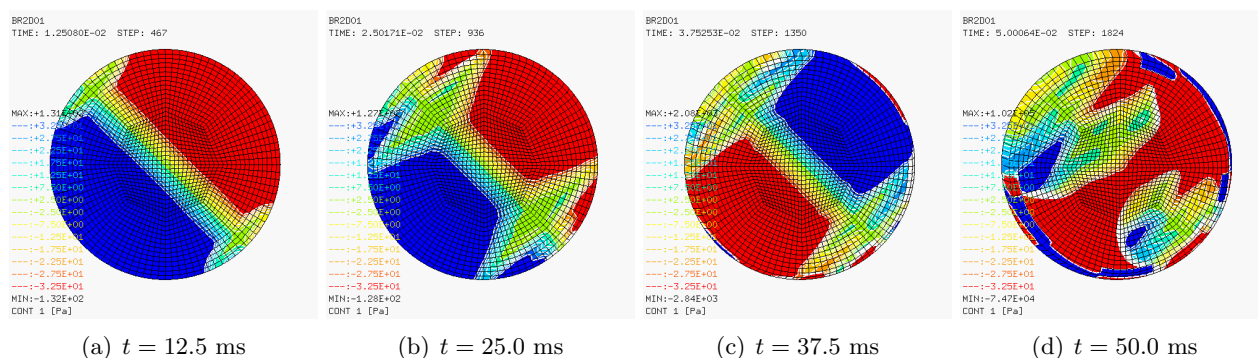


Figure 44: Distribution of stress  $\sigma_{xx}$  in a circular 2D slice in case `br2d01`.

### 3.7.2 Malfunctioning of the HYPE material in 2D, case br2d00

A single CAR4 quadrilateral in plane strain and using the same HYPE material as for the brain in the previous `br2d01` calculation is subjected to an imposed compression along the vertical direction, similar to what has been done in 3D in case `hype03` in a previous Section.

The results of this test are not convincing and are not shown for brevity. The 2D version of the HYPE material routine (at least) needs to be revised and is not usable in its present state in EPX.

### 3.7.3 Circular 3D slice testing in cases br3d01 and br3d02

Since it is not possible at the moment to obtain results with the HYPE material in 2D calculations, as shown in the previous Section, a 3D version of the (circular) slice problem is built up, by extruding the 2D mesh previously built along the  $z$  axis.

The mesh for the first calculation (`br3d01`) consists of 1920 elements of type CUBE. The advantage is that the 3D version of the constitutive routine can be used with this mesh. In order to simulate a 2D plane strain solution, we simply block the motion of all nodes along the  $z$  direction.

Results are shown in Figure 45 now showing a much more plausible development of stress in the brain region (the CSF appears in gray since the stress in it is not shown). Some probably parasitic oscillations of the solution near the interface between the two hyperelastic materials are observed.

The next model (`br3d02`) is obtained from the previous one by splitting each 8-node cube element into two 6-node prisms. Therefore, this model contains  $2 \times 1920 = 3840$  elements of type PRIS.

Results are shown in Figure 46 which is in good agreement with the solution obtained with cubes in the bulk brain material. Furthermore, no oscillations of the solution near the interface between the two hyperelastic materials are present.

From these results, it seems possible to investigate many of the head modelling aspects, in particular the material parameters to be used, in 3D slices of the head consisting of just one element through

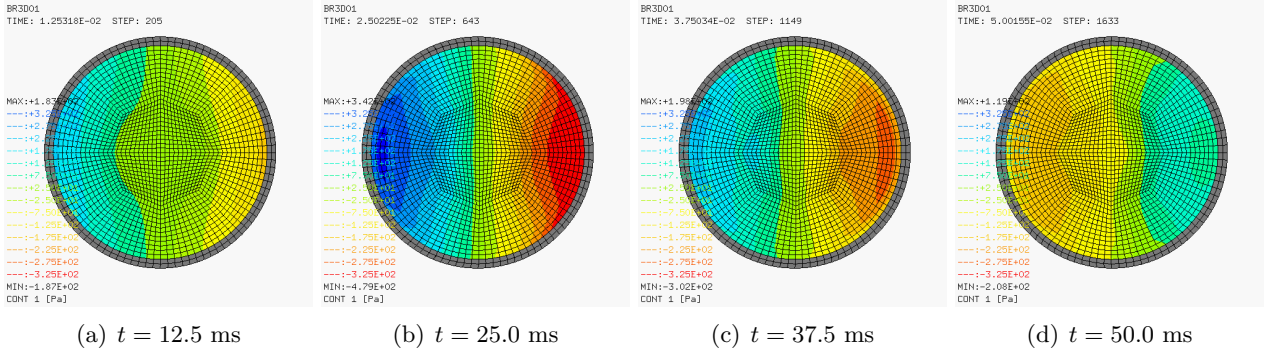


Figure 45: Distribution of stress  $\sigma_{xx}$  in a circular 3D slice (hexahedra) in case br3d01.

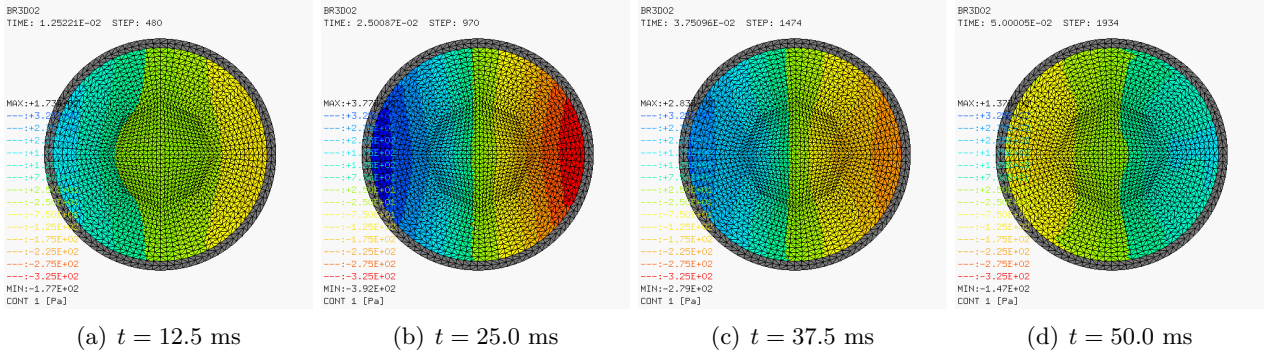


Figure 46: Distribution of stress  $\sigma_{xx}$  in a circular 3D slice (prisms) in case br3d02.

the thickness. This dramatically reduces the computer time. For example, the calculations br3d01 and br3d02 took only 1633 steps and 1934 steps, respectively, to reach the chosen final time of 50 ms. The needed CPU time was 27 and 47 s, respectively (however, the mesh was very coarse).

### 3.7.4 Full 3D slice testing in cases brat21 and brat22

The slicing technique tested in the previous Sections is now tentatively applied to a more realistic geometry than the circular slices considered so far. A slice of the full 3D model of the rat head can be obtained in basically two different ways. The first one is to import the complete 3D model in EPX and then to isolate a slice by using the simple mesh manipulating commands available in the code. All elements not belonging to the slice are assigned a dummy material, thus removing them from the calculation. A second possibility is, of course, to slice the 3D mesh in advance by the mesh generator (Cast3m here) and then to import just the slice in EPX. This second solution is a bit more laborious, but also more efficient. Both techniques will be tentatively explored below.

In the first simulation, case brat21, we import in EPX the full 3D mesh. The input is equivalent to the full 3D case brat12 as concerns the materials, time function etc. The part of the EPX input file dealing with the slicing reads:

```

GEOM SCAL FACT 0.001
  TETR brain csf TERM
COMP GROU 3 'sbrain' LECT brain TERM COND XB GT -6.25E-3
                                COND XB LT -5.75E-3
      'scsf' LECT csf TERM COND XB GT -6.25E-3
                                COND XB LT -5.75E-3
      'shead' LECT sbrain scsf TERM
NGRO 1 'spskull' LECT pskull TERM COND X GT -6.25E-3
                                COND X LT -5.75E-3

COUL GR50 LECT sbrain TERM
TURQ LECT scsf TERM

```

We have used EPX's COMP GROU directive to extract from the mesh the desired slice via the COND sub-directive. Here all elements (tetrahedra) whose centroid falls within the interval  $-6.25 \times 10^{-3} \leq$



$x \leq -5.75 \times 10^{-3}$  are selected. Two slice objects, one for the brain (**sbrain**) and one for the CSF (**scsf**) are extracted, and then grouped together to form the desired head slice (**shead**). Then a dummy material (**FANT**) is assigned to all elements *not* belonging to the slice:

```
MATE HYPE TYPE 4 RO 1040      ! Ogden (new) "Set 1" from P. Saez
                                ! (with reduced BULK)
      BULK 10.E+03 ! This is in Pa
      C01 -10.8   ! This is alpha_1
      C02  19.7   ! This is alpha_2
      C05  -0.29  ! This is mu_1
      C06   9.6E-3 ! This is mu_2
    LECT sbrain TERM
  HYPE TYPE 4 RO 1040      ! Ogden (new) modified "Set 1"
                                ! from P. Saez for CSF (2)
      BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
      C01 -10.8   ! This is alpha_1
      C02  19.7   ! This is alpha_2
      C05  -0.0029 ! This is mu_1_brain / 100
      C06   9.6E-5 ! This is mu_2_brain / 100
    LECT scsf TERM
  FANT 1.E-6 LECT head DIFF sbrain scsf TERM
```

In this way, only 28 230 elements are really active in the calculation, out of a total number of 458 501 elements. In this way, the solution proceeds much faster. However, a drawback of this technique is that the general ALIC result files are as big as in the full 3D case, since all elements (including the dummy ones) are always contained in this type of result files.

Note that the above slicing process produces an *irregular* slice, because an element (a tetrahedron in this case) is retained or not in the slice, but elements are never sectioned since this would produce a completely different mesh, possibly including also some very small elements. Furthermore, the slice contains (in average) more than one element (at least two or three) through the thickness, due to the irregularity of the full 3D starting mesh made of tetrahedra.

In order to simulate a 2D plane strain solution, the motion of all nodes in the slice (both the internal ones and those on the surface of the slice) is blocked in the  $x$  direction (which is normal to the slicing plane):

```
LINK COUP SPLT NONE
  BLOQ 13 LECT spskull TERM
  BLOQ  1 LECT shead  TERM
  VITE 2 0.1 FONC 1 LECT spskull TERM
```

This calculation arrived at the final time of 50 ms in 47 224 time steps, using 8986 s (2.5 hours) of CPU time. This computation time is much shorter than the one needed by the full 3D model (case **brat12**) which, by the way, stopped at about 11 ms due to excessive deformation of an element.

The computed stress  $\sigma_{yy}$  in the brain is shown in Figure 47. The CSF is also shown (in grey) in order to appreciate the relative motion of the two materials. The first four pictures of Figure 47 are taken at the same times as those for the full 3D solution **brat12** and can be compared with Figure 41.

Test **brat22** was a repetition of **brat21** by activating the *partial* instead of *full* ALIC results file (see the ALIC TEMP directive), in an attempt to store only the slice elements and thus reduce the amount of output data (which is always very large in 3D computations).

However, EPX refuses to produce a *splitted* ALIC TEMP file so that all results (at all time stations) would end up in a single results file. The size of such a file (although containing only part of the elements) would be too big for practical post-processing.

### 3.7.5 Full 3D slice testing in case brat23

This calculation (**brat23**) is similar to case **brat21** discussed above, but only the slice elements are imported in EPX. The 3D mesh (from file **head1.msh**) is preliminarily sliced in Cast3m by using a Gibiane command file (see file **slice.dgibi** in Appendix). Only the resulting slice geometry is saved in file **slice.msh** and is then imported in EPX.

Note that the slice mesh produced in this way may have some minor differences with respect to the (nominally equivalent) slice obtained in EPX with the first technique, due to the different way in

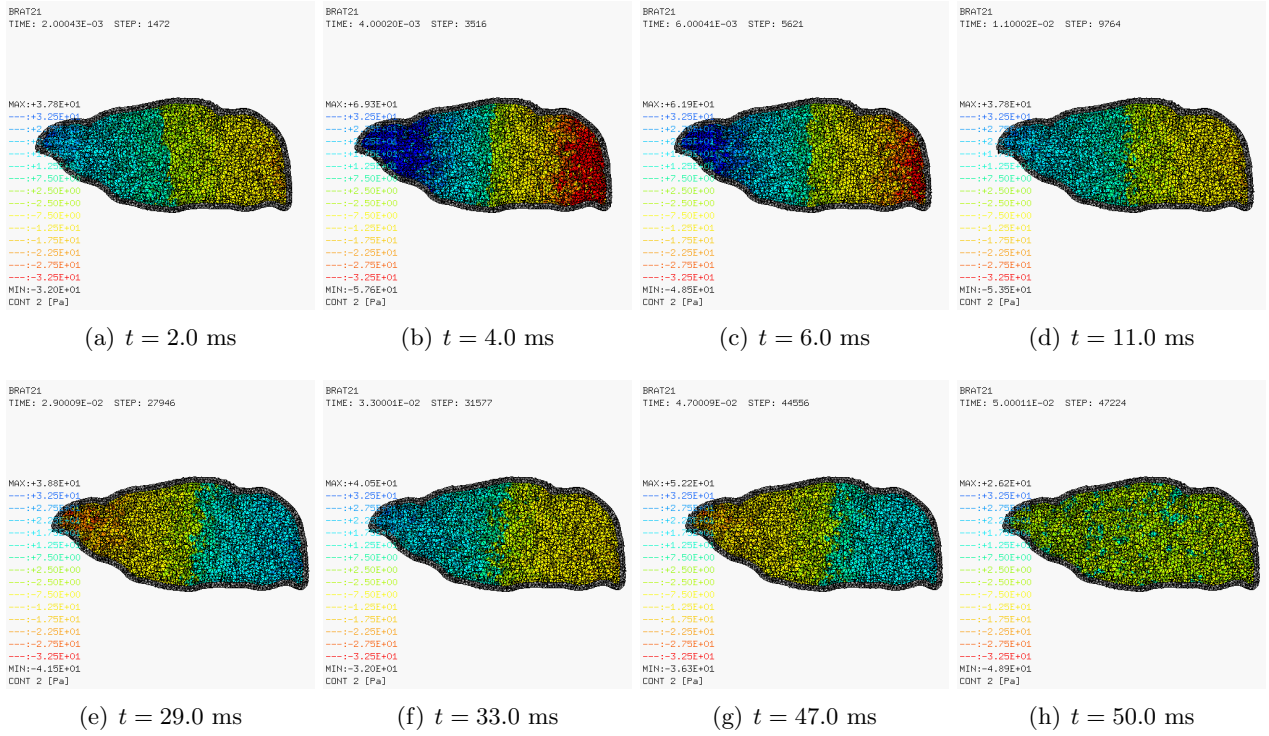


Figure 47: Distribution of stress  $\sigma_{yy}$  in a 3D slice (tetrahedra) in case brat21.

which the slicing is performed. In EPX this is based on an element's *centroid*, while in Cast3m an element is retained, if and only if all its *nodes* are contained within the slice.

The computed stress  $\sigma_{yy}$  in the brain is shown in Figure 48, which should be compared with Figure 47 obtained with the other technique. The agreement is quite good, considering the above mentioned differences in the two meshes.

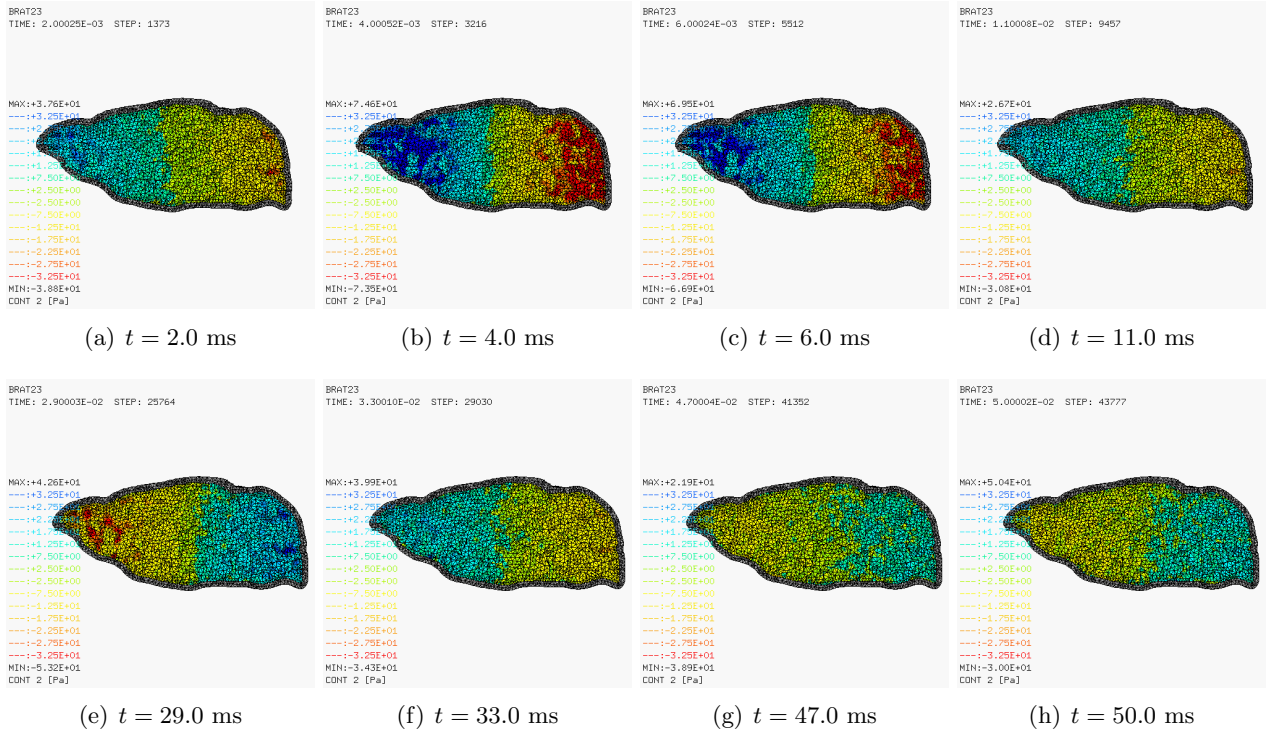


Figure 48: Distribution of stress  $\sigma_{yy}$  in a 3D slice (tetrahedra) in case brat23.

The calculation `brat23` arrived at the final time of 50 ms in 43 777 time steps and 5121 s (1.4 hours) of CPU time. Therefore, the solution time was almost halved with respect to the first technique. This model contained 21 299 elements, to be compared with the 28 230 (active) elements in case `brat21`.

These results seem to indicate that, at least for the rat head considered here, the use of head slice meshes is a convenient and relatively accurate technique in order to perform head model calibrations (parametric studies), before launching much larger and CPU-time consuming full 3D model simulations.

## References

- [1] EUROPLEXUS User's Manual, on-line version: <http://europlexus.jrc.ec.europa.eu>.
- [2] Cast3m Software: <http://www-cast3m.cea.fr/>.
- [3] F. Casadei, P. Saez, P. Díez, M. Larcher, G. Valsamos. Scoping calculations for the simulation of human head impact with EUROPLEXUS. JRC Technical Report, in publication.
- [4] PointWise Software: <http://www.pointwise.com/>.
- [5] Scalable brain atlas: <https://scalablebrainatlas.incf.org/main/index.php>.

## Appendix — Input files

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

### br2d00.epx

```
BR2D00
ECHO
!CONV WIN
DPLA LAGR
GEOM LIBR POIN 4 CAR4 1 TERM
  0 0 1 0 0 1 1 1
  1 2 4 3
MATE HYPE TYPE 4 RO 1040      ! Ogden (new) "Set 1" from P. Saez
                               ! (with reduced BULK)
                               BULK 10.E+03 ! This is in Pa
                               CO1 -10.8   ! This is alpha_1
                               CO2 19.7    ! This is alpha_2
                               CO5 -0.29   ! This is mu_1
                               CO6 9.6E-3  ! This is mu_2
                               LECT 1 TERM
FONC 1 TABL 2 0.0 1.0
      1.0 1.0
LINK COUP SPLT NONE
  BLOQ 2 LECT 1 2 TERM
  VITE 2 -1.0 FONC 1 LECT 3 4 TERM
INIT VITE 2 -1.0 LECT 3 4 TERM
ECRI DEPL VITE ACCE FINI FEXT CONT ECRO EPST TFRE 0.1
  FICH ALIC FREQ 1
OPTI PAS UTIL NOTE LOG 1
  STEL
  NOPR PLIN
  CSTA 0.5
CALC TINI 0. TFIN 0.8 PASF 1.E-3
FIN
```

### br2d00a.epx

```
BR2D00A
RESU ALIC 'br2d00.ali' GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx'  CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy'  CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sxy' CONT COMP 3 ELEM LECT 1 TERM
COUR 4 'sz'  CONT COMP 4 ELEM LECT 1 TERM
TRAC 1 2 3 4 AXES 1.0 'CONTR. [PA]' YZER
COUR 11 'ex' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'gxy' EPST COMP 3 ELEM LECT 1 TERM
COUR 14 'ez' EPST COMP 4 ELEM LECT 1 TERM
TRAC 11 12 13 14 AXES 1.0 'EPST [-]' YZER
COUR 21 'ecr1' ECRO COMP 1 ELEM LECT 1 TERM
COUR 22 'ecr2' ECRO COMP 2 ELEM LECT 1 TERM
COUR 23 'ecr3' ECRO COMP 3 ELEM LECT 1 TERM
COUR 24 'ecr4' ECRO COMP 4 ELEM LECT 1 TERM
COUR 25 'ecr5' ECRO COMP 5 ELEM LECT 1 TERM
COUR 26 'ecr6' ECRO COMP 6 ELEM LECT 1 TERM
COUR 27 'ecr7' ECRO COMP 7 ELEM LECT 1 TERM
TRAC 21 AXES 1.0 'ECR1' YZER
TRAC 22 AXES 1.0 'ECR2' YZER
TRAC 23 AXES 1.0 'ECR3' YZER
TRAC 24 AXES 1.0 'ECR4' YZER
TRAC 25 AXES 1.0 'ECR5' YZER
TRAC 26 AXES 1.0 'ECR6' YZER
TRAC 27 AXES 1.0 'ECR7' YZER
FIN
```

### br2d01.dgibi

```
*$$$ PX4CIR2D
*
* Pour generer le maillage 2D d'un quart de cercle
* avec seulement des quadrilateres a 4 noeuds.
* Le quart de cercle est defini par les deux extrems
* d'un arc (de 90 degrees) et par le centre du cercle.
*
* Input:
* =====
* P1 = premiere extremite de l'arc
* P2 = deuxieme extremite de l'arc
* PC = centre de l'arc
* N = nombre de mailles a generer sur chaque cote (doit etre pair)
* TOL= tolerance pour l'elimination des noeuds doubles
*
* Output:
* =====
* SUR = objet MAILLAGE d'elements de type QUA4
* IER = 0: pas d'erreur, .NE.0: erreur dans la generation de SUR
*
'DEBPROC' PX4CIR2D P1*'POINT' P2*'POINT' PC*'POINT'
      N*'ENTIER' TOL*'FLOTTANT';
*
*-----
*
ier=0;
n2 = n / 2;
```

```
p0 = 0 0;
pm1 = p1 plus p0;
depl pm1 tour 45 pc;
pm2 = 0.5*(pc plus p2);
pm3 = 0.5*(pc plus p1);
pm = 0.5*(pc plus pm1);
c1a = cerc n2 p1 pc pm1;
c1b = cerc n2 pm1 pc p2;
c2a = droi n2 p2 pm2;
c2b = droi n2 pm2 pc;
c3a = droi n2 pc pm3;
c3b = droi n2 pm3 p1;
c4a = droi n2 pm pm1;
c4b = droi n2 pm pm2;
c4c = droi n2 pm pm3;
sur1 = dall plan c4c c3b c1a (inve c4a);
sur2 = dall plan c4a c1b c2a (inve c4b);
sur3 = dall plan c2b c3a (inve c4c) c4b;
sur = sur1 et sur2 et sur3;
*
elim tol sur;
*
'FINPROC' sur ier;
*-----
opti echo 1;
opti dime 2 elem qua4;
opti sauv form 'br2d01.msh';
opti trac psc ftra 'br2d01_mesh.ps';
p0 = 0 0;
p1 = 0.06 0;
p2 = 0 0.06;
tol = 1.E-6;
b1 ier = px4cir2d p1 p2 p0 24 tol;
list ier;
p3 = 0.065 0;
c0 = p1 d 2 p3;
c1 = c0 rota 24 90.0 p0;
b2 = b1 tour 90 p0;
b3 = b2 tour 90 p0;
b4 = b3 tour 90 p0;
c2 = c1 tour 90 p0;
c3 = c2 tour 90 p0;
c4 = c3 tour 90 p0;
brain = b1 et b2 et b3 et b4;
elim tol brain;
csf = c1 et c2 et c3 et c4;
elim tol csf;
head = brain et csf;
elim tol head;
pskull = chan poil (cont head);
mesh = head et pskull;
tass mesh noop;
sauv form mesh;
trac qual mesh;
fin;
```

### br2d01.epx

```
BR2D01
ECHO
!CONV WIN
DPLA LAGR
CAST mesh
GEOM CAR4 brain csf TERM
COMP COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
MATE HYPE TYPE 4 RO 1040      ! Ogden (new) "Set 1" from P. Saez
                               ! (with reduced BULK)
                               BULK 10.E+03 ! This is in Pa
                               CO1 -10.8   ! This is alpha_1
                               CO2 19.7    ! This is alpha_2
                               CO5 -0.29   ! This is mu_1
                               CO6 9.6E-3  ! This is mu_2
                               LECT brain TERM
HYPE TYPE 4 RO 1040      ! Ogden (new) modified "Set 1"
                               ! from P. Saez for CSF (2)
                               BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
                               CO1 -10.8   ! This is alpha_1
                               CO2 19.7    ! This is alpha_2
                               CO5 -0.0029 ! This is mu_1_brain / 100
                               CO6 9.6E-5  ! This is mu_2_brain / 100
                               LECT csf TERM
FONC 1 TABL 3 0.0 1.0
      25.E-3 0.0
      50.E-3 0.0
LINK COUP SPLT NONE
  BLOQ 2 LECT pskull TERM
  VITE 1 0.1 FONC 1 LECT pskull TERM
INIT VITE 1 0.1 LECT brain csf TERM
ECRI CONT ECRO TFRE 10.E-3
  FICH ALIC TFRE 0.5E-3
OPTI PAS AUTO NOTE LOG 1
  STEL
  NOPR PLIN
```

```
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN
```

## br2d01z.epx

```
BR2D01Z
ECHO
CONV WIN
OPTI PRIN
RESU ALIC 'br2d01.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 0.00000E+00 0.00000E+00 3.95273E-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 : 0.00000E+00 0.00000E+00 0.00000E+00
!RSPHERE: 9.19239E-02
!RADIUS : 3.95273E-01
!ASPECT : 1.00000E+00
!NEAR : 3.03349E-01
!FAR : 5.79120E-01
SCEN GEOM NAVI FREE
ISO FILI FIEL CONT 1 SCAL USER PROG -32.5 PAS 5 32.5 TERM
COSC ICOL
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 101 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN
```

## br3d01.dgibi

```
*$$$ PX4CIR3D
*
* Pour generer le maillage 3D (plan) d'un quart de cercle
* avec seulement des quadrilateres a 4 noeuds.
* Le quart de cercle est defini par les deux extremes
* d'un arc (de 90 degrees), par le centre du cercle
* et par un autre point qui definit l'axe de rotation
* (axe perpendiculaire au plan du cercle, passant pour son centre).
*
* Input:
* =====
* P1 = premiere extremite de l'arc
* P2 = deuxieme extremite de l'arc
* PC = centre de l'arc
* PZ = autre point de l'axe
* N = nombre de mailles a generer sur chaque cote (doit etre pair)
* TOL= tolerance pour l'elimination des noeuds doubles
*
* Output:
* =====
* SUR = objet MAILLAGE d'elements de type QUA4
* IER = 0: pas d'erreur, .NE.0: erreur dans la generation de SUR
*
'DEBPROC' PX4CIR3D P1*'POINT' P2*'POINT' PC*'POINT' PZ*'POINT'
N*'ENTIER' TOL*'FLOTTANT';
-----
*
ier=0;
n2 = n / 2;
p0 = 0 0 0;
pm1 = p1 plus p0;
depl pm1 tour 45 pc pz;
pm2 = 0.5*(pc plus p2);
pm3 = 0.5*(pc plus p1);
pm = 0.5*(pc plus pm1);
c1a = cerc n2 p1 pc pm1;
c1b = cerc n2 pm1 pc p2;
c2a = droi n2 p2 pm2;
c2b = droi n2 pm2 pc;
c3a = droi n2 pc pm3;
c3b = droi n2 pm3 p1;
c4a = droi n2 pm pm1;
c4b = droi n2 pm pm2;
c4c = droi n2 pm pm3;
sur1 = dall plan c4c c3b c1a (inve c4a);
sur2 = dall plan c4a c1b c2a (inve c4b);
sur3 = dall plan c2b c3a (inve c4c) c4b;
sur = sur1 et sur2 et sur3;
*
elim tol sur;
*
'FINPROC' sur ier;
=====
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'br3d01.msh';
opti trac psc fra 'br3d01_mesh.ps';
p0 = 0 0 0;
pz = 0 0 1;
```

```
pz1 = 0 0 0.002;
p1 = 0.06 0 0;
p2 = 0 0.06 0;
tol = 1.E-6;
b1 ier = px4cir3d p1 p2 p0 pz 24 tol;
list ier;
p3 = 0.065 0 0;
c0 = p1 d 2 p3;
c1 = c0 rota 24 90.0 p0 pz;
b2 = b1 tour 90 p0 pz;
b3 = b2 tour 90 p0 pz;
b4 = b3 tour 90 p0 pz;
c2 = c1 tour 90 p0 pz;
c3 = c2 tour 90 p0 pz;
c4 = c3 tour 90 p0 pz;
brai = b1 et b2 et b3 et b4;
brain = brai volu tran 1 pz1;
elim tol brain;
cs = c1 et c2 et c3 et c4;
csf = cs volu tran 1 pz1;
elim tol csf;
head = brain et csf;
elim tol head;
pskull = head poin cyli p0 pz p3 tol;
mesh = head et pskull;
tass mesh noop;
sauv form mesh;
trac cach qual mesh;
fin;
```

## br3d01.epx

```
BR3D01
ECHO
!CONV WIN
TRID LAGR
CAST mesh
GEOM CUBE brain csf TERM
COMP COUL GR50 LECT brain TERM
TURQ LECT csf TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT csf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 3 LECT tous TERM
BLOQ 2 LECT pskull TERM
VITE 1 0.1 FONC 1 LECT pskull TERM
INIT VITE 1 0.1 LECT brain csf TERM
ECRI CONT ECR0 TFRE 1.E-3
FICH ALIC TFRE 0.5E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN
```

## br3d01z.epx

```
BR3D01Z
ECHO
CONV WIN
OPTI PRIN
RESU ALIC 'br3d01.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 0.00000E+00 0.00000E+00 3.95273E-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 : 0.00000E+00 0.00000E+00 0.00000E+00
!RSPHERE: 9.19239E-02
!RADIUS : 3.95273E-01
!ASPECT : 1.00000E+00
!NEAR : 3.03349E-01
!FAR : 5.79120E-01
SCEN GEOM NAVI FREE
! ISO FILI FIEL CONT 1 SCAL USER PROG -6.5E3 PAS 1.E3 6.5E3 TERM
ISO FILL FIEL CONT 1 SCAL USER PROG -325 PAS 50 325 TERM
SUPP LECT brain TERM
COSC ICOL
SLER CAM1 1 NFRA 1
FREQ 1
```

```

TRAC OFFS FICH AVI NOCL NFTO 101 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## br3d02.dgibi

```

*$$$ PX4CIR3D
*
* Pour generer le maillage 3D (plan) d'un quart de cercle
* avec seulement des quadrilateres a 4 noeuds.
* Le quart de cercle est defini par les deux extremes
* d'un arc (de 90 degrees), par le centre du cercle
* et par un autre point qui definit l'axe de rotation
* (axe perpendiculaire au plan du cercle, passant pour son centre).
*
* Input:
* =====
* P1 = premiere extremite de l'arc
* P2 = deuxieme extremite de l'arc
* PC = centre de l'arc
* PZ = autre point de l'axe
* N = nombre de mailles a generer sur chaque cote (doit etre pair)
* TOL= tolerance pour l'elimination des noeuds doubles
*
* Output:
* =====
* SUR = objet MAILLAGE d'elements de type QUA4
* IER = 0: pas d'erreur, .NE.0: erreur dans la generation de SUR
*
'DEBPROC' PX4CIR3D P1*'POINT' P2*'POINT' PC*'POINT' PZ*'POINT'
N*'ENTIER' TOL*'FLOTTANT';
-----
*
ier=0;
n2 = n / 2;
p0 = 0 0 0;
pm1 = p1 plus p0;
depl pm1 tour 45 pc pz;
pm2 = 0.5*(pc plus p2);
pm3 = 0.5*(pc plus p1);
pm = 0.5*(pc plus pm1);
c1a = cerc n2 p1 pc pm1;
c1b = cerc n2 pm1 pc p2;
c2a = droi n2 p2 pm2;
c2b = droi n2 pm2 pc;
c3a = droi n2 pc pm3;
c3b = droi n2 pm3 p1;
c4a = droi n2 pm pm1;
c4b = droi n2 pm pm2;
c4c = droi n2 pm pm3;
sur1 = dall plan c4c c3b c1a (inve c4a);
sur2 = dall plan c4a c1b c2a (inve c4b);
sur3 = dall plan c2b c3a (inve c4c) c4b;
sur = sur1 et sur2 et sur3;
*
elim tol sur;
*
'FINPROC' sur ier;
-----
opti echo 1;
opti dime 3 elem cub8;
opti sauv form 'br3d02.msh';
opti trac psc fra 'br3d02_mesh.ps';
p0 = 0 0 0;
pz = 0 0 1;
pz1 = 0 0 0.002;
p1 = 0.06 0 0;
p2 = 0 0.06 0;
tol = 1.E-6;
b1 ier = px4cir3d p1 p2 p0 pz 24 tol;
list ier;
b1 = chan b1 tri3;
p3 = 0.065 0 0;
c0 = p1 d 2 p3;
c1 = c0 rota 24 90.0 p0 pz;
c1 = chan c1 tri3;
b2 = b1 tour 90 p0 pz;
b3 = b2 tour 90 p0 pz;
b4 = b3 tour 90 p0 pz;
c2 = c1 tour 90 p0 pz;
c3 = c2 tour 90 p0 pz;
c4 = c3 tour 90 p0 pz;
brai = b1 et b2 et b3 et b4;
brain = brai volu tran 1 pz1;
elim tol brain;
cs = c1 et c2 et c3 et c4;
csf = cs volu tran 1 pz1;
elim tol csf;
head = brain et csf;
elim tol head;
pskull = head poin cyli p0 pz p3 tol;
mesh = head et pskull;
tass mesh noop;
sauv form mesh;
trac cach qual mesh;
fin;

```

## br3d02.epx

```

BR3D02
ECHO
!CONV WIN
TRID LAGR
CAST mesh
GEOM PRIS brain csf TERM
COMP COUL GR50 LECT brain TERM
TURQ LECT csf TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT csf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 3 LECT tous TERM
BLOQ 2 LECT pskull TERM
VITE 1 0.1 FONC 1 LECT pskull TERM
INIT VITE 1 0.1 LECT brain csf TERM
ECRI CONT EGRO TFRE 1.E-3
FICH ALIC TFRE 0.5E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN

```

## br3d02z.epx

```

BR3D02Z
ECHO
CONV WIN
OPTI PRIN
RESU ALIC 'br3d02.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 0.00000E+00 0.00000E+00 3.95273E-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 : 0.00000E+00 0.00000E+00 0.00000E+00
!RSPHERE: 9.19239E-02
!RADIUS : 3.95273E-01
!ASPECT : 1.00000E+00
!NEAR : 3.03349E-01
!FAR : 5.79120E-01
SCEN GEOM NAVI FREE
! ISO FILL FIEL CONT 1 SCAL USER PROG -6.5E3 PAS 1.E3 6.5E3 TERM
ISO FILL FIEL CONT 1 SCAL USER PROG -325 PAS 50 325 TERM
SUPP LECT brain TERM
CGSC ICOL
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 101 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## brainv.epx

```

BRAINV
ECHO
CONV WIN
LAGR TRID
CAST vol
GEOM TETR vol TERM
COMP GROU 11 'brain' LECT 1 PAS 1 414932 TERM
'csf' LECT 414933 PAS 1 576079 TERM
'head' LECT brain csf TERM
'braix1' LECT brain TERM COND XB LT -6.111965
'braix2' LECT brain DIFF braix1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965
'csfx2' LECT csf DIFF csfx1 TERM
'headx1' LECT braix1 csfx1 TERM
'headx2' LECT braix2 csfx2 TERM
'emin' LECT 3501 TERM
'emax' LECT 288434 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emin emax TERM

```

```

MATE LINE RO 8000. YOUN 1.D11 NU 0.3
      LECT head TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

## brat01.epx

```

BRAT01
ECHO
CONV WIN
LAGR TRID
GEOM LIBR POIN 10604 T3GS 21204 TERM
*NPTL= 10604 NELEM= 21204
-0.89411E+01 -0.93943E+01 0.18207E+01
-0.90577E+01 -0.93960E+01 0.18176E+01
. . . (lines skipped for brevity)
-0.70939E+01 -0.81860E+01 0.88329E+01
-0.68136E+01 -0.78949E+01 0.88452E+01
      1      2      3
      1      4      5
. . . (lines skipped for brevity)
      8336      8369      8517
      8335      8336      8517
COMP GROU 1 'surf' LECT 1 PAS 1 21204 TERM
      EPAI 1. LECT surf TERM
      COUL GR50 LECT surf TERM
MATE LINE RO 8000. YOUN 1.D11 NU 0.3
      LECT SURF TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

## brat02.epx

```

BRAT02
CONV WIN
LAGR TRID
GEOM LIBR POIN 115122 TETR 576079 TERM
*NPTL= 115122 NELEM= 576079
-0.11174E+02 -0.98448E+01 0.48253E+01
-0.11168E+02 -0.98924E+01 0.46769E+01
. . . (lines skipped for brevity)
-0.84757E+01 -0.48191E+01 0.27586E+01
-0.69070E+01 -0.57082E+01 0.81264E+01
      13852      13853      13855      13854
      13856      13857      13852      13858
. . . (lines skipped for brevity)
      101966      83697      105331      101967
      101846      101848      101845      94534
COMP GROU 9 'brain' LECT 1 PAS 1 414932 TERM
      'csf' LECT 414933 PAS 1 576079 TERM
      'head' LECT brain csf TERM
      'braix1' LECT brain TERM COND XB LT -6.111965
      'braix2' LECT brain DIFF braix1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965
      'csfx2' LECT csf DIFF csfx1 TERM
      'headx1' LECT braix1 csfx1 TERM
      'headx2' LECT braix2 csfx2 TERM
      COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
MATE LINE RO 8000. YOUN 1.D11 NU 0.3
      LECT head TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

## brat03.epx

```

BRAT03
CONV WIN
LAGR TRID
GEOM 9 '(3E13.5)' '(418)' POIN 115122 TETR 576079 TERM
COMP GROU 9 'brain' LECT 1 PAS 1 414932 TERM
      'csf' LECT 414933 PAS 1 576079 TERM
      'head' LECT brain csf TERM
      'braix1' LECT brain TERM COND XB LT -6.111965
      'braix2' LECT brain DIFF braix1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965
      'csfx2' LECT csf DIFF csfx1 TERM
      'headx1' LECT braix1 csfx1 TERM
      'headx2' LECT braix2 csfx2 TERM
      COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
MATE LINE RO 8000. YOUN 1.D11 NU 0.3
      LECT head TERM
ECRI DEPL VITE ACCE FINT FEXT FLIA FDEC CONT ECRO FREQ 100
OPTI PAS AUTO NOTE LOG 1
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

## brat04.epx

```

BRAT04
ECHO
CONV WIN
TRID ALE
CAST 'head.msh' head
DIME NALE 8828 NBLE 14484 TERM
GEOM SCAL FACT 0.001
      TETR brain csf T3GS skull TERM
COMP GROU 12 'brax1' LECT brain TERM COND XB LT -6.111965E-3
      'brax2' LECT brain DIFF brax1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965E32
      'csfx2' LECT csf DIFF csfx1 TERM
      'skux1' LECT skull TERM COND XB LT -6.111965E-3
      'skux2' LECT skull DIFF skux1 TERM
      'headx1' LECT brax1 csfx1 skux1 TERM
      'headx2' LECT brax2 csfx2 skux2 TERM
      'emin' LECT 3501 TERM
      'emax' LECT 288434 TERM
      'edtmin' LECT 54966 TERM
      'edtmax' LECT 248730 TERM
      EPAI 1.E-3 LECT skull TERM
      COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      ROUG LECT emin emax TERM
      ROSE LECT edtmin edtmax TERM
GRIL LAGR LECT brain skull TERM
      EULE LECT fsan TERM
      AUTO AUTR
MATE HYPE TYPE 4 RO 1040
      BULK 128E+06
      CO1 -0.14058E+02
      CO2 0.38793E+00
      CO5 0.88365E+06
      CO6 0.50888E+09
      LECT brain TERM
      FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
      LINE RO 1800 YOUN 1.500E9 NU 0.21
      LECT skull TERM
FONC 1 TABL 3 0.0 1.0
      2.E-3 0.0
      10.0 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
      BLOQ 13456 LECT skull TERM
      VITE 2 100.0 FONC 1 LECT skull TERM
      FSA LECT fsan TERM
      INIT VITE 2 100.0 LECT brain csf skull TERM
      VITC 2 100.0 LECT brain csf skull TERM
ECRI TPRE 1.E-5
      FICH SPLI ALIC TPRE 1.E-5
OPTI PAS AUTO NOTE LOG 1
      NOPR PLIN
      CSTA 0.5
      NOCR LECT skull TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0. TFIN 4.D-3
FIN

```

## brat05.epx

```

BRAT05
ECHO
CONV WIN
TRID ALE
EROS 1.0
CAST 'head.msh' head
DIME NALE 8828 NBLE 14484 TERM
GEOM SCAL FACT 0.001
      TETR brain csf T3GS skull TERM
COMP GROU 13 'brax1' LECT brain TERM COND XB LT -6.111965E-3
      'brax2' LECT brain DIFF brax1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965E-3
      'csfx2' LECT csf DIFF csfx1 TERM
      'skux1' LECT skull TERM COND XB LT -6.111965E-3
      'skux2' LECT skull DIFF skux1 TERM
      'headx1' LECT brax1 csfx1 skux1 TERM
      'headx2' LECT brax2 csfx2 skux2 TERM
      'emin' LECT 3501 TERM
      'emax' LECT 288434 TERM
      'edtmin' LECT 417537 TERM
      'edtmax' LECT 248730 TERM
      'eroded' LECT 3501 7951 54949 54966 213528 TERM
      EPAI 1.E-3 LECT skull TERM
      COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      ROUG LECT eroded TERM
      VERT LECT emax edtmax TERM
GRIL LAGR LECT brain skull TERM
      EULE LECT fsan TERM
      AUTO AUTR
MATE HYPE TYPE 4 RO 1040
      BULK 128E+06
      CO1 -0.14058E+02
      CO2 0.38793E+00
      CO5 0.88365E+06
      CO6 0.50888E+09
      LECT brain TERM
      FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT csf TERM
      LINE RO 1800 YOUN 1.500E9 NU 0.21

```

```

LECT skull TERM
FONC 1 TABL 3 0.0 1.0
      2.E-3 0.0
      10.0 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
BLOQ 13456 LECT skull TERM
VITE 2 100.0 FONC 1 LECT skull TERM
FSA LECT fsan TERM
INIT VITE 2 100.0 LECT brain csf skull TERM
VITG 2 100.0 LECT brain csf skull TERM
ECRI TPRE 1.E-5
FICH SPLI ALIC TPRE 1.E-5
OPTI PAS AUTO NOTE LOG 1
      STEL
      NOPR PLIN
      CSTA 0.5
      NOCR LECT skull TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0. TFIN 4.D-3 TFAI 3.51E-9
PLAY
MEAS EMIN LECT tous TERM TERM
MEAS EMAX LECT tous TERM TERM
ENDPLAY
FIN

```

## brat06.epx

```

BRAT06
ECHO
!CONV WIN
TRID ALE
EROS 1.0
CAST 'head.msh' head
DIME NALE 8828 NBLE 14484 TERM
GEOM SCAL FACT 0.001
TETR brain csf T3GS skull TERM
COMP GROU 12 'brax1' LECT brain TERM COND XB LT -6.111965E-3
      'brax2' LECT brain DIFF brax1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965E-3
      'csfx2' LECT csf DIFF csfx1 TERM
      'skux1' LECT skull TERM COND XB LT -6.111965E-3
      'skux2' LECT skull DIFF skux1 TERM
      'headx1' LECT brax1 csfx1 skux1 TERM
      'headx2' LECT brax2 csfx2 skux2 TERM
      'emin' LECT 3501 TERM
      'emax' LECT 288434 TERM
      'edtmin' LECT 54966 TERM
      'edtmax' LECT 248730 TERM
EPAI 1.E-3 LECT skull TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
GRIL LAGR LECT brain skull TERM
EULE LECT fsan TERM
AUTO AUTR
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FONC 1 TABL 3 0.0 1.0
      25.E-3 0.0
      50.E-3 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
BLOQ 13456 LECT skull TERM
VITE 2 0.1 FONC 1 LECT skull TERM
FSA LECT fsan TERM
INIT VITE 2 0.1 LECT brain csf skull TERM
VITG 2 0.1 LECT brain csf skull TERM
ECRI TPRE 1.E-5
FICH SPLI ALIC TPRE 1.E-6
OPTI PART NOTE LOG 1
      NOPR PLIN PLOG
      CSTA 0.5
      NOCR LECT skull TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0. TFIN 50.D-3 ! TFAI 4.E-8
FIN

```

## brat06z.epx

```

BRAT06Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat06.ali' GARD PSCR
COMP NGRO 1 'nvmax' LECT 93485 TERM
SORT VISU NSTO 1
FIN

```

## brat07.epx

```

BRAT07
ECHO
!CONV WIN
TRID ALE
EROS 1.0
CAST 'head.msh' head
DIME NALE 8828 NBLE 14484 TERM
GEOM SCAL FACT 0.001
TETR brain csf T3GS skull TERM
COMP GROU 12 'brax1' LECT brain TERM COND XB LT -6.111965E-3
      'brax2' LECT brain DIFF brax1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965E-3
      'csfx2' LECT csf DIFF csfx1 TERM
      'skux1' LECT skull TERM COND XB LT -6.111965E-3
      'skux2' LECT skull DIFF skux1 TERM
      'headx1' LECT brax1 csfx1 skux1 TERM
      'headx2' LECT brax2 csfx2 skux2 TERM
      'emin' LECT 3501 TERM
      'emax' LECT 288434 TERM
      'edtmin' LECT 54966 TERM
      'edtmax' LECT 248730 TERM
EPAI 1.E-3 LECT skull TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
GRIL LAGR LECT brain skull TERM
EULE LECT fsan TERM
AUTO AUTR
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FONC 1 TABL 3 0.0 1.0
      25.E-3 0.0
      50.E-3 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
BLOQ 13456 LECT skull TERM
VITE 2 0.1 FONC 1 LECT skull TERM
FSA LECT fsan TERM
INIT VITE 2 0.1 LECT brain csf skull TERM
VITG 2 0.1 LECT brain csf skull TERM
ECRI TPRE 1.E-5
FICH SPLI ALIC TPRE 1.E-6
OPTI PART NOTE LOG 1
      NOPR PLIN PLOG
      CSTA 0.5
      NOCR LECT skull TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0. TFIN 50.D-3 ! TFAI 4.E-8
FIN

```

## brat08.epx

```

BRAT08
ECHO
CONV WIN
TRID ALE
EROS 1.0
CAST 'head.msh' head
DIME NALE 8828 NBLE 14484 TERM
GEOM SCAL FACT 0.001
TETR brain csf T3GS skull TERM
COMP GROU 12 'brax1' LECT brain TERM COND XB LT -6.111965E-3
      'brax2' LECT brain DIFF brax1 TERM
      'csfx1' LECT csf TERM COND XB LT -6.111965E-3
      'csfx2' LECT csf DIFF csfx1 TERM
      'skux1' LECT skull TERM COND XB LT -6.111965E-3
      'skux2' LECT skull DIFF skux1 TERM
      'headx1' LECT brax1 csfx1 skux1 TERM
      'headx2' LECT brax2 csfx2 skux2 TERM
      'emin' LECT 3501 TERM
      'emax' LECT 288434 TERM
      'edtmin' LECT 54966 TERM
      'edtmax' LECT 248730 TERM
EPAI 1.E-3 LECT skull TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
GRIL LAGR LECT brain skull TERM
EULE LECT fsan TERM
AUTO AUTR
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1

```



```

CO6 9.6E-3 ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
OPTI TOLC 1.E-7
LINK COUP SPLT NONE
BLOQ 13456 LECT skull TERM
VITE 2 0.1 FONC 1 LECT skull TERM
FSA LECT fsan TERM
INIT VITE 2 0.1 LECT brain csf skull TERM
VITG 2 0.1 LECT brain csf skull TERM
ECRI TFRE 1.E-5
FICH SPLI ALIC TFRE 1.E-6
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
NOCR LECT skull TERM
REZO MVRE MODU VFAC 5.0
CALC TINI 0. TFIN 50.D-3 ! TFAI 4.E-8
FIN

```

## brat09.epx

```

BRAT09
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 10 'brax1' LECT brain TERM COND XB LT -6.111965E-3
'brax2' LECT brain DIFF brax1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965E-3
'csfx2' LECT csf DIFF csfx1 TERM
'headx1' LECT brax1 csfx1 TERM
'headx2' LECT brax2 csfx2 TERM
'emini' LECT 21034 TERM
'emax' LECT 210892 TERM
'edtmin' LECT 46333 TERM
'edtmax' LECT 366143 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emini emax TERM
ROSE LECT edtmin edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.029 ! This is mu_1_brain / 10
CO6 9.6E-4 ! This is mu_2_brain / 10
LECT csf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT pskull TERM
VITE 2 0.1 FONC 1 LECT pskull TERM
INIT VITE 2 0.1 LECT brain csf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN

```

## brat09p.epx

```

BRAT09P
ECHO
OPTI PRIN
RESU SPLI ALIC 'brat09.ali' GARD PSCR
COMP NGRO 1 'nbar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
GROU 1 'ebar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_bary' DEPL COMP 2 NOEU LECT nbar TERM
COUR 2 'vy_bary' VITE COMP 2 NOEU LECT nbar TERM
COUR 3 'sy_bary' CONT COMP 2 ELEM LECT ebar TERM
TRAC 1 AXES 1.0 'DISP. [M]' YZER

```

```

TRAC 2 AXES 1.0 'VELO. [M/S]' YZER
TRAC 3 AXES 1.0 'SIG_Y [PA]' YZER
FIN

```

## brat09z.epx

```

BRAT09Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat09.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.66988E-02 -6.58819E-03 1.11378E-02
! Q 5.32609E-01 4.39079E-01 4.89737E-01 5.32630E-01
VIEW -9.89410E-01 -5.39818E-02 -1.34734E-01
RIGH -4.70737E-02 9.97434E-01 -5.39444E-02
FOV -1.37300E-01 4.70307E-02 9.89412E-01
FOV 2.48819E+01
!NAVIGATION MODE: FREE CAMERA
!CENTER : -6.11197E-03 -9.06235E-03 5.30795E-03
!RSPHERE: 1.12704E-02
!RADIUS : 4.32767E-02
!ASPECT : 1.00000E+00
!NEAR : 3.20061E-02
!FAR : 5.45469E-02
SCEN GEOM NAVI FREE
ISO FILI FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
COSC ICOL
LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 291 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 289 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## brat10.epx

```

BRAT10
ECHO
CONV WIN
TRID LAGR
EROS 1.0
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 10 'brax1' LECT brain TERM COND XB LT -6.111965E-3
'brax2' LECT brain DIFF brax1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965E-3
'csfx2' LECT csf DIFF csfx1 TERM
'headx1' LECT brax1 csfx1 TERM
'headx2' LECT brax2 csfx2 TERM
'emini' LECT 21034 TERM
'emax' LECT 210892 TERM
'edtmin' LECT 46333 TERM
'edtmax' LECT 366143 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emini emax TERM
ROSE LECT edtmin edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.029 ! This is mu_1_brain / 10
CO6 9.6E-4 ! This is mu_2_brain / 10
LECT csf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT pskull TERM
VITE 2 0.1 FONC 1 LECT pskull TERM
INIT VITE 2 0.1 LECT brain csf TERM
ECRI FAIL TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 TFAI 4.E-4
FIN

```

## brat11.epx

```

BRAT11
ECHO
!CONV WIN
TRID LAGR
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 11 'brax1' LECT brain TERM COND XB LT -6.111965E-3
'brax2' LECT brain DIFF brax1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965E-3
'csfx2' LECT csf DIFF csfx1 TERM
'headx1' LECT brax1 csfx1 TERM
'headx2' LECT brax2 csfx2 TERM
'emin' LECT 21034 TERM
'emax' LECT 210892 TERM
'edtmin' LECT 46333 TERM
'edtmax' LECT 366143 TERM
'eros0' LECT
325 442 2596 4865 11840 12169 12796 14388 14979
16761 19332 19700 19705 20594 20760 21034 24095 24491
24673 28622 29474 30334 32155 32335 32608 34418 38135
38183 38635 39826 40471 42528 42533 42814 43546 43559
44298 44976 45134 45166 45185 45191 45197 45555 45647
46333 48732 49865 51028 51045 51555 52699 53221 53227
54419 54428 56701 57449 58862 60816 60820 61249 61697
63731 65319 68391 70701 72388 73405 74335 76078 76942
78745 80091 81224 82864 83571 83578 85971 87944 88348
88631 88632 88643 88876 91595 91644 91984 92617 93104
93976 93977 94331 94527 94539 94541 94889 94933 94974
95214 95215 96489 99039 99040 99047 99203 99703 99710
101280 102146 104155 106481 107593 107743 108522 108561 108924
109563 110252 112554 112976 113268 113694 113843 114060 114697
116972 117568 118945 122155 123508 123853 124962 124967 125644
127424 128262 131002 132564 132566 133021 134235 134747 135095
135233 135797 138470 138471 138472 139759 140876 141688 144318
144319 144320 147054 147173 148869 148870 148874 153876 154432
163457 165357 169108 173758 175952 177506 184897 187389 187390
187391 189011 198321 199797 207838 225833 446840
TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.029 ! This is mu_1_brain / 10
CO6 9.6E-4 ! This is mu_2_brain / 10
LECT csf TERM
FANT 1.E-3 LECT eros0 TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT pskull TERM
VITE 2 0.1 FONC 1 LECT pskull TERM
INIT VITE 2 0.1 LECT brain csf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8
FIN

```

## brat11p.epx

```

BRAT11P
ECHO
OPTI PRIN
RESU SPLI ALIC 'brat11.ali' GARD PSCR
COMP NGRO 1 'nbar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
GROU 1 'ebar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_bary' DEPL COMP 2 NOEU LECT nbar TERM
COUR 2 'vy_bary' VITE COMP 2 NOEU LECT nbar TERM
COUR 3 'sy_bary' CONT COMP 2 ELEM LECT ebar TERM
TRAC 1 AXES 1.0 'DISP. [M]' YZER
TRAC 2 AXES 1.0 'VELO. [M/S]' YZER
TRAC 3 AXES 1.0 'SIG_Y [PA]' YZER
FIN

```

## brat11z.epx

```
BRAT11Z
```

```

ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat11.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.66988E-02 -6.58819E-03 1.11378E-02
! Q 5.32609E-01 4.39079E-01 4.89737E-01 5.32630E-01
VIEW -9.89410E-01 -5.39818E-02 -1.34734E-01
RIGH -4.70737E-02 9.97434E-01 -5.39444E-02
UP -1.37300E-01 4.70307E-02 9.89412E-01
FOV 2.48819E+01
! NAVIGATION MODE: FREE CAMERA
! CENTER : -6.11197E-03 -9.06235E-03 5.30795E-03
! RSPHERE: 1.12704E-02
! RADIUS : 4.32767E-02
! ASPECT : 1.00000E+00
! NEAR : 3.20061E-02
! FAR : 5.45469E-02
SCEN GEOM NAVI FREE
ISO FILI FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
COSC ICOL
LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 281 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 279 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## brat12.epx

```

BRAT12
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 10 'brax1' LECT brain TERM COND XB LT -6.111965E-3
'brax2' LECT brain DIFF brax1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965E-3
'csfx2' LECT csf DIFF csfx1 TERM
'headx1' LECT brax1 csfx1 TERM
'headx2' LECT brax2 csfx2 TERM
'emin' LECT 21034 TERM
'emax' LECT 210892 TERM
'edtmin' LECT 46333 TERM
'edtmax' LECT 366143 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.029 ! This is mu_1_brain / 10
CO6 9.6E-4 ! This is mu_2_brain / 10
LECT csf TERM
FANT 1.E-3 LECT eros0 TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT pskull TERM
VITE 2 0.1 FONC 1 LECT pskull TERM
INIT VITE 2 0.1 LECT brain csf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN

```

## brat12p.epx

```

BRAT12P
ECHO
OPTI PRIN
RESU SPLI ALIC 'brat12.ali' GARD PSCR
COMP NGRO 1 'nbar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
GROU 1 'ebar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
SORT GRAP
AXTE 1.0 'Time [s]'

```

```

COUR 1 'dy_bary' DEPL COMP 2 NOEU LECT nbar TERM
COUR 2 'vy_bary' VITE COMP 2 NOEU LECT nbar TERM
COUR 3 'sy_bary' CONT COMP 2 ELEM LECT ebar TERM
TRAC 1 AXES 1.0 'DISP. [M]' YZER
TRAC 2 AXES 1.0 'VELO. [M/S]' YZER
TRAC 3 AXES 1.0 'SIG_Y [PA]' YZER
FIN

```

## brat12w.epx

```

BRAT12Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat12.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.66988E-02 -6.58819E-03 1.11378E-02
! Q 5.32609E-01 4.39079E-01 4.89737E-01 5.32630E-01
! VIEW -9.89410E-01 -5.39818E-02 -1.34734E-01
! RIGH -4.70737E-02 9.97434E-01 -5.39444E-02
! UP -1.37300E-01 4.70307E-02 9.89412E-01
! FOV 2.48819E+01
!!NAVIGATION MODE: FREE CAMERA
!!CENTER : -6.11197E-03 -9.06235E-03 5.30795E-03
!!RSPHERE: 1.12704E-02
!!RADIUS : 4.32767E-02
!!ASPECT : 1.00000E+00
!!NEAR : 3.20061E-02
!!FAR : 5.45469E-02
SCEN GEOM NAVI FREE
ISO FILL FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
COSC ICOL
LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTA 291 FPS 15 KFRE 10 COMP -1 REND
GOTR LOOP 289 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## brat12z.epx

```

BRAT12Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat12.ali' GARD PSCR
COMP GROU 1 'csfz1' LECT csf TERM COND ZB LT 5.3E-3
SORT VISU NSTO 1
PLAY
!CAME 1 EYE 3.66988E-02 -6.58819E-03 1.11378E-02
!! Q 5.32609E-01 4.39079E-01 4.89737E-01 5.32630E-01
! VIEW -9.89410E-01 -5.39818E-02 -1.34734E-01
! RIGH -4.70737E-02 9.97434E-01 -5.39444E-02
! UP -1.37300E-01 4.70307E-02 9.89412E-01
! FOV 2.48819E+01
!!NAVIGATION MODE: FREE CAMERA
!!CENTER : -6.11197E-03 -9.06235E-03 5.30795E-03
!!RSPHERE: 1.12704E-02
!!RADIUS : 4.32767E-02
!!ASPECT : 1.00000E+00
!!NEAR : 3.20061E-02
!!FAR : 5.45469E-02
SCEN !OBJE USLM LECT csf TERM
! DHAS CGLA
!GEOM LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
SUPP LECT brain TERM
COSC ICOL
LIMA ON
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH AVI NOCL NFTA 116 FPS 15 KFRE 10 COMP -1 REND
!GOTR LOOP 114 OFFS FICH AVI CONT NOCL REND
TRAC OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
GOTR LOOP 114 OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
GO
!TRAC OFFS FICH AVI CONT REND
TRAC OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
ENDPLAY
FIN

```

## brat13.epx

```

BRAT13
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'head1.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 10 'brax1' LECT brain TERM COND XB LT -6.111965E-3
'brax2' LECT brain DIFF brax1 TERM
'csfx1' LECT csf TERM COND XB LT -6.111965E-3
'csfx2' LECT csf DIFF csfx1 TERM

```

```

'headx1' LECT brax1 csfx1 TERM
'headx2' LECT brax2 csfx2 TERM
'emini' LECT 21034 TERM
'emax' LECT 210892 TERM
'edtmin' LECT 46333 TERM
'edtmax' LECT 366143 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
ROUG LECT emin emax TERM
ROSE LECT edtmin edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT brain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT csf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT pskull TERM
VITE 2 10. FONC 1 LECT pskull TERM
INIT VITE 2 10. LECT brain csf TERM
ECRI TPRE 0.1E-3
FICH SPLI ALIC TPRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN

```

## brat13p.epx

```

BRAT13P
ECHO
OPTI PRIN
RESU SPLI ALIC 'brat13.ali' GARD PSCR
COMP GROU 1 'nbar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
GROU 1 'ebar' LECT tous TERM
COND NEAR POIN -6.1317E-3 -8.4795E-3 5.2641E-3
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dy_bary' DEPL COMP 2 NOEU LECT nbar TERM
COUR 2 'vy_bary' VITE COMP 2 NOEU LECT nbar TERM
COUR 3 'sy_bary' CONT COMP 2 ELEM LECT ebar TERM
TRAC 1 AXES 1.0 'DISP. [M]' YZER
TRAC 2 AXES 1.0 'VELO. [M/S]' YZER
TRAC 3 AXES 1.0 'SIG_Y [PA]' YZER
FIN

```

## brat13z.epx

```

BRAT13Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat13.ali' GARD PSCR
COMP GROU 1 'csfz1' LECT csf TERM COND ZB LT 5.3E-3
SORT VISU NSTO 1
PLAY
!CAME 1 EYE 3.66988E-02 -6.58819E-03 1.11378E-02
!! Q 5.32609E-01 4.39079E-01 4.89737E-01 5.32630E-01
! VIEW -9.89410E-01 -5.39818E-02 -1.34734E-01
! RIGH -4.70737E-02 9.97434E-01 -5.39444E-02
! UP -1.37300E-01 4.70307E-02 9.89412E-01
! FOV 2.48819E+01
!!NAVIGATION MODE: FREE CAMERA
!!CENTER : -6.11197E-03 -9.06235E-03 5.30795E-03
!!RSPHERE: 1.12704E-02
!!RADIUS : 4.32767E-02
!!ASPECT : 1.00000E+00
!!NEAR : 3.20061E-02
!!FAR : 5.45469E-02
SCEN !OBJE USLM LECT csf TERM
! DHAS CGLA
!GEOM LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -2600 PAS 400 2600 TERM
SUPP LECT brain TERM
COSC ICOL
LIMA ON
!SLER CAM1 1 NFRA 1
!TRAC OFFS FICH AVI NOCL NFTA 38 FPS 15 KFRE 10 COMP -1 REND
!GOTR LOOP 36 OFFS FICH AVI CONT NOCL REND
TRAC OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
GOTR LOOP 36 OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
GO
!TRAC OFFS FICH AVI CONT REND
TRAC OFFS FICH BMP OBJE LECT csfz1 brain TERM REND
ENDPLAY
FIN

```

## brat21.epx

```
BRAT21
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 3 'sbrain' LECT brain TERM COND XB GT -6.25E-3
COND XB LT -5.75E-3
'scsf' LECT csf TERM COND XB GT -6.25E-3
COND XB LT -5.75E-3
'shead' LECT sbrain scsf TERM
NGRO 1 'spskull' LECT pskull TERM COND X GT -6.25E-3
COND X LT -5.75E-3
COUL GR50 LECT sbrain TERM
TURQ LECT scsf TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT sbrain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT scsf TERM
FANT 1.E-6 LECT head DIFF sbrain scsf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT spskull TERM
BLOQ 1 LECT shead TERM
VITE 2 0.1 FONC 1 LECT spskull TERM
INIT VITE 2 0.1 LECT sbrain scsf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN
```

## brat21c.epx

```
BRAT21C
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat21.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.93294E-02 -9.63135E-03 5.40310E-03
! Q 5.00000E-01 5.00000E-01 5.00000E-01 5.00000E-01
VIEW -1.00000E+00 -2.05103E-10 2.05102E-10
RIGH -2.05101E-10 1.00000E+00 5.55112E-17
UP 2.05104E-10 1.88738E-15 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : -6.01410E-03 -9.63135E-03 5.40310E-03
!RSPHERE: 9.06869E-03
!RADIUS : 4.53435E-02
!ASPECT : 1.00000E+00
!NEAR : 3.53679E-02
!FAR : 6.34808E-02
SCEN !OBJE USLM LECT csf TERM
! DHAS CGLA
GEOM NAVI FREE
LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -65 PAS 10 65 TERM
SUPP LECT scsf TERM
COSC ICOL
!LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 501 FPS 25 KFRE 10 COMP -1
OBJE LECT shead TERM REND
GOTR LOOP 499 OFFS FICH AVI CONT NOCL
OBJE LECT shead TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT shead TERM REND
ENDPLAY
FIN
```

## brat21z.epx

```
BRAT21Z
ECHO
CONV WIN
```

```
OPTI PRIN
RESU SPLI ALIC 'brat21.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.93294E-02 -9.63135E-03 5.40310E-03
! Q 5.00000E-01 5.00000E-01 5.00000E-01 5.00000E-01
VIEW -1.00000E+00 -2.05103E-10 2.05102E-10
RIGH -2.05101E-10 1.00000E+00 5.55112E-17
UP 2.05104E-10 1.88738E-15 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : -6.01410E-03 -9.63135E-03 5.40310E-03
!RSPHERE: 9.06869E-03
!RADIUS : 4.53435E-02
!ASPECT : 1.00000E+00
!NEAR : 3.53679E-02
!FAR : 6.34808E-02
SCEN !OBJE USLM LECT csf TERM
! DHAS CGLA
GEOM NAVI FREE
! LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
SUPP LECT sbrain TERM
COSC ICOL
LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 501 FPS 15 KFRE 10 COMP -1
OBJE LECT shead TERM REND
GOTR LOOP 499 OFFS FICH AVI CONT NOCL
OBJE LECT shead TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT shead TERM REND
ENDPLAY
FIN
```

## brat22.epx

```
BRAT22
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'headl.msh' head
GEOM SCAL FACT 0.001
TETR brain csf TERM
COMP GROU 3 'sbrain' LECT brain TERM COND XB GT -6.25E-3
COND XB LT -5.75E-3
'scsf' LECT csf TERM COND XB GT -6.25E-3
COND XB LT -5.75E-3
'shead' LECT sbrain scsf TERM
NGRO 1 'spskull' LECT pskull TERM COND X GT -6.25E-3
COND X LT -5.75E-3
COUL GR50 LECT sbrain TERM
TURQ LECT scsf TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT sbrain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT scsf TERM
FANT 1.E-6 LECT head DIFF sbrain scsf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT spskull TERM
BLOQ 1 LECT shead TERM
VITE 2 0.1 FONC 1 LECT spskull TERM
INIT VITE 2 0.1 LECT sbrain scsf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
ELEM LECT shead TERM
POIN LECT shead TERM
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN
```

## brat23.epx

```
BRAT23
ECHO
!CONV WIN
TRID LAGR
EROS 1.0
CAST 'slice.msh' shead
GEOM SCAL FACT 0.001
TETR sbrain scsf TERM
```

```

COMP COUL GR50 LECT sbrain TERM
TURQ LECT scsf TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new) "Set 1" from P. Saez
! (with reduced BULK)
BULK 10.E+03 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT sbrain TERM
HYPE TYPE 4 RO 1040 ! Ogden (new) modified "Set 1"
! from P. Saez for CSF (2)
BULK 10.E+05 ! This is in Pa and = bulk_brain * 100
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.0029 ! This is mu_1_brain / 100
CO6 9.6E-5 ! This is mu_2_brain / 100
LECT scsf TERM
FONC 1 TABL 3 0.0 1.0
25.E-3 0.0
50.E-3 0.0
LINK COUP SPLT NONE
BLOQ 13 LECT spskull TERM
BLOQ 1 LECT shead TERM
VITE 2 0.1 FONC 1 LECT spskull TERM
INIT VITE 2 0.1 LECT sbrain scsf TERM
ECRI TFRE 0.1E-3
FICH SPLI ALIC TFRE 0.1E-3
OPTI PAS AUTO NOTE LOG 1
STEL
NOPR PLIN
CSTA 0.5
CALC TINI 0. TFIN 50.D-3 PAS1 1.E-8 ! TFAI 4.E-8
FIN

```

### brat23c.epx

```

BRAT23C
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat23.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.93294E-02 -9.63135E-03 5.40310E-03
! Q 5.00000E-01 5.00000E-01 5.00000E-01 5.00000E-01
VIEW -1.00000E+00 -2.05103E-10 2.05102E-10
RIGH -2.05101E-10 1.00000E+00 5.55112E-17
UP 2.05104E-10 1.88738E-15 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : -6.01410E-03 -9.63135E-03 5.40310E-03
!RSPHERE: 9.06869E-03
!RADIUS : 4.53435E-02
!ASPECT : 1.00000E+00
!NEAR : 3.53679E-02
!FAR : 6.34808E-02
SCEN !OBJE USLM LECT csf TERM
! DHAS CGLA
GEOM NAVI FREE
LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -65 PAS 10 65 TERM
SUPP LECT scsf TERM
COSC ICOL
!LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 501 FPS 25 KFRE 10 COMP -1
OBJE LECT shead TERM REND
GOTR LOOP 499 OFFS FICH AVI CONT NOCL
OBJE LECT shead TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT shead TERM REND
ENDPLAY
FIN

```

### brat23z.epx

```

BRAT23Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'brat23.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 3.93294E-02 -9.63135E-03 5.40310E-03
! Q 5.00000E-01 5.00000E-01 5.00000E-01 5.00000E-01
VIEW -1.00000E+00 -2.05103E-10 2.05102E-10
RIGH -2.05101E-10 1.00000E+00 5.55112E-17
UP 2.05104E-10 1.88738E-15 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : -6.01410E-03 -9.63135E-03 5.40310E-03
!RSPHERE: 9.06869E-03
!RADIUS : 4.53435E-02
!ASPECT : 1.00000E+00
!NEAR : 3.53679E-02
!FAR : 6.34808E-02
SCEN !OBJE USLM LECT csf TERM

```

```

! DHAS CGLA
GEOM NAVI FREE
! LINE HEOU
ISO FILL FIEL CONT 2 SCAL USER PROG -32.5 PAS 5 32.5 TERM
SUPP LECT sbrain TERM
COSC ICOL
LIMA ON
SLER CAM1 1 NFRA 1
TRAC OFFS FICH AVI NOCL NFTO 501 FPS 15 KFRE 10 COMP -1
OBJE LECT shead TERM REND
GOTR LOOP 499 OFFS FICH AVI CONT NOCL
OBJE LECT shead TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT shead TERM REND
ENDPLAY
FIN

```

### delt01.epx

```

DELTO1
ECHO
CONV WIN
LAGR TRID
GEOM LIBR POIN 4 TETR 1 TERM
0 0 0 1 0 0 0 1 0 0 1
1 2 3 4
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
BULK 128E+06 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT 1 TERM
ECRI DEPL VITE ACCE FINT FEXT CONT EPST ECRO FREQ 1
OPTI PAS AUTO NOTE LOG 1
STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

### delt02.epx

```

DELTO2
ECHO
CONV WIN
LAGR TRID
GEOM LIBR POIN 4 TETR 1 TERM
0 0 0 1 0 0 0 1 0 0 1
1 2 3 4
MATE FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT 1 TERM
ECRI DEPL VITE ACCE FINT FEXT CONT EPST ECRO FREQ 1
OPTI PAS AUTO NOTE LOG 1
STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

### delt03.epx

```

DELTO3
ECHO
CONV WIN
LAGR TRID
GEOM LIBR POIN 4 TETR 1 TERM
0 0 0 1.E-3 0 0 0 1.E-3 0 0 0 1.E-3
1 2 3 4
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
BULK 128E+06 ! This is in Pa
CO1 -10.8 ! This is alpha_1
CO2 19.7 ! This is alpha_2
CO5 -0.29 ! This is mu_1
CO6 9.6E-3 ! This is mu_2
LECT 1 TERM
ECRI DEPL VITE ACCE FINT FEXT CONT EPST ECRO FREQ 1
OPTI PAS AUTO NOTE LOG 1
STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

### delt04.epx

```

DELTO4
ECHO
CONV WIN
LAGR TRID
GEOM LIBR POIN 4 TETR 1 TERM
0 0 0 1.E-3 0 0 0 1.E-3 0 0 0 1.E-3
1 2 3 4
MATE FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT 1 TERM
ECRI DEPL VITE ACCE FINT FEXT CONT EPST ECRO FREQ 1
OPTI PAS AUTO NOTE LOG 1
STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

### genmesh.dgibi

```

opti echo 1;
opti dime 3 elem tet4;
opti trac psc ftra 'genmesh.ps';
opti rest form 'brainv.msh';
rest form;
list (nbel vol);
list (nbno vol);
lbra = lect 1 pas 1 414932;
lcsf = lect 414933 pas 1 576079;
brain = vol elem lbra;
csf = vol elem lcsf;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
p0 = 0 0 0;
vol2 = vol plus p0;
tol = 1.E-7;
elim tol vol2;
list (nbel vol2);
list (nbno vol2);
skull = enve vol2;
list (nbel skull);
list (nbno skull);
oubl vol2;
fsan = enve csf;
head = brain et csf et skull et fsan;
opti sauv form 'head.msh';
tass head noop;
sauv form head;
trac cach head;
fin;

```

## genmesh\_new.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
brain = 'READ' 'brain.read' 'MESH' 'ELEM';
list (nbno brain);
list (nbel brain);
csf = 'READ' 'csf.read' 'MESH' 'ELEM';
list (nbno csf);
list (nbel csf);
braincsf = brain et csf;
list (nbno braincsf);
list (nbel braincsf);
*opti sauv form 'braincsf.msh';
opti trac psc ftra 'braincsf_mesh.ps';
*sauv form braincsf;
trac cach braincsf;
p0 = 0 0 0;
vol2 = braincsf plus p0;
tol = 1.E-7;
elim tol vol2;
list (nbel vol2);
list (nbno vol2);
skull = enve vol2;
list (nbel skull);
list (nbno skull);
oubl vol2;
fsan = enve csf;
head = brain et csf et skull et fsan;
opti sauv form 'head.msh';
tass head noop;
sauv form head;
trac cach head;
fin;

```

## genmeshl.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
brain = 'READ' 'brain.read' 'MESH' 'ELEM';
list (nbno brain);
list (nbel brain);
csf = 'READ' 'csf.read' 'MESH' 'ELEM';
list (nbno csf);
list (nbel csf);
braincsf = brain et csf;
list (nbno braincsf);
list (nbel braincsf);
tol = 1.E-7;
elim tol braincsf;
opti trac psc ftra 'braincsf_mesh.ps';
trac cach braincsf;
list (nbno braincsf);
list (nbel braincsf);
pskull = chan poil (enve braincsf);
list (nbel pskull);
list (nbno pskull);
head = brain et csf et pskull;
opti sauv form 'headl.msh';
tass head noop;
sauv form head;
trac cach head;
fin;

```

## head46.dgibi

```

opti echo 1;
opti sauv form 'head46.msh';
opti trac psc ftra 'head46_mesh.ps';
opti dime 3 elem cub8;
*****
'DEBPROC' PXPSP8 cent*'POINT' rcub*'FLOTTANT' rsph*'FLOTTANT'
ncub*'ENTIER';
*
*-----
* Generates a mesh of hexahedra consisting of 1/8 of a cube
* surrounded by 1/8 of a sphere.
* The origin of the cube is at point cent and the radius of the cube
* is rcub. The radius of the sphere is rsph (> rcub).
* The number of subdivisions along the cube radius is ncub.
* The cube is oriented along the global axes X, Y, Z and
* the resulting mesh (cube end sphere portion) lies in the first
* octant.
*
* Input :
* -----
* cent : center of the cube/sphere
* rcub : radius (half-side) of the cube
* rsph : radius of the sphere
* ncub : number of subdivisions along the cube radius
*
* Output :
* -----
* hexa : mesh of hexahedra in 3d
* ss : mesh of spherical external surface (quadrilaterals in 3d)
*-----
*
* Longueurs caracteristiques du maillage
*
rsph_c = ((2**0.5)/2)*rsph;
rsph_s = ((3**0.5)/3)*rsph;
P = 0.5*rsph*3.14159;
*
* Points
*
dc = rcub / ncub;
OPTI DENS dc;
*
tol = 0.01*dc;
*
0 = 0. 0. 0.;
Cx = rcub 0. 0.;
Cy = 0. rcub 0.;
Cz = 0. 0. rcub;
Cdb = rcub rcub 0.;
Cdh = rcub rcub rcub;
Czx = rcub 0. rcub;
Cyz = 0. rcub rcub;
*
* Lignes cube
*
OCx = 0 DROIT Cx;
OCy = 0 DROIT Cy;
OCz = 0 DROIT Cz;
CxCzx = Czx DROIT Cx;
CyCyz = Cy DROIT Cyz;
CdbCdh = Cdb DROIT Cdh;
CxCdb = Cx DROIT Cdb;
CyCdb = Cy DROIT Cdb;
CzxCdh = Czx DROIT Cdh;
CyzCdh = Cyz DROIT Cdh;
CzCzx = Cz DROIT Czx;
CzCyz = Cz DROIT Cyz;
*
* Surfaces cube
*
carre_y1 = OCx CxCzx CzCzx OCz DALLER PLAN;
carre_x1 = OCy CyCyz CzCyz OCz DALLER PLAN;
carre_z1 = OCx CxCdb CyCdb OCy DALLER PLAN;
carre_y2 = CyCdb CdbCdh CyzCdh CyCyz DALLER PLAN;
carre_x2 = CxCdb CdbCdh CzxCdh CxCzx DALLER PLAN;
carre_z2 = CzCzx CzxCdh CyzCdh CzCyz DALLER PLAN;
*
* Cube
*
cube = carre_y1 carre_x1 carre_z1 carre_y2 carre_x2 carre_z2 pave;
*
* Points sphere
*
ds = P / (2*ncub);
OPTI DENS ds;
*
ps1 = rsph 0 0;
ps2 = rsph_c rsph_c 0;
ps3 = rsph_s rsph_s rsph_s;
ps4 = rsph_c 0 rsph_c;
ps5 = 0 rsph 0;
ps6 = 0 rsph_c rsph_c;
ps7 = 0 0 rsph;
*
* Lignes sphere
*
ls1 = CERC ncub ps1 0 ps2;
ls2 = CERC ncub ps2 0 ps3;
ls3 = CERC ncub ps3 0 ps4;
ls4 = CERC ncub ps4 0 ps1;
*
ls5 = CERC ncub ps5 0 ps2;
ls6 = CERC ncub ps3 0 ps6;

```

```

ls7 = CERC ncub ps6 0 ps5;
ls8 = CERC ncub ps7 0 ps4;
ls9 = CERC ncub ps6 0 ps7;
*
* Surfaces sphere
*
ss1 = DALL ls1 ls2 ls3 ls4 'SPHE' 0;
ss2 = DALL ls5 ls2 ls6 ls7 'SPHE' 0;
ss3 = DALL ls8 ls3 ls6 ls9 'SPHE' 0;
*
ss = ss1 et ss2 et ss3;
elim tol ss;
*
* Volumes sphere
*
vs1 = carre_x2 VOLU ss1;
vs2 = carre_y2 VOLU ss2;
vs3 = carre_z2 VOLU ss3;
*
hexa = cube et vs1 et vs2 et vs3;
elim tol hexa;
DEPL hexa 'PLUS' cent;
*
'FINPROC' hexa ss;
=====
p0 = 0 0 0;
px = 1 0 0;
py = 0 1 0;
pz = 0 0 1;
rcub = 20.E-3;
rsph = 60.E-3;
ncub = 20;
*
* Brain
*
bra1 sbra1 = pxsph8 p0 rcub rsph ncub;
bra2 = bra1 tour 90 p0 pz;
bra3 = bra2 tour 90 p0 pz;
bra4 = bra3 tour 90 p0 pz;
braa = bra1 et bra2 et bra3 et bra4;
tol = (0.001*rcub) / ncub;
elim tol braa;
brab = braa tour 180.E0 p0 px;
brain = braa et brab;
elim tol brain;
*
* CSF (cerebro-spinal fluid)
*
scsf1 = sbra1 plus p0;
scsf2 = scsf1 homo (65.0 / 60.0) p0;
csf1 = scsf1 VOLU scsf2;
csf2 = csf1 tour 90 p0 pz;
csf3 = csf2 tour 90 p0 pz;
csf4 = csf3 tour 90 p0 pz;
csfa = csf1 et csf2 et csf3 et csf4;
elim tol csfa;
csfb = csfa tour 180.E0 p0 px;
csf = csfa et csfb;
elim tol csf;
*
* Skull
*
ssku1 = scsf2 plus p0;
ssku2 = ssku1 homo (70.0 / 65.0) p0;
sku1 = ssku1 VOLU ssku2;
sku2 = sku1 tour 90 p0 pz;
sku3 = sku2 tour 90 p0 pz;
sku4 = sku3 tour 90 p0 pz;
skua = sku1 et sku2 et sku3 et sku4;
elim tol skua;
skub = skua tour 180.E0 p0 px;
skull = skua et skub;
elim tol skull;
ssku2b = ssku2 tour 90 p0 pz;
ssku2c = ssku2b tour 90 p0 pz;
ssku2d = ssku2c tour 90 p0 pz;
ssku2u = ssku2 et ssku2b et ssku2c et ssku2d;
ssku2v = ssku2u tour 180.E0 p0 px;
ssku2t = ssku2u et ssku2v;
elim tol (ssku2t et skull);
*
* PMATs for contact on external surface of the skull
*
psext = chan poi1 ssku2t;
n = nbel psext;
i = 0;
ni = 0;
repe loop n;
  i = i + 1;
  ei = psext elem i;
  pi = ei poin 1;
  xi yi zi = coor pi;
  si (>eg xi 45.E-3);
  ni = ni + 1;
  si (ega ni 1);
  pscon = ei;
  sinon;
  pscon = pscon et ei;
  finsi;
finsi;
fin loop;
*
* Pinballs for contact on external surface of the brain and on

```

```

* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Frontal region.
*
braexp = chan poi1 (enve brain);
xbr = coor 1 braexp;
bx = xbr poin supe 45.E-3;
brapin = brain elem appu larg bx nove;
list (nbel brapin);
*
skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
xsk = coor 1 skuint;
sx = xsk poin supe 50.E-3;
skupin = skull elem appu larg sx nove;
list (nbel skupin);
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Neck region.
*
*braexp = chan poi1 (enve brain);
*xbr = coor 1 braexp;
*bx2 = xbr poin infe -45.E-3;
brapi2 = brain elem appu larg bx2 nove;
list (nbel brapi2);
*
*skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
*xsk = coor 1 skuint;
*sx2 = xsk poin infe -50.E-3;
skupi2 = skull elem appu larg sx2 nove;
list (nbel skupi2);
*
* Obstacle
*
o1 = 72.E-3 -40.E-3 -40.E-3;
o2 = 72.E-3 -40.E-3 40.E-3;
o3 = 72.E-3 40.E-3 40.E-3;
o4 = 72.E-3 40.E-3 -40.E-3;
c1 = o1 d 40 o2;
c2 = o2 d 40 o3;
c3 = o3 d 40 o4;
c4 = o4 d 40 o1;
obst = dall c1 c2 c3 c4 plan;
*
* Assemble complete mesh
*
fsa1 = csf poin sphe p0 (p0 plus (60.E-3 0 0)) 1.E-5;
fsa2 = csf poin sphe p0 (p0 plus (65.E-3 0 0)) 1.E-5;
fsan = fsa1 et fsa2;
mesh = brain et csf et skull et obst et pscon et fsan et brapin et
      skupin et brapi2 et skupi2;
tass mesh noop;
sauv form mesh;
trac cach qual braa;
trac cach qual brab;
trac cach qual brain;
trac cach qual csfa;
trac cach qual csfb;
trac cach qual csf;
trac cach qual skua;
trac cach qual skub;
trac cach qual skull;
trac cach qual obst;
trac cach qual (skull et pscon);
trac cach qual (brapin et skupin);
trac cach qual (brapi2 et skupi2);
trac cach qual (brapin et skupin et brapi2 et skupi2);
trac cach qual mesh;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
list (nbel skull);
list (nbno skull);
list (nbel obst);
list (nbno obst);
list (nbel pscon);
list (nbno pscon);
list (nbel mesh);
list (nbno mesh);
*
fin;

```

## head46.epx

```

HEAD46
ECHO
!CONV WIN
CAST mesh
TRID ALE
DIME NALE 6402 NBLE 26284 TERM
GEOM CUBE brain CUBE csf CUB8 skull Q4GS obst PMAT pscon TERM
COMP EPAI 2.0E-3 LECT obst TERM
      2.0E-3 LECT pscon TERM
GROU 6 'bray1' LECT brain TERM COND YB LT 0
      'bray2' LECT brain DIFF bray1 TERM
      'csfy1' LECT csf TERM COND YB LT 0
      'csfy2' LECT csf DIFF csfy1 TERM
      'skuy1' LECT skull TERM COND YB LT 0
      'skuy2' LECT skull DIFF skuy1 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM

```

```

JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
GRIL LAGR LECT brain skull obst pscon TERM
EULE LECT fsan TERM
AUTO AUTR
MATE HYPE TYPE 1 RO 1040 ! Mooney-Rivlin
      BULK 128E+06 ! This is in Pa
      CO1 185.0 ! These are in Pa
      CO2 46.3
      LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM
OPTI PINS EQVL ASN GRID DPIN 1.01
LINK COUP SPLT NONE
BLOQ 123456 LECT obst TERM
FSA LECT fsan TERM
PINB BODY MLEV 0 LECT obst TERM
      BODY DIAM 2.0E-3 LECT pscon TERM
      BODY DIAM 5.0E-3 LECT brapin TERM
      BODY DIAM 5.0E-3 LECT skupin TERM
      BODY DIAM 5.0E-3 LECT brapi2 TERM
      BODY DIAM 5.0E-3 LECT skupi2 TERM
EXCL PAIR 1 3
      PAIR 2 3
      PAIR 1 4
      PAIR 2 4
      PAIR 1 5
      PAIR 2 5
      PAIR 1 6
      PAIR 2 6
      PAIR 3 5
      PAIR 3 6
      PAIR 4 5
      PAIR 4 6
INIT VITE 1 10.0 LECT brain csf skull TERM
VITG 1 10.0 LECT brain csf skull TERM
ECRI TFRE 1.E-3
NOPO NOEL
FICH SPLI ALIC TFRE 1.E-5
OPTI NOTE LOG 1
NOPR PLIN
CSTA 0.5
NOCR LECT obst pscon TERM
REZO MVRE MODU VFAC 5.0
CALC TINI 0.0 TFIN 4.E-3
FIN

```

## head46v.epx

```

HEAD46Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head46.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
VIEW 0.00000E+00 1.00000E+00 2.05103E-10
RIGH 1.00000E+00 0.00000E+00 0.00000E+00
UP 0.00000E+00 -2.05103E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
! von Mises equivalent stress
ISO FILL FIEL ECR0 2 SCAL USER PROG 0.5E6 PAS 0.5E6 7.0E6 TERM
SUPP LECT brain TERM
!ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
! SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head46w.epx

```

HEAD46W
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head46.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE -2.00124E-01 -3.20482E-01 -3.73257E-08
! Q 6.83013E-01 6.83013E-01 -1.83013E-01 -1.83013E-01
VIEW 5.00000E-01 8.66025E-01 1.77623E-10
RIGH 8.66025E-01 -5.00000E-01 -1.02552E-10
UP 5.55112E-17 -2.05101E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 4.76748E-03 3.43995E-02 -3.72529E-08
!RSPHERE: 1.02446E-01
!RADIUS : 4.09782E-01
!ASPECT : 1.00000E+00
!NEAR : 3.07337E-01
!FAR : 6.14673E-01
SCEN OBJE USLM LECT bray2 TERM
      DHAS CGLA
      GEOM NAVI FREE
      FACE SBAC
      LINE HEOU SSHA SFRE
      REFE FRAM
      PINB CDES
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 77 FPS 25 KFRE 5 COMP -1
OBJE LECT bray2 skuy2 obst TERM REND
GOTR LOOP 75 OFFS FICH AVI CONT NOCL
OBJE LECT bray2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT bray2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head46z.epx

```

HEAD46Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head46.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
VIEW 0.00000E+00 1.00000E+00 2.05103E-10
RIGH 1.00000E+00 0.00000E+00 0.00000E+00
UP 0.00000E+00 -2.05103E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
!ISO FILL FIEL ECR0 1 SCAL USER PROG -6.E5 PAS 0.5E5 0.5E5 TERM
! SUPP LECT brain TERM
! COSC ICOL
ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 77 FPS 25 KFRE 5 COMP -1
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 75 OFFS FICH AVI CONT NOCL
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head47.dgibi

```

opti echo 1;
opti sauv form 'head47.msh';
opti trac psc ftra 'head47_mesh.ps';
opti dime 3 elem cub8;
=====
'DEBPROC' PKSPH8 cent*'POINT' rcub*'FLOTTANT' rsph*'FLOTTANT'

```



```

ncub*'ENTIER';
*
*-----
* Generates a mesh of hexahedra consisting of 1/8 of a cube
* surrounded by 1/8 of a sphere.
* The origin of the cube is at point cent and the radius of the cube
* is rcub. The radius of the sphere is rsph (> rcub).
* The number of subdivisions along the cube radius is ncub.
* The cube is oriented along the global axes X, Y, Z and
* the resulting mesh (cube end sphere portion) lies in the first
* octant.
*
* Input :
* -----
* cent : center of the cube/sphere
* rcub : radius (half-side) of the cube
* rsph : radius of the sphere
* ncub : number of subdivisions along the cube radius
*
* Output :
* -----
* hexa : mesh of hexahedra in 3d
* ss : mesh of spherical external surface (quadrilaterals in 3d)
*-----
* Longueurs caracteristiques du maillage
*
rsph_c = ((2**0.5)/2)*rsph;
rsph_s = ((3**0.5)/3)*rsph;
P = 0.5*rsph*3.14159;
*
* Points
*
dc = rcub / ncub;
OPTI DENS dc;
*
tol = 0.01*dc;
*
O = 0. 0. 0.;
Cx = rcub 0. 0.;
Cy = 0. rcub 0.;
Cz = 0. 0. rcub;
Cdb = rcub rcub 0.;
Cdh = rcub rcub rcub;
Czx = rcub 0. rcub;
Cyz = 0. rcub rcub;
*
* Lignes cube
*
OCx = 0 DROIT Cx;
OCy = 0 DROIT Cy;
OCz = 0 DROIT Cz;
CxCxz = Cxz DROIT Cx;
CyCyz = Cy DROIT Cyz;
CdbCdh = Cdb DROIT Cdh;
CxCdb = Cx DROIT Cdb;
CyCdb = Cy DROIT Cdb;
CzxCdh = Cxz DROIT Cdh;
CyzCdh = Cyz DROIT Cdh;
CzCzx = Cz DROIT Cxz;
CzCyz = Cz DROIT Cyz;
*
* Surfaces cube
*
carre_y1 = OCx CxCxz CzCzx OCz DALLER PLAN;
carre_x1 = OCy CyCyz CzCyz OCz DALLER PLAN;
carre_z1 = OCx CxCdb CyCdb OCy DALLER PLAN;
carre_y2 = CyCdb CdbCdh CyzCdh CyCyz DALLER PLAN;
carre_x2 = CxCdb CdbCdh CzxCdh CxCxz DALLER PLAN;
carre_z2 = CzCzx CzxCdh CyzCdh CzCyz DALLER PLAN;
*
* Cube
*
cube = carre_y1 carre_x1 carre_z1 carre_y2 carre_x2 carre_z2 pave;
*
* Points sphere
*
ds = P / (2*ncub);
OPTI DENS ds;
*
ps1 = rsph 0 0;
ps2 = rsph_c rsph_c 0;
ps3 = rsph_s rsph_s rsph_s;
ps4 = rsph_c 0 rsph_c;
ps5 = 0 rsph 0;
ps6 = 0 rsph_c rsph_c;
ps7 = 0 0 rsph;
*
* Lignes sphere
*
ls1 = CERC ncub ps1 0 ps2;
ls2 = CERC ncub ps2 0 ps3;
ls3 = CERC ncub ps3 0 ps4;
ls4 = CERC ncub ps4 0 ps1;
*
ls5 = CERC ncub ps5 0 ps2;
ls6 = CERC ncub ps3 0 ps6;
ls7 = CERC ncub ps6 0 ps5;
ls8 = CERC ncub ps7 0 ps4;
ls9 = CERC ncub ps6 0 ps7;
*
* Surfaces sphere
*
ss1 = DALL ls1 ls2 ls3 ls4 'SPHE' 0;
ss2 = DALL ls5 ls2 ls6 ls7 'SPHE' 0;
ss3 = DALL ls8 ls3 ls6 ls9 'SPHE' 0;
*
ss = ss1 et ss2 et ss3;
elim tol ss;
*
* Volumes sphere
*
vs1 = carre_x2 VOLU ss1;
vs2 = carre_y2 VOLU ss2;
vs3 = carre_z2 VOLU ss3;
*
hexa = cube et vs1 et vs2 et vs3;
elim tol hexa;
DEPL hexa 'PLUS' cent;
*
'FINPROC' hexa ss;
*-----
p0 = 0 0 0;
px = 1 0 0;
py = 0 1 0;
pz = 0 0 1;
rcub = 20.E-3;
rsph = 60.E-3;
ncub = 20;
*
* Brain
*
bra1 sbra1 = pxsph8 p0 rcub rsph ncub;
bra2 = bra1 tour 90 p0 pz;
bra3 = bra2 tour 90 p0 pz;
bra4 = bra3 tour 90 p0 pz;
braa = bra1 et bra2 et bra3 et bra4;
tol = (0.001*rcub) / ncub;
elim tol braa;
brab = braa tour 180.E0 p0 px;
brain = braa et brab;
elim tol brain;
*
* CSF (cerebro-spinal fluid)
*
scsf1 = sbra1 plus p0;
scsf2 = scsf1 homo (65.0 / 60.0) p0;
csf1 = scsf1 VOLU scsf2;
csf2 = csf1 tour 90 p0 pz;
csf3 = csf2 tour 90 p0 pz;
csf4 = csf3 tour 90 p0 pz;
csfa = csf1 et csf2 et csf3 et csf4;
elim tol csfa;
csfb = csfa tour 180.E0 p0 px;
csf = csfa et csfb;
elim tol csf;
*
* Skull
*
ssku1 = scsf2 plus p0;
ssku2 = ssku1 homo (70.0 / 65.0) p0;
sku1 = ssku1 VOLU ssku2;
sku2 = sku1 tour 90 p0 pz;
sku3 = sku2 tour 90 p0 pz;
sku4 = sku3 tour 90 p0 pz;
skua = sku1 et sku2 et sku3 et sku4;
elim tol skua;
skub = skua tour 180.E0 p0 px;
skull = skua et skub;
elim tol skull;
ssku2b = ssku2 tour 90 p0 pz;
ssku2c = ssku2b tour 90 p0 pz;
ssku2d = ssku2c tour 90 p0 pz;
ssku2u = ssku2 et ssku2b et ssku2c et ssku2d;
ssku2v = ssku2u tour 180.E0 p0 px;
ssku2t = ssku2u et ssku2v;
elim tol (ssku2t et skull);
*
* PMATs for contact on external surface of the skull
*
psext = chan poi1 ssku2t;
n = nbel psext;
i = 0;
ni = 0;
repe loop n;
i = i + 1;
ei = psext elem i;
pi = ei poin i;
xi yi zi = coor pi;
si (>eg xi 45.E-3);
ni = ni + 1;
s1 (ega ni 1);
pscon = ei;
sinon;
pscon = pscon et ei;
finsi;
finsi;
fin loop;
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Frontal region.
*
braexp = chan poi1 (enve brain);
xbr = coor 1 braexp;
bx = xbr poin supe 45.E-3;

```

```

brapin = brain elem appu larg bx nove;
list (nbel brapin);
*
skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
xsk = coor 1 skuint;
sx = xsk poin supe 50.E-3;
skupin = skull elem appu larg sx nove;
list (nbel skupin);
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Neck region.
*
*braexp = chan poi1 (enve brain);
*xbr = coor 1 braexp;
bx2 = xbr poin infe -45.E-3;
brapi2 = brain elem appu larg bx2 nove;
list (nbel brapi2);
*
*skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
*xsk = coor 1 skuint;
*sx2 = xsk poin infe -50.E-3;
*skupi2 = skull elem appu larg sx2 nove;
list (nbel skupi2);
*
* Obstacle
*
o1 = 72.E-3 -40.E-3 -40.E-3;
o2 = 72.E-3 -40.E-3 40.E-3;
o3 = 72.E-3 40.E-3 40.E-3;
o4 = 72.E-3 40.E-3 -40.E-3;
c1 = o1 d 40 o2;
c2 = o2 d 40 o3;
c3 = o3 d 40 o4;
c4 = o4 d 40 o1;
obst = dall c1 c2 c3 c4 plan;
*
* Assemble complete mesh
*
fsa1 = csf poin sphe p0 (p0 plus (60.E-3 0 0)) 1.E-5;
fsa2 = csf poin sphe p0 (p0 plus (65.E-3 0 0)) 1.E-5;
fsan = fsa1 et fsa2;
mesh = brain et csf et skull et obst et pscon et fsan et brapin et
      skupin et brapi2 et skupi2;
tass mesh noop;
sauv form mesh;
trac cach qual braa;
trac cach qual brab;
trac cach qual brain;
trac cach qual csfa;
trac cach qual csfb;
trac cach qual csf;
trac cach qual skua;
trac cach qual skub;
trac cach qual skull;
trac cach qual obst;
trac cach qual (skull et pscon);
trac cach qual (brapin et skupin);
trac cach qual (brapi2 et skupi2);
trac cach qual (brapin et skupin et brapi2 et skupi2);
trac cach qual mesh;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
list (nbel skull);
list (nbno skull);
list (nbel obst);
list (nbno obst);
list (nbel pscon);
list (nbno pscon);
list (nbel mesh);
list (nbno mesh);
*
fin;

```

## head47.epx

```

HEAD47
ECHO
!CONV WIN
CAST mesh
TRID ALE
DIME NALE 6402 NBLE 26284 TERM
GEOM CUBE brain CUBE csf CUB8 skull Q4GS obst PMAT pscon TERM
COMP EPAI 2.0E-3 LECT obst TERM
      2.0E-3 LECT pscon TERM
GROU 6 'bray1' LECT brain TERM COND YB LT 0
      'bray2' LECT brain DIFF bray1 TERM
      'csfy1' LECT csf TERM COND YB LT 0
      'csfy2' LECT csf DIFF csfy1 TERM
      'skuy1' LECT skull TERM COND YB LT 0
      'skuy2' LECT skull DIFF skuy1 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
GRIL LAGR LECT brain skull obst pscon TERM
EULE LECT fsan TERM
AUTO AUTR

```

```

MATE HYPE TYPE 4 RO 1040 ! Ogden (new formulation)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM
OPTI PINS EQVL ASN GRID DPIN 1.01
LINK COUP SPLT NONE
BLOQ 123456 LECT obst TERM
FSA LECT fsan TERM
PINB BODY MLEV 0 LECT obst TERM
      BODY DIAM 2.0E-3 LECT pscon TERM
      BODY DIAM 5.0E-3 LECT brapin TERM
      BODY DIAM 5.0E-3 LECT skupin TERM
      BODY DIAM 5.0E-3 LECT brapi2 TERM
      BODY DIAM 5.0E-3 LECT skupi2 TERM
EXCL PAIR 1 3
      PAIR 2 3
      PAIR 1 4
      PAIR 2 4
      PAIR 1 5
      PAIR 2 5
      PAIR 1 6
      PAIR 2 6
      PAIR 3 5
      PAIR 3 6
      PAIR 4 5
      PAIR 4 6
INIT VITE 1 10.0 LECT brain csf skull TERM
VITG 1 10.0 LECT brain csf skull TERM
ECRI TFRE 1.E-3
      NOPO NOEL
      FICH SPLI ALIC TFRE 1.E-5
OPTI NOTE LOG 1
      NOPR PLIN
      CSTA 0.5
      NOCR LECT obst pscon TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0.0 TFIN 4.E-3
FIN

```

## head47v.epx

```

HEAD47Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head47.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
      ! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
      VIEW 0.00000E+00 1.00000E+00 2.05103E-10
      RIGH 1.00000E+00 0.00000E+00 0.00000E+00
      UP 0.00000E+00 -2.05103E-10 1.00000E+00
      FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
! von Mises equivalent stress
ISO FILL FIEL ECR0 2 SCAL USER PROG 0.5E6 PAS 0.5E6 7.0E6 TERM
SUPP LECT brain TERM
!ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
! SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTA 401 FPS 25 KFRE 5 COMP -1
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENPLAY
FIN

```

## head47w.epx

```

HEAD47W
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head47.ali' GARD PSCR

```

```

COMP COUL BLAN LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE -2.00124E-01 -3.20482E-01 -3.73257E-08
! Q 6.83013E-01 6.83013E-01 -1.83013E-01 -1.83013E-01
VIEW 5.00000E-01 8.66025E-01 1.77623E-10
RIGH 8.66025E-01 -5.00000E-01 -1.02552E-10
UP 5.55112E-17 -2.05101E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 4.76748E-03 3.43995E-02 -3.72529E-08
!RSPHERE: 1.02446E-01
!RADIUS : 4.09782E-01
!ASPECT : 1.00000E+00
!NEAR : 3.07337E-01
!FAR : 6.14673E-01
SCEN OBJE USLM LECT bray2 TERM
DHAS CGLA
GEOM NAVI FREE
FACE SBAC
LINE HEOU SSHA SFRE
REFE FRAM
PINB CDES
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
OBJE LECT bray2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
OBJE LECT bray2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT bray2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head47z.epx

```

HEAD47Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head47.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
VIEW 0.00000E+00 1.00000E+00 2.05103E-10
RIGH 1.00000E+00 0.00000E+00 0.00000E+00
UP 0.00000E+00 -2.05103E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
!ISO FILL FIEL ECR0 1 SCAL USER PROG -6.E5 PAS 0.5E5 0.5E5 TERM
! SUPP LECT brain TERM
! COSC ICOL
ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head48.dgibi

```

opti echo 1;
opti sauv form 'head48.msh';
opti trac psc ftra 'head48_mesh.ps';
opti dime 3 elem cub8;
*****
'DEBPROC' PXSPH8 cent*'POINT' rcub*'FLOTTANT' rsph*'FLOTTANT'
      ncub*'ENTIER';
*
*****
* Generates a mesh of hexahedra consisting of 1/8 of a cube

```

```

* surrounded by 1/8 of a sphere.
* The origin of the cube is at point cent and the radius of the cube
* is rcub. The radius of the sphere is rsph (> rcub).
* The number of subdivisions along the cube radius is ncub.
* The cube is oriented along the global axes X, Y, Z and
* the resulting mesh (cube end sphere portion) lies in the first
* octant.
*
* Input :
* -----
* cent : center of the cube/sphere
* rcub : radius (half-side) of the cube
* rsph : radius of the sphere
* ncub : number of subdivisions along the cube radius
*
* Output :
* -----
* hexa : mesh of hexahedra in 3d
* ss : mesh of spherical external surface (quadrilaterals in 3d)
*-----*
*
* Longueurs caracteristiques du maillage
*
rsph_c = ((2**0.5)/2)*rsph;
rsph_s = ((3**0.5)/3)*rsph;
P = 0.5*rsph*3.14159;
*
* Points
*
dc = rcub / ncub;
OPTI DENS dc;
*
tol = 0.01*dc;
*
0 = 0. 0. 0.;
Cx = rcub 0. 0.;
Cy = 0. rcub 0.;
Cz = 0. 0. rcub;
Cdb = rcub rcub 0.;
Cdh = rcub rcub rcub;
Czx = rcub 0. rcub;
Cyz = 0. rcub rcub;
*
* Lignes cube
*
OCx = 0 DROIT Cx;
OCy = 0 DROIT Cy;
OCz = 0 DROIT Cz;
CxCzx = Czx DROIT Cx;
CyCyz = Cy DROIT Cyz;
CdbCdh = Cdb DROIT Cdh;
CxCdb = Cx DROIT Cdb;
CyCdb = Cy DROIT Cdb;
CxCdh = Czx DROIT Cdh;
CyzCdh = Cyz DROIT Cdh;
CzCzx = Cz DROIT Czx;
CzCyz = Cz DROIT Cyz;
*
* Surfaces cube
*
carre_y1 = OCx CxCzx CzCzx OCz DALLER PLAN;
carre_x1 = OCy CyCyz CzCyz OCz DALLER PLAN;
carre_z1 = OCx CxCdb CyCdb OCy DALLER PLAN;
carre_y2 = CyCdb CdbCdh CyzCdh CyCyz DALLER PLAN;
carre_x2 = CxCdb CdbCdh CxCdh CxCzx DALLER PLAN;
carre_z2 = CzCzx CxCdh CyzCdh CzCyz DALLER PLAN;
*
* Cube
*
cube = carre_y1 carre_x1 carre_z1 carre_y2 carre_x2 carre_z2 pave;
*
* Points sphere
*
ds = P / (2*ncub);
OPTI DENS ds;
*
ps1 = rsph 0 0;
ps2 = rsph_c rsph_c 0;
ps3 = rsph_s rsph_s rsph_s;
ps4 = rsph_c 0 rsph_c;
ps5 = 0 rsph 0;
ps6 = 0 rsph_c rsph_c;
ps7 = 0 0 rsph;
*
* Lignes sphere
*
ls1 = CERC ncub ps1 0 ps2;
ls2 = CERC ncub ps2 0 ps3;
ls3 = CERC ncub ps3 0 ps4;
ls4 = CERC ncub ps4 0 ps1;
*
ls5 = CERC ncub ps5 0 ps2;
ls6 = CERC ncub ps3 0 ps6;
ls7 = CERC ncub ps6 0 ps5;
ls8 = CERC ncub ps7 0 ps4;
ls9 = CERC ncub ps6 0 ps7;
*
* Surfaces sphere
*
ss1 = DALL ls1 ls2 ls3 ls4 'SPHE' 0;
ss2 = DALL ls5 ls2 ls6 ls7 'SPHE' 0;
ss3 = DALL ls8 ls3 ls6 ls9 'SPHE' 0;
*

```

```

ss = ss1 et ss2 et ss3;
elim tol ss;
*
* Volumes sphere
*
vs1 = carre_x2 VOLU ss1;
vs2 = carre_y2 VOLU ss2;
vs3 = carre_z2 VOLU ss3;
*
hexa = cube et vs1 et vs2 et vs3;
elim tol hexa;
DEPL hexa 'PLUS' cent;
*
'FINPROC' hexa ss;
*
p0 = 0 0 0;
px = 1 0 0;
py = 0 1 0;
pz = 0 0 1;
rcub = 20.E-3;
rsph = 60.E-3;
ncub = 20;
*
* Brain
*
bra1 sbra1 = pxsph8 p0 rcub rsph ncub;
bra2 = bra1 tour 90 p0 pz;
bra3 = bra2 tour 90 p0 pz;
bra4 = bra3 tour 90 p0 pz;
braa = bra1 et bra2 et bra3 et bra4;
tol = (0.001*rcub) / ncub;
elim tol braa;
brab = braa tour 180.E0 p0 px;
brain = braa et brab;
elim tol brain;
*
* CSF (cerebro-spinal fluid)
*
scsf1 = sbra1 plus p0;
scsf2 = scsf1 homo (65.0 / 60.0) p0;
csf1 = scsf1 VOLU scsf2;
csf2 = csf1 tour 90 p0 pz;
csf3 = csf2 tour 90 p0 pz;
csf4 = csf3 tour 90 p0 pz;
csfa = csf1 et csf2 et csf3 et csf4;
elim tol csfa;
csfb = csfa tour 180.E0 p0 px;
csf = csfa et csfb;
elim tol csf;
*
* Skull
*
ssku1 = scsf2 plus p0;
ssku2 = ssku1 homo (70.0 / 65.0) p0;
sku1 = ssku1 VOLU ssku2;
sku2 = sku1 tour 90 p0 pz;
sku3 = sku2 tour 90 p0 pz;
sku4 = sku3 tour 90 p0 pz;
skua = sku1 et sku2 et sku3 et sku4;
elim tol skua;
skub = skua tour 180.E0 p0 px;
skull = skua et skub;
elim tol skull;
ssku2b = ssku2 tour 90 p0 pz;
ssku2c = ssku2b tour 90 p0 pz;
ssku2d = ssku2c tour 90 p0 pz;
ssku2u = ssku2 et ssku2b et ssku2c et ssku2d;
ssku2v = ssku2u tour 180.E0 p0 px;
ssku2t = ssku2u et ssku2v;
elim tol (ssku2t et skull);
*
* PMATs for contact on external surface of the skull
*
psext = chan poi1 ssku2t;
n = nbel psext;
i = 0;
ni = 0;
repe loop n;
  i = i + 1;
  ei = psext elem i;
  pi = ei poin i;
  xi yi zi = coor pi;
  si (>eg xi 45.E-3);
  ni = ni + 1;
  si (ega ni 1);
  pscon = ei;
  sinon;
  pscon = pscon et ei;
  fins1;
  fin loop;
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Frontal region.
*
braexp = chan poi1 (enve brain);
xbr = coor 1 braexp;
bx = xbr poin supe 45.E-3;
brapin = brain elem appu larg bx nove;
list (nbel brapin);
*
skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;

```

```

xsk = coor 1 skuint;
sx = xsk poin supe 50.E-3;
skupin = skull elem appu larg sx nove;
list (nbel skupin);
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Neck region.
*
*braexp = chan poi1 (enve brain);
*xbr = coor 1 braexp;
bx2 = xbr poin infe -45.E-3;
brapi2 = brain elem appu larg bx2 nove;
list (nbel brapi2);
*
*skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
*xsk = coor 1 skuint;
sx2 = xsk poin infe -50.E-3;
skupi2 = skull elem appu larg sx2 nove;
list (nbel skupi2);
*
* Obstacle
*
o1 = 72.E-3 -40.E-3 -40.E-3;
o2 = 72.E-3 -40.E-3 40.E-3;
o3 = 72.E-3 40.E-3 40.E-3;
o4 = 72.E-3 40.E-3 -40.E-3;
c1 = o1 d 40 o2;
c2 = o2 d 40 o3;
c3 = o3 d 40 o4;
c4 = o4 d 40 o1;
obst = dall c1 c2 c3 c4 plan;
*
* Assemble complete mesh
*
fsa1 = csf poin sphe p0 (p0 plus (60.E-3 0 0)) 1.E-5;
fsa2 = csf poin sphe p0 (p0 plus (65.E-3 0 0)) 1.E-5;
fsan = fsa1 et fsa2;
mesh = brain et csf et skull et obst et pscon et fsan et brapin et
      skupin et brapi2 et skupi2;
tass mesh noop;
sauv form mesh;
trac cach qual braa;
trac cach qual brab;
trac cach qual brain;
trac cach qual csfa;
trac cach qual csfb;
trac cach qual csf;
trac cach qual skua;
trac cach qual skub;
trac cach qual skull;
trac cach qual obst;
trac cach qual (skull et pscon);
trac cach qual (brapin et skupin);
trac cach qual (brapi2 et skupi2);
trac cach qual (brapin et skupin et brapi2 et skupi2);
trac cach qual mesh;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
list (nbel skull);
list (nbno skull);
list (nbel obst);
list (nbno obst);
list (nbel pscon);
list (nbno pscon);
list (nbel mesh);
list (nbno mesh);
*
fin;

```

## head48.epx

```

HEAD48
ECHO
! CONV WIN
CAST mesh
TRID ALE
DIME NALE 6402 NBLE 26284 TERM
GEMO CUBE brain CUBE csf CUB8 skull Q4GS obst PMAT pscon TERM
COMP EPAI 2.0E-3 LECT obst TERM
      2.0E-3 LECT pscon TERM
      GROU 6 'bray1' LECT brain TERM COND YB LT 0
      'bray2' LECT brain DIFF bray1 TERM
      'csfy1' LECT csf TERM COND YB LT 0
      'csfy2' LECT csf DIFF csfy1 TERM
      'skuy1' LECT skull TERM COND YB LT 0
      'skuy2' LECT skull DIFF skuy1 TERM
      COUL GR50 LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
GRIL LAGR LECT brain skull obst pscon TERM
      EULE LECT fsan TERM
      AUTO AUTR
MATE HYPE TYPE 4 RO 1040 ! Ogden (new formulation)
      BULK 128E+06 ! This is in Pa
      CO1 23.5 ! This is alpha_1
      CO2 -4.0 ! This is alpha_2

```

```

      CO5      8.32E-4 ! This is mu_1
      CO6     -0.41  ! This is mu_2
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM
OPTI PINS EQVL ASN GRID DPIN 1.01
LINK COUP SPLT NONE
BLOQ 123456 LECT obst TERM
FSA LECT fsan TERM
PINB BODY MLEV 0 LECT obst TERM
BODY DIAM 2.0E-3 LECT pscon TERM
BODY DIAM 5.0E-3 LECT brapin TERM
BODY DIAM 5.0E-3 LECT skupin TERM
BODY DIAM 5.0E-3 LECT brapi2 TERM
BODY DIAM 5.0E-3 LECT skupi2 TERM
EXCL PAIR 1 3
      PAIR 2 3
      PAIR 1 4
      PAIR 2 4
      PAIR 1 5
      PAIR 2 5
      PAIR 1 6
      PAIR 2 6
      PAIR 3 5
      PAIR 3 6
      PAIR 4 5
      PAIR 4 6
INIT VITE 1 10.0 LECT brain csf skull TERM
VITG 1 10.0 LECT brain csf skull TERM
ECRI TPRE 1.E-3
      NOPO NOEL
      FICH SPLI ALIC TPRE 1.E-5
OPTI NOTE LOG 1
      NOPR PLIN
      CSTA 0.5
      NOCR LECT obst pscon TERM
      REZO MVRE MODU VFAC 5.0
CALC TINI 0.0 TFIN 4.E-3
FIN

```

## head48v.epx

```

HEAD48Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head48.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
VIEW 0.00000E+00 1.00000E+00 2.05103E-10
RIGH 1.00000E+00 0.00000E+00 0.00000E+00
UP 0.00000E+00 -2.05103E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
! von Mises equivalent stress
ISO FILL FIEL ECR0 2 SCAL USER PROG 0.5E6 PAS 0.5E6 7.0E6 TERM
SUPP LECT brain TERM
!ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
! SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head48w.epx

```

HEAD48W
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head48.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM

```

```

TURQ LECT csf TERM
JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE -2.00124E-01 -3.20482E-01 -3.73257E-08
! Q 6.83013E-01 6.83013E-01 -1.83013E-01 -1.83013E-01
VIEW 5.00000E-01 8.66025E-01 1.77623E-10
RIGH 8.66025E-01 -5.00000E-01 -1.02552E-10
UP 5.55112E-17 -2.05101E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 4.76748E-03 3.43995E-02 -3.72529E-08
!RSPHERE: 1.02446E-01
!RADIUS : 4.09782E-01
!ASPECT : 1.00000E+00
!NEAR : 3.07337E-01
!FAR : 6.14673E-01
SCEN OBJE USLM LECT bray2 TERM
      DHAS CGLA
      GEOM NAVI FREE
      FACE SBAC
      LINE HEOU SSHA SFRE
      REFE FRAM
      PINB CDES
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
      OBJE LECT bray2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
      OBJE LECT bray2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
      OBJE LECT bray2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head48z.epx

```

HEAD48Z
ECHO
CONV WIN
OPTI PRIN
RESU SPLI ALIC 'head48.ali' GARD PSCR
COMP COUL BLAN LECT brain TERM
      TURQ LECT csf TERM
      JAUN LECT skull TERM
      VERT LECT obst TERM
      ROUG LECT pscon TERM
SORT VISU NSTO 1
PLAY
CAME 1 EYE 6.17977E-03 -4.19304E-01 -8.60007E-11
! Q 7.07107E-01 7.07107E-01 0.00000E+00 0.00000E+00
VIEW 0.00000E+00 1.00000E+00 2.05103E-10
RIGH 1.00000E+00 0.00000E+00 0.00000E+00
UP 0.00000E+00 -2.05103E-10 1.00000E+00
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 6.17977E-03 0.00000E+00 0.00000E+00
!RSPHERE: 1.19801E-01
!RADIUS : 4.19304E-01
!ASPECT : 1.00000E+00
!NEAR : 2.99503E-01
!FAR : 6.58906E-01
SCEN GEOM NAVI FREE
!VECT SCCO FIEL VITE SCAL USER PROG 4.5 PAS 0.5 11.0 TERM SIVE
!ISO FILL FIEL ECR0 1 SCAL USER PROG -6.E5 PAS 0.5E5 0.5E5 TERM
! SUPP LECT brain TERM
! COSC ICOL
ISO FILL FIEL ECR0 1 SCAL USER PROG 1.E5 PAS 1.E5 14.E5 TERM
SUPP LECT csf TERM
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTO 401 FPS 25 KFRE 5 COMP -1
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GOTR LOOP 399 OFFS FICH AVI CONT NOCL
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
GO
TRAC OFFS FICH AVI CONT
      OBJE LECT bray2 csfy2 skuy2 obst TERM REND
ENDPLAY
FIN

```

## head55.dgibi

```

opti echo 1;
opti sauv form 'head55.msh';
opti trac psc ftra 'head55_mesh.ps';
opti dime 3 elem cub8;
*****
p0 = 0 0 0;
px = 1 0 0;
pxy = 1 1 0;
py = 0 1 0;
pz = 0 0 1;
c1 = p0 d 3 px;
c2 = px d 3 pxy;
c3 = pxy d 3 py;
c4 = py d 3 p0;

```

```

base = dall c1 c2 c3 c4 plan;
head = base volu tran 3 pz;
tol = 0.001;
elim tol (head et pz);
head2 = head plus (1.5 0 0);
mesh = head et head2;
tass mesh noop;
trac cach qual mesh;
sauv form mesh;
fin;

```

## head55.epx

```

HEAD55
ECHO
  CONV WIN
  CAST mesh
  TRID LAGR
  GEOM CUBE mesh TERM
  COMP COUL GR50 LECT mesh TERM
  MATE HYPE TYPE 1 RO 1040 ! Mooney-Rivlin
    BULK 128E+06 ! This is in Pa
    CO1 185.0 ! These are in Pa
    CO2 46.3
    LECT head TERM
  HYPE TYPE 1 RO 1040 ! Mooney-Rivlin
    BULK 0.0 ! This is in Pa
    CO1 185.0 ! These are in Pa
    CO2 46.3
    LECT head2 TERM
  ECRI COOR DEPL VITE ACCE FINT FEXT FLIA FDEC CONT EPST ECRO TFRE 1.E-3
  FICH SPLI ALIC TFRE 1.E-5
  OPTI NOTE LOG 1
  PAS UTIL
  NOPR PLIN
  CSTA 0.5
  CALC TINI 0.0 TFIN 4.E-3 PASF 1.E-6
  FIN

```

## head56.dgibi

```

opti echo 1;
opti sauv form 'head56.msh';
opti trac psc ftra 'head56_mesh.ps';
opti dime 3 elem cub8;
*****
'DEBPROC' PXSPH8 cent*'POINT' rcub*'FLOTTANT' rsph*'FLOTTANT'
  ncub*'ENTIER';
*
*-----
* Generates a mesh of hexahedra consisting of 1/8 of a cube
* surrounded by 1/8 of a sphere.
* The origin of the cube is at point cent and the radius of the cube
* is rcub. The radius of the sphere is rsph (> rcub).
* The number of subdivisions along the cube radius is ncub.
* The cube is oriented along the global axes X, Y, Z and
* the resulting mesh (cube end sphere portion) lies in the first
* octant.
*
* Input :
* -----
* cent : center of the cube/sphere
* rcub : radius (half-side) of the cube
* rsph : radius of the sphere
* ncub : number of subdivisions along the cube radius
*
* Output :
* -----
* hexa : mesh of hexahedra in 3d
* ss : mesh of spherical external surface (quadrilaterals in 3d)
*-----
* Longueurs caracteristiques du maillage
*
rsph_c = ((2**0.5)/2)*rsph;
rsph_s = ((3**0.5)/3)*rsph;
P = 0.5*rsph*3.14159;
*
* Points
*
dc = rcub / ncub;
OPTI DENS dc;
*
tol = 0.01*dc;
*
O = 0. 0. 0.;
Cx = rcub 0. 0.;
Cy = 0. rcub 0.;
Cz = 0. 0. rcub;
Cdb = rcub rcub 0.;
Cdh = rcub rcub rcub;
Czx = rcub 0. rcub;
Cyz = 0. rcub rcub;
*
* Lignes cube
*
OCx = 0 DROIT Cx;
OCy = 0 DROIT Cy;
OCz = 0 DROIT Cz;

```

```

CxCzx = Cxz DROIT Cx;
CyCyz = Cy DROIT Cyz;
CdbCdh = Cdb DROIT Cdh;
CxCdb = Cx DROIT Cdb;
CyCdb = Cy DROIT Cdb;
CzCdh = Cxz DROIT Cdh;
CyzCdh = Cyz DROIT Cdh;
CzCzx = Cz DROIT Cxz;
CzCyz = Cz DROIT Cyz;
*
* Surfaces cube
*
carre_y1 = OCx CxCzx CzCzx OCz DALLER PLAN;
carre_x1 = OCy CyCyz CzCyz OCz DALLER PLAN;
carre_z1 = OCx CxCdb CyCdb OCy DALLER PLAN;
carre_y2 = CyCdb CdbCdh CyzCdh CyCyz DALLER PLAN;
carre_x2 = CxCdb CdbCdh CzCdh CxCzx DALLER PLAN;
carre_z2 = CzCzx CzxCdh CyzCdh CzCyz DALLER PLAN;
*
* Cube
*
cube = carre_y1 carre_x1 carre_z1 carre_y2 carre_x2 carre_z2 pave;
*
* Points sphere
*
ds = P / (2*ncub);
OPTI DENS ds;
*
ps1 = rsph 0 0;
ps2 = rsph_c rsph_c 0;
ps3 = rsph_s rsph_s rsph_s;
ps4 = rsph_c 0 rsph_c;
ps5 = 0 rsph 0;
ps6 = 0 rsph_c rsph_c;
ps7 = 0 0 rsph;
*
* Lignes sphere
*
ls1 = CERC ncub ps1 0 ps2;
ls2 = CERC ncub ps2 0 ps3;
ls3 = CERC ncub ps3 0 ps4;
ls4 = CERC ncub ps4 0 ps1;
*
ls5 = CERC ncub ps5 0 ps2;
ls6 = CERC ncub ps3 0 ps6;
ls7 = CERC ncub ps6 0 ps5;
ls8 = CERC ncub ps7 0 ps4;
ls9 = CERC ncub ps6 0 ps7;
*
* Surfaces sphere
*
ss1 = DALL ls1 ls2 ls3 ls4 'SPHE' 0;
ss2 = DALL ls5 ls2 ls6 ls7 'SPHE' 0;
ss3 = DALL ls8 ls3 ls6 ls9 'SPHE' 0;
*
ss = ss1 et ss2 et ss3;
elim tol ss;
*
* Volumes sphere
*
vs1 = carre_x2 VOLU ss1;
vs2 = carre_y2 VOLU ss2;
vs3 = carre_z2 VOLU ss3;
*
hexa = cube et vs1 et vs2 et vs3;
elim tol hexa;
DEPL hexa 'PLUS' cent;
*
'FINPROC' hexa ss;
*****
p0 = 0 0 0;
px = 1 0 0;
py = 0 1 0;
pz = 0 0 1;
rcub = 20.E-3;
rsph = 60.E-3;
ncub = 20;
*
* Brain
*
bra1 sbra1 = pxsph8 p0 rcub rsph ncub;
bra2 = bra1 tour 90 p0 pz;
bra3 = bra2 tour 90 p0 pz;
bra4 = bra3 tour 90 p0 pz;
braa = bra1 et bra2 et bra3 et bra4;
tol = (0.001*rcub) / ncub;
elim tol braa;
brab = braa tour 180.E0 p0 px;
brain = braa et brab;
elim tol brain;
*
* CSF (cerebro-spinal fluid)
*
scsf1 = sbra1 plus p0;
scsf2 = scsf1 homo (65.0 / 60.0) p0;
csf1 = scsf1 VOLU scsf2;
csf2 = csf1 tour 90 p0 pz;
csf3 = csf2 tour 90 p0 pz;
csf4 = csf3 tour 90 p0 pz;
csfa = csf1 et csf2 et csf3 et csf4;
elim tol csfa;
csfb = csfa tour 180.E0 p0 px;
csf = csfa et csfb;

```

```

elim tol csf;
*
* Skull
*
ssku1 = scsf2 plus p0;
ssku2 = ssku1 homo (70.0 / 65.0) p0;
sku1 = ssku1 VOLU ssku2;
sku2 = sku1 tour 90 p0 pz;
sku3 = sku2 tour 90 p0 pz;
sku4 = sku3 tour 90 p0 pz;
skua = sku1 et sku2 et sku3 et sku4;
elim tol skua;
skub = skua tour 180.E0 p0 px;
skull = skua et skub;
elim tol skull;
ssku2b = ssku2 tour 90 p0 pz;
ssku2c = ssku2b tour 90 p0 pz;
ssku2d = ssku2c tour 90 p0 pz;
ssku2u = ssku2 et ssku2b et ssku2c et ssku2d;
ssku2v = ssku2u tour 180.E0 p0 px;
ssku2t = ssku2u et ssku2v;
elim tol (ssku2t et skull);
*
* PMATs for contact on external surface of the skull
*
psext = chan poi1 ssku2t;
n = nbel psext;
i = 0;
ni = 0;
repe loop n;
  i = i + 1;
  ei = psext elem i;
  pi = ei poin 1;
  xi yi zi = coor pi;
  si (>eg xi 45.E-3);
  ni = ni + 1;
  si (ega ni 1);
  pscon = ei;
  sinon;
  pscon = pscon et ei;
  finsi;
  fin loop;
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Frontal region.
*
braexp = chan poi1 (enve brain);
xbr = coor 1 braexp;
bx = xbr poin supe 45.E-3;
brapin = brain elem appu larg bx nove;
list (nbel brapin);
*
skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
xsk = coor 1 skuint;
sx = xsk poin supe 50.E-3;
skupin = skull elem appu larg sx nove;
list (nbel skupin);
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Neck region.
*
*braexp = chan poi1 (enve brain);
*xbr = coor 1 braexp;
*bx2 = xbr poin infe -45.E-3;
*brapi2 = brain elem appu larg bx2 nove;
*list (nbel brapi2);
*
*skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
*xsk = coor 1 skuint;
*sx2 = xsk poin infe -50.E-3;
*skupi2 = skull elem appu larg sx2 nove;
*list (nbel skupi2);
*
* Obstacle
*
o1 = 72.E-3 -40.E-3 -40.E-3;
o2 = 72.E-3 -40.E-3 40.E-3;
o3 = 72.E-3 40.E-3 40.E-3;
o4 = 72.E-3 40.E-3 -40.E-3;
c1 = o1 d 40 o2;
c2 = o2 d 40 o3;
c3 = o3 d 40 o4;
c4 = o4 d 40 o1;
obst = dall c1 c2 c3 c4 plan;
*
* Assemble complete mesh
*
fsa1 = csf poin sphe p0 (p0 plus (60.E-3 0 0)) 1.E-5;
fsa2 = csf poin sphe p0 (p0 plus (65.E-3 0 0)) 1.E-5;
fsan = fsa1 et fsa2;
mesh = brain et csf et skull et obst et pscon et fsan et brapin et
skupin et brapi2 et skupi2;
tass mesh noop;
sauv form mesh;
trac cach qual braa;
trac cach qual brab;
trac cach qual brain;
trac cach qual csfa;
trac cach qual csfb;
trac cach qual csf;

```

```

trac cach qual skua;
trac cach qual skub;
trac cach qual skull;
trac cach qual obst;
trac cach qual (skull et pscon);
trac cach qual (brapin et skupin);
trac cach qual (brapi2 et skupi2);
trac cach qual (brapin et skupin et brapi2 et skupi2);
trac cach qual mesh;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
list (nbel skull);
list (nbno skull);
list (nbel obst);
list (nbno obst);
list (nbel pscon);
list (nbno pscon);
list (nbel mesh);
list (nbno mesh);
*
fin;

```

## head56.epx

```

HEAD56
ECHO
  CONV WIN
  CAST mesh
  TRID ALE
  DIME NALE 6402 NBLE 26284 TERM
  GEOM CUBE brain CUBE csf CUB8 skull Q4GS obst PMAT pscon TERM
  COMP EPAI 2.0E-3 LECT obst TERM
  2.0E-3 LECT pscon TERM
  GROU 6 'bray1' LECT brain TERM COND YB LT 0
  'bray2' LECT brain DIFF bray1 TERM
  'csfy1' LECT csf TERM COND YB LT 0
  'csfy2' LECT csf DIFF csfy1 TERM
  'skuy1' LECT skull TERM COND YB LT 0
  'skuy2' LECT skull DIFF skuy1 TERM
  COUL GR50 LECT brain TERM
  TURQ LECT csf TERM
  JAUN LECT skull TERM
  VERT LECT obst TERM
  ROUG LECT pscon TERM
  GRIL LAGR LECT brain skull obst pscon TERM
  EULE LECT fsan TERM
  AUTO AUTR
  MATE HYPE TYPE 1 RO 1040 ! Mooney-Rivlin
  BULK 128E+06 ! This is in Pa
  CO1 185.0 ! These are in Pa
  CO2 46.3
  LECT brain TERM
  FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
  LECT csf TERM
  LINE RO 1800 YOUN 1.500E9 NU 0.21
  LECT skull TERM
  FANT 1.E+3 LECT obst TERM
  FANT 1.E-3 LECT pscon TERM
  OPTI PINS EQVL ASN GRID DPIN 1.01
  LINK COUP SPLT NONE
  BLOQ 123456 LECT obst TERM
  FSA LECT fsan TERM
  ! PINB BODY MLEV 0 LECT obst TERM
  ! BODY DIAM 2.0E-3 LECT pscon TERM
  ! BODY DIAM 5.0E-3 LECT brapin TERM
  ! BODY DIAM 5.0E-3 LECT skupin TERM
  ! BODY DIAM 5.0E-3 LECT brapi2 TERM
  ! BODY DIAM 5.0E-3 LECT skupi2 TERM
  ! EXCL PAIR 1 3
  ! PAIR 2 3
  ! PAIR 1 4
  ! PAIR 2 4
  ! PAIR 1 5
  ! PAIR 2 5
  ! PAIR 1 6
  ! PAIR 2 6
  ! PAIR 3 5
  ! PAIR 3 6
  ! PAIR 4 5
  ! PAIR 4 6
  ! INIT VITE 1 10.0 LECT brain csf skull TERM
  ! VITG 1 10.0 LECT brain csf skull TERM
  ECRI TFRE 1.E-3
  NOPO NOEL
  FICH SPLI ALIC TFRE 1.E-5
  OPTI NOTE LOG 1
  NOPR PLIN
  CSTA 0.5
  NOCR LECT obst pscon TERM
  REZO MVRE MODU VFAC 5.0
  CALC TINI 0.0 TFIN 4.E-3
  FIN

```

## head66.dgibi

```

opti echo 1;
opti sauv form 'head66.msh';
opti trac psc ftra 'head66_mesh.ps';
opti dime 3 elem cub8;
*****

```

```

'DEBPROC' PKSPH8 cent*'POINT' rcub*'FLOTTANT' rsph*'FLOTTANT'
          ncub*'ENTIER';
*
*-----
* Generates a mesh of hexahedra consisting of 1/8 of a cube
* surrounded by 1/8 of a sphere.
* The origin of the cube is at point cent and the radius of the cube
* is rcub. The radius of the sphere is rsph (> rcub).
* The number of subdivisions along the cube radius is ncub.
* The cube is oriented along the global axes X, Y, Z and
* the resulting mesh (cube end sphere portion) lies in the first
* octant.
* Input :
* -----
* cent : center of the cube/sphere
* rcub : radius (half-side) of the cube
* rsph : radius of the sphere
* ncub : number of subdivisions along the cube radius
*
* Output :
* -----
* hexa : mesh of hexahedra in 3d
* ss : mesh of spherical external surface (quadrilaterals in 3d)
*-----
* Longueurs caracteristiques du maillage
*
rsph_c = ((2**0.5)/2)*rsph;
rsph_s = ((3**0.5)/3)*rsph;
P = 0.5*rsph*3.14159;
*
* Points
*
dc = rcub / ncub;
OPTI DENS dc;
*
tol = 0.01*dc;
*
O = 0. 0. 0.;
Cx = rcub 0. 0.;
Cy = 0. rcub 0.;
Cz = 0. 0. rcub;
Cdb = rcub rcub 0.;
Cdh = rcub rcub rcub;
Czx = rcub 0. rcub;
Cyz = 0. rcub rcub;
*
* Lignes cube
*
OCx = 0 DROIT Cx;
OCy = 0 DROIT Cy;
OCz = 0 DROIT Cz;
CxCz = Cxz DROIT Cx;
CyCyz = Cy DROIT Cyz;
CdbCdh = Cdb DROIT Cdh;
CxCdb = Cx DROIT Cdb;
CyCdb = Cy DROIT Cdb;
CzxCdh = Czx DROIT Cdh;
CyzCdh = Cyz DROIT Cdh;
CzCz = Cz DROIT Cz;
CzCyz = Cz DROIT Cyz;
*
* Surfaces cube
*
carre_y1 = OCx CxCz CzCz OCz DALLER PLAN;
carre_x1 = OCy CyCyz CzCyz OCz DALLER PLAN;
carre_z1 = OCx CxCdb CyCdb OCy DALLER PLAN;
carre_y2 = CyCdb CdbCdh CyzCdh CyCyz DALLER PLAN;
carre_x2 = CxCdb CdbCdh CzxCdh CxCz DALLER PLAN;
carre_z2 = CzCz CzxCdh CyzCdh CzCyz DALLER PLAN;
*
* Cube
*
cube = carre_y1 carre_x1 carre_z1 carre_y2 carre_x2 carre_z2 pave;
*
* Points sphere
*
ds = P / (2*ncub);
OPTI DENS ds;
*
ps1 = rsph 0 0;
ps2 = rsph_c rsph_c 0;
ps3 = rsph_s rsph_s rsph_s;
ps4 = rsph_c 0 rsph_c;
ps5 = 0 rsph 0;
ps6 = 0 rsph_c rsph_c;
ps7 = 0 0 rsph;
*
* Lignes sphere
*
ls1 = CERC ncub ps1 0 ps2;
ls2 = CERC ncub ps2 0 ps3;
ls3 = CERC ncub ps3 0 ps4;
ls4 = CERC ncub ps4 0 ps1;
*
ls5 = CERC ncub ps5 0 ps2;
ls6 = CERC ncub ps3 0 ps6;
ls7 = CERC ncub ps6 0 ps5;
ls8 = CERC ncub ps7 0 ps4;
ls9 = CERC ncub ps6 0 ps7;
*
* Surfaces sphere
*
ss1 = DALL ls1 ls2 ls3 ls4 'SPHE' 0;
ss2 = DALL ls5 ls2 ls6 ls7 'SPHE' 0;
ss3 = DALL ls8 ls3 ls6 ls9 'SPHE' 0;
*
ss = ss1 et ss2 et ss3;
elim tol ss;
*
* Volumes sphere
*
vs1 = carre_x2 VOLU ss1;
vs2 = carre_y2 VOLU ss2;
vs3 = carre_z2 VOLU ss3;
*
hexa = cube et vs1 et vs2 et vs3;
elim tol hexa;
DEPL hexa 'PLUS' cent;
*
'FINPROC' hexa ss;
*-----
p0 = 0 0 0;
px = 1 0 0;
py = 0 1 0;
pz = 0 0 1;
rcub = 20.E-3;
rsph = 60.E-3;
ncub = 20;
*
* Brain
*
bra1 sbra1 = pxsph8 p0 rcub rsph ncub;
bra2 = bra1 tour 90 p0 pz;
bra3 = bra2 tour 90 p0 pz;
bra4 = bra3 tour 90 p0 pz;
braa = bra1 et bra2 et bra3 et bra4;
tol = (0.001*rcub) / ncub;
elim tol braa;
brab = braa tour 180.E0 p0 px;
brain = braa et brab;
elim tol brain;
*
* CSF (cerebro-spinal fluid)
*
scsf1 = sbra1 plus p0;
scsf2 = scsf1 homo (65.0 / 60.0) p0;
csf1 = scsf1 VOLU scsf2;
csf2 = csf1 tour 90 p0 pz;
csf3 = csf2 tour 90 p0 pz;
csf4 = csf3 tour 90 p0 pz;
csfa = csf1 et csf2 et csf3 et csf4;
elim tol csfa;
csfb = csfa tour 180.E0 p0 px;
csf = csfa et csfb;
elim tol csf;
*
* Skull
*
ssku1 = scsf2 plus p0;
ssku2 = ssku1 homo (70.0 / 65.0) p0;
sku1 = ssku1 VOLU ssku2;
sku2 = sku1 tour 90 p0 pz;
sku3 = sku2 tour 90 p0 pz;
sku4 = sku3 tour 90 p0 pz;
skua = sku1 et sku2 et sku3 et sku4;
elim tol skua;
skub = skua tour 180.E0 p0 px;
skull = skua et skub;
elim tol skull;
ssku2b = ssku2 tour 90 p0 pz;
ssku2c = ssku2b tour 90 p0 pz;
ssku2d = ssku2c tour 90 p0 pz;
ssku2u = ssku2 et ssku2b et ssku2c et ssku2d;
ssku2v = ssku2u tour 180.E0 p0 px;
ssku2t = ssku2u et ssku2v;
elim tol (ssku2t et skull);
*
* PMATs for contact on external surface of the skull
*
psext = chan poil ssku2t;
n = nbel psext;
i = 0;
ni = 0;
repe loop n;
i = i + 1;
ei = psext elem i;
pi = ei poin i;
xi yi zi = coor pi;
si (>eg xi 45.E-3);
ni = ni + 1;
si (ega ni 1);
pscon = ei;
sinon;
pscon = pscon et ei;
finsi;
finsi;
fin loop;
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Frontal region.
*
braexp = chan poil (enve brain);
xbr = coor 1 braexp;

```



```

bx = xbr poin supe 45.E-3;
brapin = brain elem appu larg bx nove;
list (nbel brapin);
*
skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
xsk = coor 1 skuint;
sx = xsk poin supe 50.E-3;
skupin = skull elem appu larg sx nove;
list (nbel skupin);
*
* Pinballs for contact on external surface of the brain and on
* the internal surface of the skull (to avoid excessive narrowing
* of the CSF fluid gap). Neck region.
*
*braexp = chan poi1 (enve brain);
*xbr = coor 1 braexp;
bx2 = xbr poin infe -45.E-3;
brapi2 = brain elem appu larg bx2 nove;
list (nbel brapi2);
*
*skuint = skull poin sphe p0 (p0 plus (0 0 65.E-3)) 1.E-5;
*xsk = coor 1 skuint;
sx2 = xsk poin infe -50.E-3;
skupi2 = skull elem appu larg sx2 nove;
list (nbel skupi2);
*
* Obstacle
*
o1 = 72.E-3 -40.E-3 -40.E-3;
o2 = 72.E-3 -40.E-3 40.E-3;
o3 = 72.E-3 40.E-3 40.E-3;
o4 = 72.E-3 40.E-3 -40.E-3;
c1 = o1 d 40 o2;
c2 = o2 d 40 o3;
c3 = o3 d 40 o4;
c4 = o4 d 40 o1;
obst = dall c1 c2 c3 c4 plan;
*
* Assemble complete mesh
*
fsa1 = csf poin sphe p0 (p0 plus (60.E-3 0 0)) 1.E-5;
fsa2 = csf poin sphe p0 (p0 plus (65.E-3 0 0)) 1.E-5;
fsan = fsa1 et fsa2;
mesh = brain et csf et skull et obst et pscon et fsan et brapin et
skupin et brapi2 et skupi2;
tass mesh noop;
sauv form mesh;
trac cach qual braa;
trac cach qual brab;
trac cach qual brain;
trac cach qual csfa;
trac cach qual csfb;
trac cach qual csf;
trac cach qual skua;
trac cach qual skub;
trac cach qual skull;
trac cach qual obst;
trac cach qual (skull et pscon);
trac cach qual (brapin et skupin);
trac cach qual (brapi2 et skupi2);
trac cach qual (brapin et skupin et brapi2 et skupi2);
trac cach qual mesh;
list (nbel brain);
list (nbno brain);
list (nbel csf);
list (nbno csf);
list (nbel skull);
list (nbno skull);
list (nbel obst);
list (nbno obst);
list (nbel pscon);
list (nbno pscon);
list (nbel mesh);
list (nbno mesh);
*
fin;

```

## head66.epx

```

HEAD66
ECHO
CONV WIN
CAST mesh
TRID ALE
DIME NALE 6402 NBLE 26284 TERM
GEMO CUBE brain CUBE csf CUB8 skull Q4GS obst PMAT pscon TERM
COMP EPAI 2.0E-3 LECT obst TERM
2.0E-3 LECT pscon TERM
GROU 6 'bray1' LECT brain TERM COND YB LT 0
'bray2' LECT brain DIFF bray1 TERM
'csfy1' LECT csf TERM COND YB LT 0
'csfy2' LECT csf DIFF csfy1 TERM
'skuy1' LECT skull TERM COND YB LT 0
'skuy2' LECT skull DIFF skuy1 TERM
COUL GR50 LECT brain TERM
TURQ LECT csf TERM
JAUN LECT skull TERM
VERT LECT obst TERM
ROUG LECT pscon TERM
GRIL LAGR LECT brain skull obst pscon TERM
EULE LECT fsan TERM

```

```

AUTO AUTR
MATE HYPE TYPE 4 RO 1040
BULK 128E+06
CO1 -0.14058E+02
CO2 0.38793E+00
CO5 0.88365E+06
CO6 0.50888E+09
LECT brain TERM
FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
LECT csf TERM
LINE RO 1800 YOUN 1.500E9 NU 0.21
LECT skull TERM
FANT 1.E+3 LECT obst TERM
FANT 1.E-3 LECT pscon TERM
OPTI PINS EQVL ASN GRID DPIN 1.01
LINK COUP SPLIT NONE
BLOQ 123456 LECT obst TERM
FSA LECT fsan TERM
! PINB BODY MLEV 0 LECT obst TERM
! BODY DIAM 2.0E-3 LECT pscon TERM
! BODY DIAM 5.0E-3 LECT brapin TERM
! BODY DIAM 5.0E-3 LECT skupin TERM
! BODY DIAM 5.0E-3 LECT brapi2 TERM
! BODY DIAM 5.0E-3 LECT skupi2 TERM
! EXCL PAIR 1 3
! PAIR 2 3
! PAIR 1 4
! PAIR 2 4
! PAIR 1 5
! PAIR 2 5
! PAIR 1 6
! PAIR 2 6
! PAIR 3 5
! PAIR 3 6
! PAIR 4 5
! PAIR 4 6
! INIT VITE 1 10.0 LECT brain csf skull TERM
! VITG 1 10.0 LECT brain csf skull TERM
ECRI TFRE 1.E-3
NOPO NOEL
FICH SPLI ALIC TFRE 1.E-5
OPTI NOTE LOG 1
NOPR PLIN
CSTA 0.5
NUCR LECT obst pscon TERM
REZO MVRE MODU VFAC 5.0
CALC TINI 0.0 TFIN 4.E-3
FIN

```

## huma01.epx

```

HUMA01
CONV WIN
LAGR TRID
OPTI NOPR
GEMO SCAL FACT 0.001
9 '(3E13.5)' '(418)' POIN 905237 TETR 5300984 TERM
COMP GROU 9
'fluid' LECT
1 4 7 11 15 19 20 21 26
5300968 5300970 5300973 5300976 5300977 5300978 5300982 5300983 5300984
TERM
'grey_l' LECT
2 8 9 14 16 17 22 25 30
5300955 5300969 5300971 5300981
TERM
'white_l' LECT
3 24 73 78 120 195 238 248 249
5300948 5300951 5300957 5300958 5300974 5300975
TERM
'grey_r' LECT
5 6 10 12 13 18 23 28 29
5300930 5300933 5300937 5300950 5300980
TERM
'white_r' LECT
65 67 118 173 181 203 204 213 284
5300961 5300963 5300966 5300972 5300979
TERM
'emin' LECT 1117091 TERM
'emax' LECT 5217 TERM
'edtmn' LECT 2357453 TERM
'edtmax' LECT 1180305 TERM
NGRO 5
'white_r' LECT
1 4 12 15 16 21 42 51 72
905229 905230 905235
TERM
'nfluid_f' LECT
4392 4970 5067 5071 5077 6327 6671 6722 6874
904511 904513 904670
TERM
'nfluid_l' LECT
4392 5067 6327 6671 6722 6874 7140 7309 7509

```

```

      8081   9455   9471   9574   9835   10281   10613   10864   10974
904469  904482  904489  904511  904670
TERM
'nfluid_2' LECT
      5067   6671   6874   7309   7509   8081   9471   9835   10613
. . .
904457  904466  904482  904489  904511  904670
TERM
'nfluid_d' LECT
      10032  11541  12350  18066  22344  22345  22346  23805  23806
. . .
901617  902396  903438  903440  903661  904189
TERM
      COUL TURQ LECT fluid TERM
      GR50 LECT grey_l grey_r TERM
      BLAN LECT white_l white_r TERM
      ROUG LECT emin edtmin TERM
      VERT LECT emax edtmax TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
      LECT grey_l white_l grey_r white_r TERM
      FLUI RO 1040 C 1451 PINI 1.E5 PREF 1.E5 PMIN 0 VISC 1.002E-3
      LECT fluid TERM
ECRI FREQ 100
      NOPO NOEL
      FICH ALIC TFRE 1.E-3
      FICH PVTK TFRE 1.E-3
      GROU AUTO
OPTI PAS AUTO NOTE LOG 1
      STEL
CALC TINI 0. TEND 1.DO NMAX 0
FIN

```

## hype01.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
h = 0.001;
l = 1.0;
p1 = 0 0 0;
p2 = 0 h 0;
p3 = 0 h h;
p4 = 0 0 h;
base = manu qua4 p1 p2 p3 p4;
pm = p1 plus ((0.5*1) 0 0);
bar1 = base volu tran 500 pm;
tol = 0.000001;
elim tol (bar1 et pm);
bar2 = bar1 plus pm;
elim tol (bar1 et bar2);
bar = bar1 et bar2;
e1 = bar1 elem cont pm;
e2 = bar2 elem cont pm;
endp = chan poi1 ((base plus pm) plus pm);
elim tol (bar et endp);
mesh = bar et endp et e1 et e2;
tass mesh;
opti sauv form 'hype01.msh';
sauv form mesh;
opti trac psc ftra 'hype01_mesh.ps';
trac cach qual mesh;
fin;

```

## hype01.epx

```

HYPE01
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM CUBE bar TERM
COMP COUL VERT LECT bar TERM
MATE VM23 RO 8000. YOUN 2.0E11 NU 0.0 ELAS 2.0E11
      TRAC 1 2.0E11 1.E0
      LECT bar TERM
LINK COUP
      BLOQ 123 LECT endp TERM
INIT VITE 1 100.0 LECT bar DIFF endp TERM
ECRI DEPL VITE CONT ECRO TFREQ 0.1E-3
      FICH ALIC TEMP FREQ 1
      POIN LECT p1 pm TERM
      ELEM LECT e1 e2 TERM
OPTI PAS AUTO NOTEST
      LOG 1
CALC TINI 0. TEND 0.5E-3
SUIT
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dx_2' DEPL COMP 1 NOEU LECT p1 TERM
COUR 2 'dx_3' DEPL COMP 1 NOEU LECT pm TERM
COUR 3 'sg_1' CONT COMP 1 ELEM LECT e1 TERM
COUR 4 'sg_2' CONT COMP 1 ELEM LECT e2 TERM

```

```

TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'CONTR. [PA]' YZER
TRAC 4 AXES 1.0 'CONTR. [PA]' YZER
FIN

```

## hype02.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
h = 0.001;
l = 1.0;
p1 = 0 0 0;
p2 = 0 h 0;
p3 = 0 h h;
p4 = 0 0 h;
base = manu qua4 p1 p2 p3 p4;
pm = p1 plus ((0.5*1) 0 0);
bar1 = base volu tran 500 pm;
tol = 0.000001;
elim tol (bar1 et pm);
bar2 = bar1 plus pm;
elim tol (bar1 et bar2);
bar = bar1 et bar2;
e1 = bar1 elem cont pm;
e2 = bar2 elem cont pm;
endp = chan poi1 ((base plus pm) plus pm);
elim tol (bar et endp);
mesh = bar et endp et e1 et e2;
tass mesh;
opti sauv form 'hype02.msh';
sauv form mesh;
opti trac psc ftra 'hype02_mesh.ps';
trac cach qual mesh;
fin;

```

## hype02.epx

```

HYPE02
ECHO
!CONV win
CAST mesh
TRID LAGR
GEOM CUBE bar TERM
COMP COUL VERT LECT bar TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
      LECT bar TERM
LINK COUP
      BLOQ 1 LECT endp TERM
INIT VITE 1 1.0 LECT bar DIFF endp TERM
ECRI DEPL VITE CONT ECRO TFREQ 0.5E-3
      FICH ALIC TFRE 0.5E-3
      FICH ALIC TEMP FREQ 1
      POIN LECT p1 pm TERM
      ELEM LECT e1 e2 TERM
OPTI PAS UTIL
      NOTE
      LOG 1
      STEL
CALC TINI 0. TEND 50.E-3 PASF 1.E-6
SUIT
Post-treatment
ECHO
RESU ALIC TEMP GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'dx_2' DEPL COMP 1 NOEU LECT p1 TERM
COUR 2 'dx_3' DEPL COMP 1 NOEU LECT pm TERM
COUR 3 'sg_1' CONT COMP 1 ELEM LECT e1 TERM
COUR 4 'sg_2' CONT COMP 1 ELEM LECT e2 TERM
TRAC 1 AXES 1.0 'DISPL. [M]' YZER
TRAC 2 AXES 1.0 'DISPL. [M]' YZER
TRAC 3 AXES 1.0 'CONTR. [PA]' YZER
TRAC 4 AXES 1.0 'CONTR. [PA]' YZER
SUIT
Post-treatment from ALIC file
RESU ALIC GARD PSCR
COMP NGRO 1 'axis' LECT bar TERM COND LINE
      X1 0 Y1 0 Z1 0
      X2 1 Y2 0 Z2 0 TOL 1.E-5
SORT GRAP
AXTE 1.0 'Time [s]'
SCOU 11 'sx_1' CONT COMP 1 T 0.005 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
TRAC 11 AXES 1.0 'sx_1ms' YZER
FIN

```

## hype02a.epx

```

HYPE02A
RESU ALIC 'hype02.ali' GARD PSCR
COMP NGRO 1 'axis' LECT bar TERM COND LINE
      X1 0 Y1 0 Z1 0
      X2 1 Y2 0 Z2 0 TOL 1.E-5
SORT GRAP

```

```

AXTE 1.0 'Time [s]'
SCOU 1 'sx_10ms' CONT COMP 1 T 0.01 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 2 'sx_20ms' CONT COMP 1 T 0.02 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 3 'sx_30ms' CONT COMP 1 T 0.03 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 4 'sx_40ms' CONT COMP 1 T 0.04 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 5 'sx_50ms' CONT COMP 1 T 0.05 SAXE 1.0 'curr_abcissa'
      LECT axis TERM

TRAC 5 AXES 1.0 'sx' YZER
TRAC 1 2 3 4 5 AXES 1.0 'sx' YZER
COLO NOIR VERT TURQ ROSE ROUG
SCOU 11 'vx_10ms' VITE COMP 1 T 0.01 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 12 'vx_20ms' VITE COMP 1 T 0.02 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 13 'vx_30ms' VITE COMP 1 T 0.03 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 14 'vx_40ms' VITE COMP 1 T 0.04 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 15 'vx_50ms' VITE COMP 1 T 0.05 SAXE 1.0 'curr_abcissa'
      LECT axis TERM

TRAC 11 12 13 14 15 AXES 1.0 'vx' YZER
COLO NOIR VERT TURQ ROSE ROUG
FIN

```

## hype02b.epx

```

HYPE02B
RESU ALIC 'hype02.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 9.75000E-01 5.00000E-04 1.25550E-01 ! Last 50 elems
! 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 : 9.75000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 2.50100E-02
!RADIUS : 1.25050E-01
!ASPECT : 1.00000E+00
!NEAR : 9.75390E-02
!FAR : 1.75070E-01
CAME 2 EYE 9.90000E-01 5.00000E-04 5.06248E-02 ! Last 20 elems
! 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 : 9.90000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 1.00250E-02
!RADIUS : 5.01248E-02
!ASPECT : 1.00000E+00
!NEAR : 3.90973E-02
!FAR : 7.01747E-02
CAME 3 EYE 9.95000E-01 5.00000E-04 2.57487E-02 ! Last 10 elems
! 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 : 9.95000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 5.04975E-03
!RADIUS : 2.52487E-02
!ASPECT : 1.00000E+00
!NEAR : 1.96940E-02
!FAR : 3.53482E-02
SCEN GEOM NAVI FREE
LIMA ON
SLER CAM1 3 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTA 101 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## hype02c.epx

```

HYPE02C
RESU ALIC 'hype02.ali' GARD PSCR
COMP NGRO 1 'axis' LECT bar TERM COND LINE
      X1 0.9 Y1 0 Z1 0
      X2 1.0 Y2 0 Z2 0 TOL 1.E-5

SORT GRAP
AXTE 1.0 'Time [s]'
SCOU 1 'sx_10ms' CONT COMP 1 T 0.01 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 2 'sx_20ms' CONT COMP 1 T 0.02 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 3 'sx_30ms' CONT COMP 1 T 0.03 SAXE 1.0 'curr_abcissa'
      LECT axis TERM

```

```

SCOU 4 'sx_40ms' CONT COMP 1 T 0.04 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 5 'sx_50ms' CONT COMP 1 T 0.05 SAXE 1.0 'curr_abcissa'
      LECT axis TERM

TRAC 1 2 3 4 5 AXES 1.0 'sx' YZER
COLO NOIR VERT TURQ ROSE ROUG
SCOU 11 'vx_10ms' VITE COMP 1 T 0.01 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 12 'vx_20ms' VITE COMP 1 T 0.02 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 13 'vx_30ms' VITE COMP 1 T 0.03 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 14 'vx_40ms' VITE COMP 1 T 0.04 SAXE 1.0 'curr_abcissa'
      LECT axis TERM
SCOU 15 'vx_50ms' VITE COMP 1 T 0.05 SAXE 1.0 'curr_abcissa'
      LECT axis TERM

TRAC 11 12 13 14 15 AXES 1.0 'vx' YZER
COLO NOIR VERT TURQ ROSE ROUG
FIN

```

## hype02d.epx

```

HYPE02D
RESU ALIC 'hype02.ali' GARD PSCR
SORT VISU NSTO 1
PLAY
CAME 1 EYE 9.75000E-01 5.00000E-04 1.25550E-01 ! Last 50 elems
! 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 : 9.75000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 2.50100E-02
!RADIUS : 1.25050E-01
!ASPECT : 1.00000E+00
!NEAR : 9.75390E-02
!FAR : 1.75070E-01
CAME 2 EYE 9.90000E-01 5.00000E-04 5.06248E-02 ! Last 20 elems
! 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 : 9.90000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 1.00250E-02
!RADIUS : 5.01248E-02
!ASPECT : 1.00000E+00
!NEAR : 3.90973E-02
!FAR : 7.01747E-02
CAME 3 EYE 9.95000E-01 5.00000E-04 2.57487E-02 ! Last 10 elems
! 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 : 9.95000E-01 5.00000E-04 5.00000E-04
!RSPHERE: 5.04975E-03
!RADIUS : 2.52487E-02
!ASPECT : 1.00000E+00
!NEAR : 1.96940E-02
!FAR : 3.53482E-02
SCEN GEOM NAVI FREE
LIMA ON
SLER CAM1 1 NFRA 1
FREQ 1
TRAC OFFS FICH AVI NOCL NFTA 101 FPS 10 KFRE 10 COMP -1 REND
GOTR LOOP 99 OFFS FICH AVI CONT NOCL REND
GO
TRAC OFFS FICH AVI CONT REND
ENDPLAY
FIN

```

## hype03.epx

```

HYPE03
ECHO
CONV win
TRID LAGR
GEOM LIBR POIN 8 CUB8 1 TERM
0 0 0 1 0 0 1 1 0 0 1 0
0 0 1 1 0 1 0 1 1 0 1 1
1 2 3 4 5 6 7 8
COMP COUL VERT LECT 1 TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 128E+06 ! This is in Pa
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
      LECT 1 TERM
FONC 1 TABL 6 0 1 0.8 1 0.8001 0 1.1901 0 1.2 -1 10.0 -1
LINK COUP
      BLOQ 3 LECT 1 2 3 4 TERM
      VITE 3 -1.0 FONC 1 LECT 5 6 7 8 TERM
INIT VITE 3 -1.0 LECT 5 6 7 8 TERM
ECRI DEPL VITE CONT ECRO EPST TFRE 0.1
FICH ALIC FREQ 1
OPTI PAS UTIL NOTE

```

```

LOG 1
STEL
CALC TINI 0. TEND 0.88 PASF 1.E-3
PLAY
CAME 3 EYE 2.40751E+00 -6.39880E+00 2.37119E+00
! Q 7.86566E-01 6.03553E-01 9.62340E-02 8.81822E-02
VIEW -2.57834E-01 9.32497E-01 -2.52925E-01
RIGH 9.65926E-01 2.54887E-01 -4.49435E-02
UP -2.25576E-02 2.55894E-01 9.66442E-01
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 5.00000E-01 5.00000E-01 5.00000E-01
!RSPHERE: 1.64404E+00
!RADIUS : 7.39820E+00
!ASPECT : 1.00000E+00
!NEAR : 5.75416E+00
!FAR : 1.06863E+01
SCEN GEOM NAVI FREE
INIT CGLA
LIMA ON
SLER CAM1 3 NFRA 1
FREQ 0 TFRE 0.01
TRAC OFFS FICH AVI NOCL NFTO 89 FPS 10 KFRE 10 COMP -1 DEFO REND
GOTR LOOP 87 OFFS FICH AVI CONT NOCL DEFO REND
GO
TRAC OFFS FICH AVI CONT DEFO REND
ENDPLAY
SUIT
Post-treatment
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx_1' CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy_1' CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sz_1' CONT COMP 3 ELEM LECT 1 TERM
COUR 11 'ex_1' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey_1' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'ez_1' EPST COMP 3 ELEM LECT 1 TERM
TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
COLO NOIR NOIR ROUG
TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
COLO NOIR NOIR ROUG
FIN

```

## hype03a.epx

```

HYPE03A
RESU ALIC 'hype03.ali' GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx' CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy' CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sz' CONT COMP 3 ELEM LECT 1 TERM
COUR 4 'sxy' CONT COMP 4 ELEM LECT 1 TERM
COUR 5 'syz' CONT COMP 5 ELEM LECT 1 TERM
COUR 6 'sxz' CONT COMP 6 ELEM LECT 1 TERM
TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
TRAC 4 5 6 AXES 1.0 'CONTR. [PA]' YZER
COUR 11 'ex' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'ez' EPST COMP 3 ELEM LECT 1 TERM
COUR 14 'gxy' EPST COMP 4 ELEM LECT 1 TERM
COUR 15 'gyz' EPST COMP 5 ELEM LECT 1 TERM
COUR 16 'gxz' EPST COMP 6 ELEM LECT 1 TERM
TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
TRAC 14 15 16 AXES 1.0 'EPST [-]' YZER
COUR 21 'ecr1' ECRO COMP 1 ELEM LECT 1 TERM
COUR 22 'ecr2' ECRO COMP 2 ELEM LECT 1 TERM
COUR 23 'ecr3' ECRO COMP 3 ELEM LECT 1 TERM
COUR 24 'ecr4' ECRO COMP 4 ELEM LECT 1 TERM
COUR 25 'ecr5' ECRO COMP 5 ELEM LECT 1 TERM
COUR 26 'ecr6' ECRO COMP 6 ELEM LECT 1 TERM
COUR 27 'ecr1' ECRO COMP 7 ELEM LECT 1 TERM
TRAC 21 AXES 1.0 'ECR1' YZER
TRAC 22 AXES 1.0 'ECR2' YZER
TRAC 23 AXES 1.0 'ECR3' YZER
TRAC 24 AXES 1.0 'ECR4' YZER
TRAC 25 AXES 1.0 'ECR5' YZER
TRAC 26 AXES 1.0 'ECR6' YZER
TRAC 27 AXES 1.0 'ECR7' YZER
FIN

```

## hype04.epx

```

HYPE04
ECHO
CONV win
TRID LAGR
GEOM LIBR POIN 8 CUBE 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 2 3 4 5 6 7 8
COMP COUL VERT LECT 1 TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
BULK 128E+06 ! This is in Pa
C01 -10.8 ! This is alpha_1
C02 19.7 ! This is alpha_2
C05 -0.29 ! This is mu_1
C06 9.6E-3 ! This is mu_2

```

```

LECT 1 TERM
FONC 1 TABL 6 0 1 0 8 1 0 0.8001 0 1.1901 0 1.2 -1 10.0 -1
LINK COUP
BLOQ 3 LECT 1 2 3 4 TERM
BLOQ 12 LECT tous TERM
VITE 3 -1.0 FONC 1 LECT 5 6 7 8 TERM
INIT VITE 3 -1.0 LECT 5 6 7 8 TERM
ECRI DEPL VITE CONT ECRO EPST TFRE 0.1
FICH ALIC FREQ 1
OPTI PAS UTIL NOTE
LOG 1
STEL
CALC TINI 0. TEND 2.0 PASF 1.E-3
PLAY
CAME 3 EYE 2.40751E+00 -6.39880E+00 2.37119E+00
! Q 7.86566E-01 6.03553E-01 9.62340E-02 8.81822E-02
VIEW -2.57834E-01 9.32497E-01 -2.52925E-01
RIGH 9.65926E-01 2.54887E-01 -4.49435E-02
UP -2.25576E-02 2.55894E-01 9.66442E-01
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 5.00000E-01 5.00000E-01 5.00000E-01
!RSPHERE: 1.64404E+00
!RADIUS : 7.39820E+00
!ASPECT : 1.00000E+00
!NEAR : 5.75416E+00
!FAR : 1.06863E+01
SCEN GEOM NAVI FREE
INIT CGLA
LIMA ON
SLER CAM1 3 NFRA 1
FREQ 0 TFRE 0.01
TRAC OFFS FICH AVI NOCL NFTO 201 FPS 10 KFRE 10 COMP -1 DEFO REND
GOTR LOOP 199 OFFS FICH AVI CONT NOCL DEFO REND
GO
TRAC OFFS FICH AVI CONT DEFO REND
ENDPLAY
SUIT
Post-treatment
RESU ALIC GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx_1' CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy_1' CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sz_1' CONT COMP 3 ELEM LECT 1 TERM
COUR 11 'ex_1' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey_1' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'ez_1' EPST COMP 3 ELEM LECT 1 TERM
TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
COLO NOIR NOIR ROUG
TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
COLO NOIR NOIR ROUG
FIN

```

## hype04a.epx

```

HYPE04A
RESU ALIC 'hype04.ali' GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx' CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy' CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sz' CONT COMP 3 ELEM LECT 1 TERM
COUR 4 'sxy' CONT COMP 4 ELEM LECT 1 TERM
COUR 5 'syz' CONT COMP 5 ELEM LECT 1 TERM
COUR 6 'sxz' CONT COMP 6 ELEM LECT 1 TERM
TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
TRAC 4 5 6 AXES 1.0 'CONTR. [PA]' YZER
COUR 11 'ex' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'ez' EPST COMP 3 ELEM LECT 1 TERM
COUR 14 'gxy' EPST COMP 4 ELEM LECT 1 TERM
COUR 15 'gyz' EPST COMP 5 ELEM LECT 1 TERM
COUR 16 'gxz' EPST COMP 6 ELEM LECT 1 TERM
TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
TRAC 14 15 16 AXES 1.0 'EPST [-]' YZER
COUR 21 'ecr1' ECRO COMP 1 ELEM LECT 1 TERM
COUR 22 'ecr2' ECRO COMP 2 ELEM LECT 1 TERM
COUR 23 'ecr3' ECRO COMP 3 ELEM LECT 1 TERM
COUR 24 'ecr4' ECRO COMP 4 ELEM LECT 1 TERM
COUR 25 'ecr5' ECRO COMP 5 ELEM LECT 1 TERM
COUR 26 'ecr6' ECRO COMP 6 ELEM LECT 1 TERM
COUR 27 'ecr1' ECRO COMP 7 ELEM LECT 1 TERM
TRAC 21 AXES 1.0 'ECR1' YZER
TRAC 22 AXES 1.0 'ECR2' YZER
TRAC 23 AXES 1.0 'ECR3' YZER
TRAC 24 AXES 1.0 'ECR4' YZER
TRAC 25 AXES 1.0 'ECR5' YZER
TRAC 26 AXES 1.0 'ECR6' YZER
TRAC 27 AXES 1.0 'ECR7' YZER
FIN

```

## hype05.epx

```

HYPE05
ECHO
CONV win
TRID LAGR
GEOM LIBR POIN 8 CUBE 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 2 3 4 5 6 7 8

```

```

COMP COUL VERT LECT 1 TERM
MATE HYPE TYPE 4 RO 1040 ! Ogden (new)
      BULK 0.0 ! Incompressible (?)
      CO1 -10.8 ! This is alpha_1
      CO2 19.7 ! This is alpha_2
      CO5 -0.29 ! This is mu_1
      CO6 9.6E-3 ! This is mu_2
      LECT 1 TERM
FONC 1 TABL 6 0 1 0.8 1 0.8001 0 1.1901 0 1.2 -1 10.0 -1
LINK COUP
      BLOQ 3 LECT 1 2 3 4 TERM
      BLOQ 12 LECT tous TERM
      VITE 3 -1.0 FONC 1 LECT 5 6 7 8 TERM
INIT VITE 3 -1.0 LECT 5 6 7 8 TERM
ECRI DEPL VITE CONT ECRO EPST TPRE 0.1
      FICH ALIC FREQ 1
OPTI PAS UTIL NOTE
      LOG 1
      STEL
CALC TINI 0. TEND 2.0 PASF 1.E-3
PLAY
CAME 3 EYE 2.40751E+00 -6.39880E+00 2.37119E+00
! Q 7.86566E-01 6.03553E-01 9.62340E-02 8.81822E-02
VIEW -2.57834E-01 9.32497E-01 -2.52925E-01
RIGH 9.65926E-01 2.54887E-01 -4.49435E-02
UP -2.25576E-02 2.55894E-01 9.66442E-01
FOV 2.48819E+01
!NAVIGATION MODE: ROTATING CAMERA
!CENTER : 5.00000E-01 5.00000E-01 5.00000E-01
!RSPHERE: 1.64404E+00
!RADIUS : 7.39820E+00
!ASPECT : 1.00000E+00
!NEAR : 5.75416E+00
!FAR : 1.06863E+01
SCEN GEOM NAVI FREE
      INIT OGLA
      LIMA ON
      SLER CAM1 3 NFRA 1
      FREQ 0 TPRE 0.01
      TRAC OFFS FICH AVI NOCL NPTO 201 FPS 10 KPRE 10 COMP -1 DEFO REND
      GOTR LOOP 199 OFFS FICH AVI CONT NOCL DEFO REND
      GO
      TRAC OFFS FICH AVI CONT DEFO REND
      ENDPLOY
      SUIT
      Post-treatment
      RESU ALIC GARD PSCR
      SORT GRAP
      AXTE 1.0 'Time [s]'
      COUR 1 'sx_1' CONT COMP 1 ELEM LECT 1 TERM
      COUR 2 'sy_1' CONT COMP 2 ELEM LECT 1 TERM
      COUR 3 'sz_1' CONT COMP 3 ELEM LECT 1 TERM
      COUR 11 'ex_1' EPST COMP 1 ELEM LECT 1 TERM
      COUR 12 'ey_1' EPST COMP 2 ELEM LECT 1 TERM
      COUR 13 'ez_1' EPST COMP 3 ELEM LECT 1 TERM
      TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
      COLO NOIR NOIR ROUG
      TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
      COLO NOIR NOIR ROUG
      FIN

```

## hype05a.epx

```

HYPE05A
RESU ALIC 'hype05.ali' GARD PSCR
SORT GRAP
AXTE 1.0 'Time [s]'
COUR 1 'sx' CONT COMP 1 ELEM LECT 1 TERM
COUR 2 'sy' CONT COMP 2 ELEM LECT 1 TERM
COUR 3 'sz' CONT COMP 3 ELEM LECT 1 TERM
COUR 4 'sxy' CONT COMP 4 ELEM LECT 1 TERM
COUR 5 'syz' CONT COMP 5 ELEM LECT 1 TERM
COUR 6 'sxz' CONT COMP 6 ELEM LECT 1 TERM
TRAC 1 2 3 AXES 1.0 'CONTR. [PA]' YZER
TRAC 4 5 6 AXES 1.0 'CONTR. [PA]' YZER
COUR 11 'ex' EPST COMP 1 ELEM LECT 1 TERM
COUR 12 'ey' EPST COMP 2 ELEM LECT 1 TERM
COUR 13 'ez' EPST COMP 3 ELEM LECT 1 TERM
COUR 14 'gxy' EPST COMP 4 ELEM LECT 1 TERM
COUR 15 'gyz' EPST COMP 5 ELEM LECT 1 TERM
COUR 16 'gxz' EPST COMP 6 ELEM LECT 1 TERM
TRAC 11 12 13 AXES 1.0 'EPST [-]' YZER
TRAC 14 15 16 AXES 1.0 'EPST [-]' YZER
COUR 21 'ecr1' ECRO COMP 1 ELEM LECT 1 TERM
COUR 22 'ecr2' ECRO COMP 2 ELEM LECT 1 TERM
COUR 23 'ecr3' ECRO COMP 3 ELEM LECT 1 TERM
COUR 24 'ecr4' ECRO COMP 4 ELEM LECT 1 TERM
COUR 25 'ecr5' ECRO COMP 5 ELEM LECT 1 TERM
COUR 26 'ecr6' ECRO COMP 6 ELEM LECT 1 TERM
COUR 27 'ecr1' ECRO COMP 7 ELEM LECT 1 TERM
TRAC 21 AXES 1.0 'ECR1' YZER
TRAC 22 AXES 1.0 'ECR2' YZER
TRAC 23 AXES 1.0 'ECR3' YZER
TRAC 24 AXES 1.0 'ECR4' YZER
TRAC 25 AXES 1.0 'ECR5' YZER
TRAC 26 AXES 1.0 'ECR6' YZER
TRAC 27 AXES 1.0 'ECR7' YZER
FIN

```

## readbrain.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
brain = 'READ' 'brain.read' 'MESH' 'ELEM';
opti sauv form 'brain.msh';
opti trac psc ftra 'brain_mesh.ps';
sauv form brain;
trac cach brain;
fin;

```

## readbrainv.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
vol = 'READ' 'brainvol.read' 'MESH' 'ELEM';
opti sauv form 'brainv.msh';
opti trac psc ftra 'brainv_mesh.ps';
sauv form vol;
trac cach vol;
fin;

```

## readcsf.dgibi

```

opti echo 1;
opti dime 3 elem cub8;
csf = 'READ' 'csf.read' 'MESH' 'ELEM';
opti sauv form 'csf.msh';
opti trac psc ftra 'csf_mesh.ps';
sauv form csf;
trac cach csf;
fin;

```

## readstl2.f

```

PROGRAM readstl2
IMPLICIT NONE
INTEGER, PARAMETER :: NPTMAX = 1000000, NELMAX = 10000000,
  > NNMAX = 8, ESETMAX=5, NSETMAX=5
REAL(8) :: X(NPTMAX), Y(NPTMAX), Z(NPTMAX)
INTEGER :: LABNOD(NPTMAX), LABELM(NELMAX), LRENUMN(NPTMAX),
  > LRENUME(NELMAX)
INTEGER :: IJK(NNMAX,NELMAX), NPTL, NELEM, I, NOD(NNMAX), NND,
  > NSESETS, NNSSETS, LL, N, IVAL(20)
INTEGER :: ESET(NELMAX,ESETMAX), NSET(NPTMAX,NSETMAX),
  > NSESET(ESETMAX), NNSSET(NSETMAX)
CHARACTER*132 :: LINE
CHARACTER*80 :: NAMESET(ESETMAX), NAMNSET(NSETMAX)

*
* initializations
NPTL = 0
NELEM = 0
NNSSETS = 0
NSESETS = 0
NNSSET(:) = 0
NESET(:) = 0
DO I = 1, NPTMAX
  LRENUMN(I) = 0
END DO
DO I = 1, NELMAX
  LRENUME(I) = 0
END DO

*
* read nodes
READ (5,'(A)') LINE
IF (LINE(1:5) /= '*Node') STOP 'BAD FIRST LINE'
1 READ (5,'(A)') LINE
IF (LINE(1:8) == '*Element') GO TO 2
NPTL = NPTL + 1
IF (NPTL > NPTMAX) STOP 'TOO BIG NODE LABEL'
READ (LINE,*) LABNOD(NPTL), X(NPTL), Y(NPTL), Z(NPTL)
LRENUMN(LABNOD(NPTL)) = NPTL
GO TO 1

*
* read elements
2 IF (LINE(9:15) /= ', type=') STOP 'BAD ELEMENT DECLARATION'
IF (LINE(16:17) == 'S3') THEN
  NND = 3
ELSEIF (LINE(16:19) == 'C3D4') THEN
  NND = 4
ELSE
  STOP 'BAD ELEMENT TYPE'
ENDIF
3 READ (5, '(A)', END=100) LINE
IF (LINE(1:6) == '*Elset') GO TO 4
IF (LINE(1:5) == '*Nset') GO TO 9
NELEM = NELEM + 1
IF (NELEM > NELMAX) STOP 'TOO MANY ELEMENTS'
SELECT CASE (NND)
CASE (3)
  READ (LINE,*) LABELM(NELEM), NOD(1), NOD(2), NOD(3)
  LRENUME(LABELM(NELEM)) = NELEM
  IF (MAX(NOD(1), NOD(2), NOD(3)) > NPTMAX)
    > STOP 'TOO BIG NOD INDEX'
  IF (LRENUMN(NOD(1)) > 0) THEN
    IJK(1,NELEM) = LRENUMN(NOD(1))
  ELSE
    STOP 'WRONG NODE INDEX'
  ENDIF
IF (LRENUMN(NOD(2)) > 0) THEN

```

```

      IJK(2,NELEM) = LRENUMN(NOD(2))
    ELSE
      STOP 'WRONG NODE INDEX'
    ENDIF
    IF (LRENUMN(NOD(3)) > 0) THEN
      IJK(3,NELEM) = LRENUMN(NOD(3))
    ELSE
      STOP 'WRONG NODE INDEX'
    ENDIF
CASE (4)
  READ (LINE,*) LABELM(NELEM), NOD(1), NOD(2), NOD(3), NOD(4)
  LRENUME(LABELM(NELEM)) = NELEM
  IF (MAX(NOD(1), NOD(2), NOD(3), NOD(4)) > NPTMAX)
    STOP 'TOO BIG NOD INDEX'
  IF (LRENUMN(NOD(1)) > 0) THEN
    IJK(1,NELEM) = LRENUMN(NOD(1))
  ELSE
    STOP 'WRONG NODE INDEX'
  ENDIF
  IF (LRENUMN(NOD(2)) > 0) THEN
    IJK(2,NELEM) = LRENUMN(NOD(2))
  ELSE
    STOP 'WRONG NODE INDEX'
  ENDIF
  IF (LRENUMN(NOD(3)) > 0) THEN
    IJK(3,NELEM) = LRENUMN(NOD(3))
  ELSE
    STOP 'WRONG NODE INDEX'
  ENDIF
  IF (LRENUMN(NOD(4)) > 0) THEN
    IJK(4,NELEM) = LRENUMN(NOD(4))
  ELSE
    STOP 'WRONG NODE INDEX'
  ENDIF
END SELECT
GO TO 3
*
* read element sets
4 IF (LINE(7:14) /= ' ', elset=') STOP 'BAD ELEMENT SET DECLARATION'
  NESETS = NESETS + 1
  IF (NESETS > ESETMAX) STOP 'TOO MANY ELEMENT SETS'
  LL = LEN_TRIM(LINE)
  NAMESET(NESETS) = LINE(15:LL)
5 READ (5, 'A)', END=100) LINE
  IF (LINE(1:6) == '*elset') GO TO 4
  IF (LINE(1:5) == '*Nset') GO TO 9
  CALL READ_COMMA_SEPARATED (LINE, N, IVAL)
  IF (N > 0) THEN
    DO I = 1, N
      ESET(NESET(NESETS)+I,NESETS) = LRENUME(IVAL(I))
    END DO
    NESET(NESETS) = NESET(NESETS) + N
  ENDIF
  GO TO 5
*
* read node sets
9 IF (LINE(6:12) /= ' ', nset=') STOP 'BAD NODE SET DECLARATION'
  NNSETS = NNSETS + 1
  IF (NNSETS > NSETMAX) STOP 'TOO MANY NODE SETS'
  LL = LEN_TRIM(LINE)
  NAMNSET(NNSETS) = LINE(13:LL)
10 READ (5, 'A)', END=100) LINE
  IF (LINE(1:5) == '*Nset') GO TO 9
  CALL READ_COMMA_SEPARATED (LINE, N, IVAL)
  IF (N > 0) THEN
    DO I = 1, N
      NSET(NNSET(NNSETS)+I,NNSETS) = LRENUMN(IVAL(I))
    END DO
    NNSET(NNSETS) = NNSET(NNSETS) + N
  ENDIF
  GO TO 10
*
100 WRITE (6, '(A6,I8,1X,A6,I8)') '*NPTL=', NPTL, 'NELEM=', NELEM
*
  WRITE (6, '(3(1X,E12.5))') (X(I), Y(I), Z(I), I = 1, NPTL)
  SELECT CASE (NND)
CASE (3)
  WRITE (6, '(3I8)') (IJK(1,I), IJK(2,I), IJK(3,I), I = 1, NELEM)
CASE (4)
  WRITE (6, '(4I8)') (IJK(1,I), IJK(2,I), IJK(3,I), IJK(4,I),
    > I = 1, NELEM)
  END SELECT
*
! WRITE (6,*) '*NESETS=', NESETS
! WRITE (6,*) '*NNSETS=', NNSETS
IF (NESETS > 0) THEN
  WRITE (6,*) 'COMP GROU', NESETS
  DO I = 1, NESETS
    LINE(1:1) = ""
    LINE(2:) = ADJUSTL (TRIM (NAMESET(I)))
    LINE = TRIM (LINE) // "' LECT"
    WRITE (6, '(A)') TRIM (LINE)
    WRITE (6, '(9I8)') ESET(1:NESET(I),I)
    WRITE (6, '(A)') "TERM"
  END DO
ENDIF

```

```

*
IF (NNSETS > 0) THEN
  IF (NESETS == 0) THEN
    WRITE (6,*) 'COMP NGRO', NNSETS
  ELSE
    WRITE (6,*) ' NGRO', NNSETS
  ENDIF
  DO I = 1, NNSETS
    LINE(1:1) = ""
    LINE(2:) = ADJUSTL (TRIM (NAMNSET(I)))
    LINE = TRIM (LINE) // "' LECT"
    WRITE (6, '(A)') TRIM (LINE)
    WRITE (6, '(9I8)') NSET(1:NNSET(I),I)
    WRITE (6, '(A)') "TERM"
  END DO
ENDIF
*
END PROGRAM readst12
*****
SUBROUTINE READ_COMMA_SEPARATED (LINE, N, IVAL)
  IMPLICIT NONE
*
* args
  CHARACTER*132, INTENT(IN) :: LINE
  INTEGER, INTENT(OUT) :: N, IVAL(20)
*
* locals
  INTEGER :: I
*
  N = 0
  IVAL(:) = -HUGE(IVAL)
  READ (LINE, *, END = 10, ERR = 10) IVAL
  10 DO I = 1, 20
    IF (IVAL(I) == -HUGE(IVAL)) EXIT
    N = N + 1
  END DO
*
END SUBROUTINE READ_COMMA_SEPARATED

```

## slice.dgibi

```

'DEBPROC' pxextr3d m*'MALLAGE' 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 'slice_mesh.ps';
opti sauv form 'slice.msh';
opti rest form 'headl.msh';
rest form;
list;
oubl braincsf;
sbrain = pxextr3d brain -6.40E0 -5.80E0 -1000 1000 -1000 1000;
list (nbel sbrain);
trac cach sbrain;
scsf = pxextr3d csf -6.40E0 -5.80E0 -1000 1000 -1000 1000;
list (nbel scsf);
trac cach scsf;
spskull = pxextr3d pskull -6.40E0 -5.80E0 -1000 1000 -1000 1000;
list (nbno spskull);
trac cach spskull;
oubl brain;
oubl csf;
oubl pskull;
shead = sbrain et scsf et spskull;
tass shead noop;
sauv form shead;
trac cach shead;
fin;

```

## List of input files

### B

|              |    |
|--------------|----|
| br2d00.epx   | 50 |
| br2d00a.epx  | 50 |
| br2d01.dgibi | 50 |
| br2d01.epx   | 50 |
| br2d01z.epx  | 51 |
| br3d01.dgibi | 51 |
| br3d01.epx   | 51 |
| br3d01z.epx  | 51 |
| br3d02.dgibi | 52 |
| br3d02.epx   | 52 |
| br3d02z.epx  | 52 |
| brainv.epx   | 52 |
| brat01.epx   | 53 |
| brat02.epx   | 53 |
| brat03.epx   | 53 |
| brat04.epx   | 53 |
| brat05.epx   | 53 |
| brat06.epx   | 54 |
| brat06z.epx  | 54 |
| brat07.epx   | 54 |
| brat08.epx   | 54 |
| brat09.epx   | 55 |
| brat09p.epx  | 55 |
| brat09z.epx  | 55 |
| brat10.epx   | 55 |
| brat11.epx   | 55 |
| brat11p.epx  | 56 |
| brat11z.epx  | 56 |
| brat12.epx   | 56 |
| brat12p.epx  | 56 |
| brat12w.epx  | 57 |
| brat12z.epx  | 57 |
| brat13.epx   | 57 |
| brat13p.epx  | 57 |
| brat13z.epx  | 57 |
| brat21.epx   | 58 |
| brat21c.epx  | 58 |
| brat21z.epx  | 58 |
| brat22.epx   | 58 |
| brat23.epx   | 58 |
| brat23c.epx  | 59 |
| brat23z.epx  | 59 |

### D

|            |    |
|------------|----|
| delt01.epx | 59 |
| delt02.epx | 59 |
| delt03.epx | 59 |
| delt04.epx | 59 |

### G

|                   |    |
|-------------------|----|
| genmesh.dgibi     | 59 |
| genmesh_new.dgibi | 60 |
| genmeshl.dgibi    | 60 |

### H

|              |    |
|--------------|----|
| head46.dgibi | 60 |
| head46.epx   | 61 |
| head46v.epx  | 62 |
| head46w.epx  | 62 |
| head46z.epx  | 62 |
| head47.dgibi | 62 |
| head47.epx   | 64 |
| head47v.epx  | 64 |
| head47w.epx  | 64 |
| head47z.epx  | 65 |
| head48.dgibi | 65 |
| head48.epx   | 66 |
| head48v.epx  | 67 |
| head48w.epx  | 67 |
| head48z.epx  | 67 |
| head55.dgibi | 67 |
| head55.epx   | 68 |
| head56.dgibi | 68 |
| head56.epx   | 69 |
| head66.dgibi | 69 |
| head66.epx   | 71 |
| huma01.epx   | 71 |
| hype01.dgibi | 72 |
| hype01.epx   | 72 |
| hype02.dgibi | 72 |
| hype02.epx   | 72 |
| hype02a.epx  | 72 |
| hype02b.epx  | 73 |
| hype02c.epx  | 73 |
| hype02d.epx  | 73 |
| hype03.epx   | 73 |
| hype03a.epx  | 74 |
| hype04.epx   | 74 |
| hype04a.epx  | 74 |
| hype05.epx   | 74 |
| hype05a.epx  | 75 |

### R

|                  |    |
|------------------|----|
| readbrain.dgibi  | 75 |
| readbrainv.dgibi | 75 |
| readcsf.dgibi    | 75 |
| readst12.f       | 75 |

### S

|             |    |
|-------------|----|
| slice.dgibi | 76 |
|-------------|----|





*Europe Direct is a service to help you find answers  
to your questions about the European Union.*

**Freephone number (\*):**

**00 800 6 7 8 9 10 11**

(\* ) The information given is free, as are most calls (though some operators, phone boxes or hotels may charge you).

More information on the European Union is available on the internet (<http://europa.eu>).

## **HOW TO OBTAIN EU PUBLICATIONS**

### **Free publications:**

- one copy:  
via EU Bookshop (<http://bookshop.europa.eu>);
- more than one copy or posters/maps:  
from the European Union's representations ([http://ec.europa.eu/represent\\_en.htm](http://ec.europa.eu/represent_en.htm));  
from the delegations in non-EU countries ([http://eeas.europa.eu/delegations/index\\_en.htm](http://eeas.europa.eu/delegations/index_en.htm));  
by contacting the Europe Direct service ([http://europa.eu/europedirect/index\\_en.htm](http://europa.eu/europedirect/index_en.htm)) or  
calling 00 800 6 7 8 9 10 11 (freephone number from anywhere in the EU) (\*).

(\* ) The information given is free, as are most calls (though some operators, phone boxes or hotels may charge you).

### **Priced publications:**

- via EU Bookshop (<http://bookshop.europa.eu>).

## JRC Mission

As the science and knowledge service of the European Commission, the Joint Research Centre's mission is to support EU policies with independent evidence throughout the whole policy cycle.



**EU Science Hub**  
[ec.europa.eu/jrc](https://ec.europa.eu/jrc)



@EU\_ScienceHub



EU Science Hub - Joint Research Centre



Joint Research Centre



EU Science Hub

