CASTEM 2000 Interfacing Capabilities

Luca Massidda - W&S - CRS4

October 29, 1999

Contents

1 Introduction

In this set of examples we want to show the CASTEM 2000 capabilities to interface with other finite element analysis or physical simulation software.
CASTEM2000 is a computer code for the analysis of structures by the finite element method (FEM). The program was developed by the Department of Mechanics and Tecnology (DMT) of the French Atomic Energy Commission (CEA), it is an high level instrument, that can be used as a support in the design and analysis of components in the nuclear field as in the traditiona industrial sector. CASTEM 2000 is a complete system integrating not only the functions of calculation (processing), but also for the construction of the model (pre-processing) and for the evaluation of the results (post-processing). CASTEM 2000 can analyze linear or non-linear, static or dynamic problems in structural mechanics, heat trasfer and so on.
The main difference between CASTEM and other Finite Element computer programs is in that, while the latter allow the user to solve a certain number of well defined but fixed problems, in CASTEM the user is free to define his problem as he likes, since the program gives him all of the instruments the Finite Element Method has and is free to use them for the definition and solution of the problem, not being forced to any standard solution [1].
This is one of the major advantages of the CASTEM FEA package, this is possible due to its extreme flexibility, and to the possibility for the user to manipulate freely objects as meshes and nodal or elemental fields.
The interface between CASTEM and the other programs can be obtained through its capability to import and export UCD file.
UCD is for Unstructured Cell Data, it is a text file format independent from the Finite Element program used, with which is possible to exchange data and results, for example is one of the file formats used in AVS.
AVS is a powerful and flexible tool for the interactive visualization of complex 3D data, its complexity may discourage the occasional or first -time user since its interface has to be visually programmed for each visualization task the user has. [3]
Although CASTEM 2000 has a visualization tool by itself, it currently does not cover efficiently the complete spectrum of visualizations that may be useful in large and complex 3d industrial applications. Therefore AVS may be used to make the post-processing of CASTEM results. [2]
A UCD file contains the information on the mesh of the model, with a list of the nodes with their nodal coordinates and a list of the elements defined with their type and the connectivity with the nodes. Moreover at the and of the file, one or more nodal or elemental fields may be defined, with a list of the nodal (or element) numbers with one ore more scalar quantities associated with it. These quantities and these fields may be the result of a Finite Element Analysis performed inside CASTEM or a similar program. Therefore they may contain information on the

nodal displacements in a particular direction or the stress field in the structure or the temperature field, or may come as a result from another simulation package and be used inside CASTEM as an input.
For example, the result of the calculation of the convection coefficient on a surface, performed with a CFD program may be used as an input for a thermal-mechanic calculation inside CASTEM.
In this report four examples will be described, trying to show not only how to export data from CASTEM to AVS for the post-processing, but also hot to use the UCD files as an interface with other programs, for the pre-processing of the model and to enhance the analysis capabilities of CASTEM.
It will be shown how to treat the objects that can be read from an UCD file, as well as the objects generated by CASTEM itself, to overcome its limits in pre-processing and to import the results of other simulation software.
The listings are reported in the Appendix.

2 Preprocessing in CASTEM

The first program may be useful to demonstrate the capabilities of CASTEM to mesh a structure with the finite element paradigm. In CASTEM the mesh has to be "guided" on the lines, surfaces and volumes, composing the physical domain of the model. The definition of the elements in which a structure has to be divided must be guided by the operator, who has to define for example the number of nodes or elements on a line or how they are distributed on a surface. This practice may appear tedious, but is essential for the quality of the results that may be obtained with a finite element program. No automatic mesher may obtain the same quality of results of a skilled finite element user.
One of the most interesting things is that the model and the mesh may be fully defined with a set of parameters, and the model may be regenerated immediately when a dimension has to be changed or is necessary to perform a more accurate analysis with a finer mesh. It is not necessary to redraw the model, it is only necessary to modify the parameters used to describe the structure.
The model in the example is a simple "shell" structure, symmetric with respect to a central axis. It is formed from a hemispherical shell with an internal radius RI and an external radius RE connected to a portion of a cylinder of length LAMBDA. The two surfaces may not have the same center, therefore the thickness of the hemispherical shell may not be necessarily constant. The distance on the axis between the two centers is DELTA.

```
RI = 200.E-3;
RE = 210.E-3;
DELTA = -5.E-3;
LAMBDA = -200.E-3;
```

The volume of this structure is described with CUB8 elements, a brick with 8 nodes, one at each corner.
One quarter of the structure is modeled taking advantage of the symmetry of the structure and the loads that will be applied.
The volume of the model must be defined starting from the surfaces containing it, these in turn must be defined starting from the lines at their contour and the lines and the arcs are drawn knowing the position of the start and final points.
Therefore it is first necessary to define a number of characteristic points in the structure (A1, B1, ...) then define the arcs and lines (AB1, BC1, ...) and the the surfaces (SURFI1,SURFI2,...).

> A1 = RI 0. 0.;
> B1 = (RI * C45) (RI * S45) 0.;
> ...
> AB1 = **CERCLE** N1 A1 O1 B1;
> BC1 = **CERCLE** N1 B1 O1 C1;
> ...
> SURFI1 = **DALLER** AB1 BG1 GF1 FA1 SPHER O1;

It could be noted the the point G1 and G2 in the middle of the spherical surfaces are not strictly necessary to define the surfaces. This point is used to divide the quarter of the hemisphere in in three different surfaces. So that these smaller ones could be described with a contour formed by four lines and be easily meshed with quadrilaterals. If this division shouldn't be performed the surface could be described with three arcs and its mesh could be made well using triangular element and really poorly using quadrilateral elements.
Once the internal and external surfaces are defined, the volume VOL1 is defined extruding the internal surface to the external surface and creating a certain number of planes in the middle. The number of elements in the structure is fixed by the N1 and N2 parameters changing them is possible to obtain a coarser or finer mesh.

> VOL1 = SURFI **VOLU** N2 SURFE;

Finally, for the reasons that will be clearer in the description of the other programs other surfaces are created, SURFX, SURFY and SURFZ, starting from the lines at the contour of the volume. These surfaces were not necessary to describe the volume and once created do not form a part of the volume. This is to say that are described by different nodes then those used in the volume, with different node number but with the same nodal coordinates.

> SURFYEF = EF1 **REGLER** N2 EF2;
> SURFYFA = FA1 **REGLER** N2 FA2;
> ...

The processor do not know that we want those surface to be part of the volume previously defined, so they are basically treated as if they do not have any connection with the rest of the mesh.
To override this the ELIM procedure is used: the nodes of the surfaces are compared with those of the volume and if the distance between two nodes is smaller than a fixed value the node in the surface is suppressed and replaced with the one belonging to the volume, so the geometrical objects, are associated to the volume even if these surfaces were not used to define it.

ELIM (RI / 10000) SURFX VOL1;
...

Finally the program saves all the geometrical objects, and not only the volume in different UCD files.

OPTION SORTIR FINVOL;
SORTIR AVS VOL1;
OPTION SORTIR FINSURX;
SORTIR AVS SURFX;
...

Saving only the main mesh, the one describing the volume,only the geometrical information on the volume would be saved and the information on the surfaces surrounding it would be lost.

3 Importing meshes

In the second program a simple thermal analysis is performed on the model defined by the previous program. Of course the instructions of this program may be attached directly to the previous one in which the mesh had been described, but the scope of the program is to show how to import in CASTEM meshes and data from other programs (or from CASTEM itself) and how to manipulate it. Why doing that? Because the data that may be imported might be the result of some other simulation software, and because sometimes it could be useful to make a mesh in some other program. For example, when a CAD drawing is ready to be meshed, when a mesh is coming from another program where it was used for some other analysis, or when a complex volume mesh has to be drawn, "manual" meshing in CASTEM is practically impossible and an automated mesher or a mesh optimizer is necessary. When CASTEM reads from an UCD-AVS file stores the data in an object of TABLE type, it is a structured object with different sub-objects contained in it. In this case the mesh is stored in the LEMAILLA field, and copied in a MESH object, as VOL1 or SURFX, the names are the same of the previous program but of course this is not a necessity.

OPTION LECTURE FINVOL;
TABL = **LIRE** AVS;
VOL1 = TABL.LEMAILLA;

OPTION LECTURE FINSURI;
TABL = **LIRE** AVS;
SURFI = TABL.LEMAILLA; ...

If now the program draws the VOL1 object and we look at the geometrical properties associated to it, we will see that only the volume geometry is associated with the mesh [Fig. 1]. Drawing together VOL1, SURFI, SURFE, ... we will

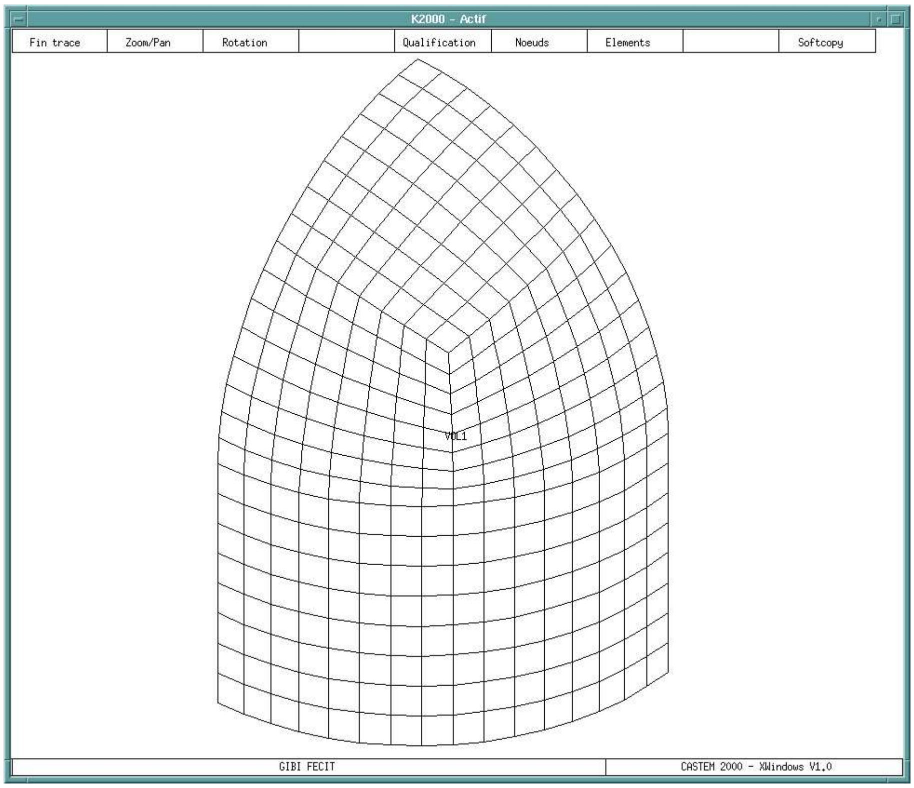

Figure 1: Geometry before ELIM

see together all the geometrical objects, but we will see that these objects do not have common nodes and are not connected one to the other.
To associate again the geometrical properties to the model is necessary that the different geometrical object share the same nodes, so with the ELIM command the nodes of the different surfaces are suppressed and replaced with the nodes of the mesh on the volume.

ELIM PAR1 SURFI VOL1;

ELIM PAR1 SURFE VOL1;
...

If now the program draws VOL1 again and we look at the geometrical properties associated to it we will see that not only the volume but all of the surrounding surfaces are included [Fig. 2]. All of the operations that could be done on the

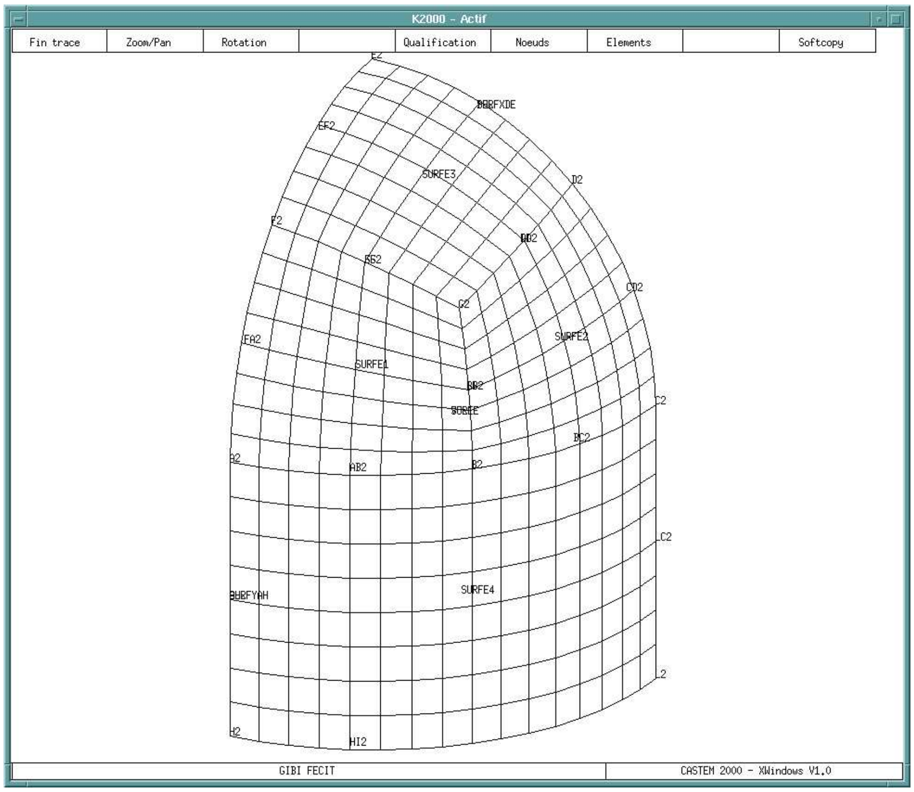

Figure 2: Geometry after ELIM

VOL1 will be done to its nodes and therefore to the nodes in common with the surfaces, and all the operations we could do on the surfaces will be done to the volume through the shared nodes. For example in this program the thermal isotropic model defined on VOL1 and the convection models defined on SURFI and SURFE will share the same nodes and will be connected.

MOD1 = **MODEL** VOL1 **THERMIQUE ISOTROPE;**
MOD2 = **MODEL** SURFI **CONVECTION**;
MOD3 = **MODEL** SURFE **CONVECTION**;

The analysis performed calculates the temperature field in the structure with convection at the internal and at the external surface with ambient temperatures of 600 inside and 100 outside. The result is a nodal field, since temperature is

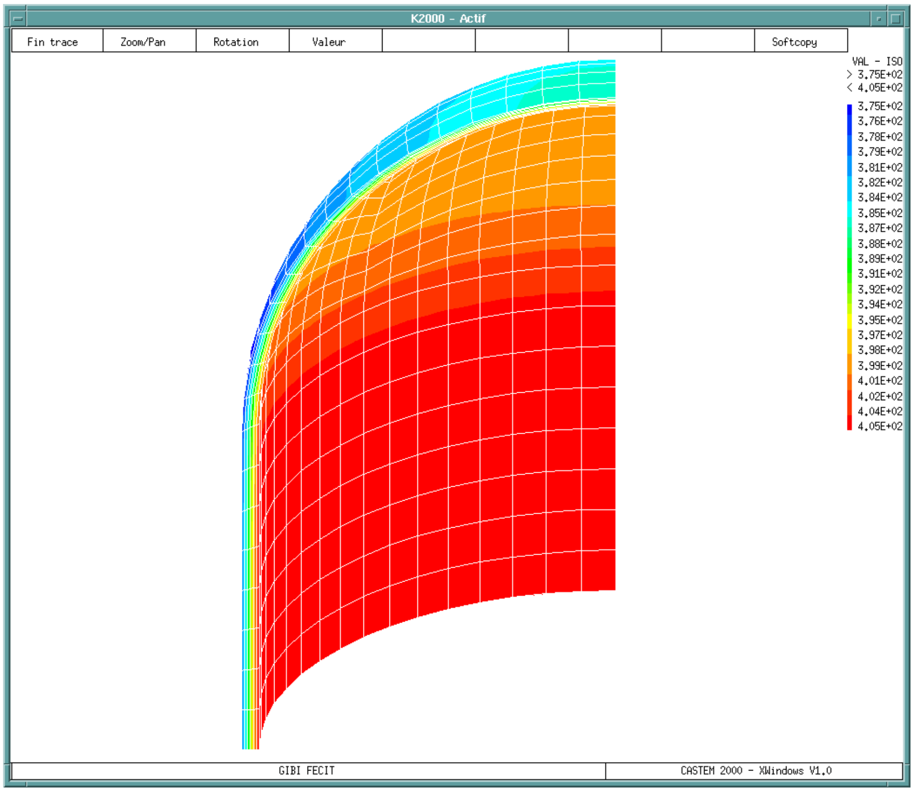

Figure 3: Temperature field

defined at each node of the structure [Fig. 3]. This field is exported, together with the necessary supporting mesh in a new UCD-AVS file.

OPTION SORTIR FINTEMP;
SORTIR AVS VOL1 TEMP1;

4 Importing data

In the first mechanical analysis program the mesh geometry generated by the first program described is imported and used to create a mechanical model.
The procedure used is the same described in the previous example: consists in loading the different meshes separately and assembling them in a unique mesh with all of the geometrical characteristics connected to it.
In this program is also imported the nodal temperature field generated with the program previously described.

OPTION LECTURE FINTEMP;
TABL = **LIRE** AVS;

```
CHT1 = TABL.LECHPOIN;
PUNT1 = TABL.MAILSUPP;
```

In this case the field has been generated by CASTEM, but the more interesting feature we want to stress out is that this field could have been generated in any other way, it is just a mesh with the definition of a scalar quantity at each node. For the scalar field the same considerations applied previously for the geometric objects apply, so the nodal field is defined generally speaking on a different mesh with different nodes then VOL1, using the same ELIM command is possible to suppress the nodes of the mesh on which the field is defined and to transfer this nodal field on the nodes of the VOL1 object.

ELIM PAR1 PUNT1 VOL1;

The mechanical analysis is quite simple, the material has a linear elastic definition with constant mechanical properties.
The loads are due to the thermal deformations that are not uniform in the structure, generally speaking any heat flow can generate a stress field in a structure. The procedure to calculate the stresses due to the thermal deformations is to reduce the problem to a simple static one.
First the stress field in an unconstrained structure is derived:

SIGT1 = **THET** MOD1 MAT1 CHT1;

Then from this nodal field, another nodal field is derived containing the nodal forces that are necessary to create that state of stress.

FT1 = **BSIGMA** MOD1 SIGT1;

Then the structure is constrained and the nodal displacements are calculated when the nodal loads previously calculated are applied.

DEP1 = **RESO** RIGCL FT1;

It is now possible to calculate the geometry of the deformed structure and to calculate the stress field, the correct stress field is obtained subtracting to this the one previously calculate for an unconstrained structure.

S2 = **SIGMA** MOD1 MAT1 DEP1;
S1 = S2 - SIGT1;

For the stress is shown how to manipulate the fields of the structure to obtain new and undefined field directly inside the program.
Here from the element field containing all the stress components. Are derived different element fields each with a different stress component.

SEXX = **EXCO** SMXX S1;
...

Then each of these element fields are converted to a nodal field (it is not necessary, it is just to show how to do).

SPXX = **NOMC** S (**CHANGER CHPO** MOD1 SEXX);
...

Finally from these different nodal fields, each of them lying on the same mesh a new nodal field is derived, with the Von Mises equivalent stress, not included among the standard outputs of the program [Fig. 4].

SPT1 = ((SPXX - SPYY)**2) + ((SPYY - SPZZ)**2) + ((SPZZ - SPXX)**2);
SPT2 = (SPXY**2) + (SPYZ**2) + (SPXZ**2);
SPVM = ((SPT1 + (6 * SPT2))**0.5)/(2.**0.5);

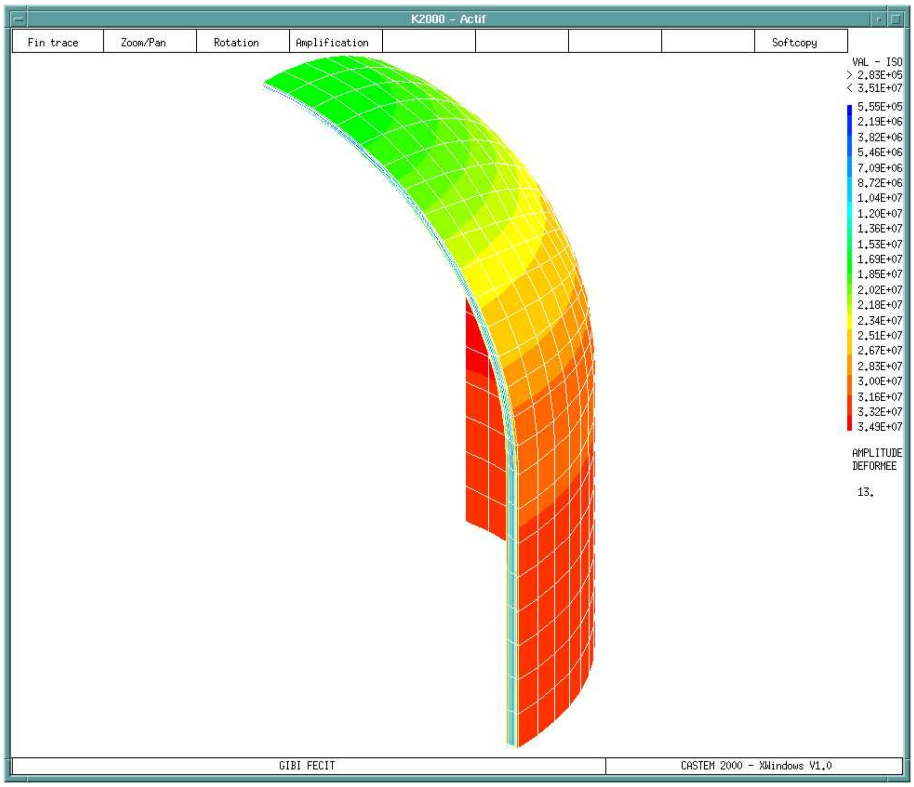

Figure 4: Von Mises equivalent stress

This result can evidently be exported with the same procedure used for the nodal temperature field of the previous program.

5 Manipulating fields

Finally in the last program is described one of the more interesting capabilities of the CASTEM program, that is its ability to manipulate and use nodal and element field, and to be easily programmed for new and different tasks.
The program is essentially identical to the previous one except that here the modulus of elasticity is supposed to vary with the temperature.
Here as in most of the other finite element programs is possible to define a law of behavior of the elasticity modulus with respect to the temperature for example. Normally this behavior is defined in standard linear or polynomial laws; or it is defined with a set of points and the real values are extrapolated from that not very accurately.
The difference is that in CASTEM the modulus of elasticity E as well as any other characteristic of the material is simply an element field and has all of the possibilities of this kind of object.
Therefore what is done is to derive a new nodal field manipulating the nodal temperature field imported in the program, convert it in the corresponding element field and simply use it in the definition of the properties of the material.

CHPEY1 = 200.E9*(CHT1-(**MANU CHPO** VOL1 1 T 375.));
CHPEY2 = 100.E9*(CHT1 - (**MANU CHPO** VOL1 1 T 400.));
CHPEY3 = (CHPEY1 - CHPEY2)/(400. - 375.);

Here the behavior law is simple (linear), but can be what anyone wants. Moreover it is not fixed the nature and the number of the parameters from which the element field depends, so a lot of really complicated behavior laws for a material can be obtained, and the description of the material can be as accurate as one wants, without any unnecessary simplification forced by the limitations of the program used.
The results of a mechanical analysis are quite different of course [Fig. 5].

The possibility to import any scalar field in the model, show the possibility to describe the behavior of a material depending on parameters that can not be calculated inside the software or can more easily be calculated with other dedicated package, for example describing the damage in the material or the modification in the behavior laws of the material from the exposition of the structure to a light source, for some plastics; or to the neutron irradiation in nuclear engineering applications.

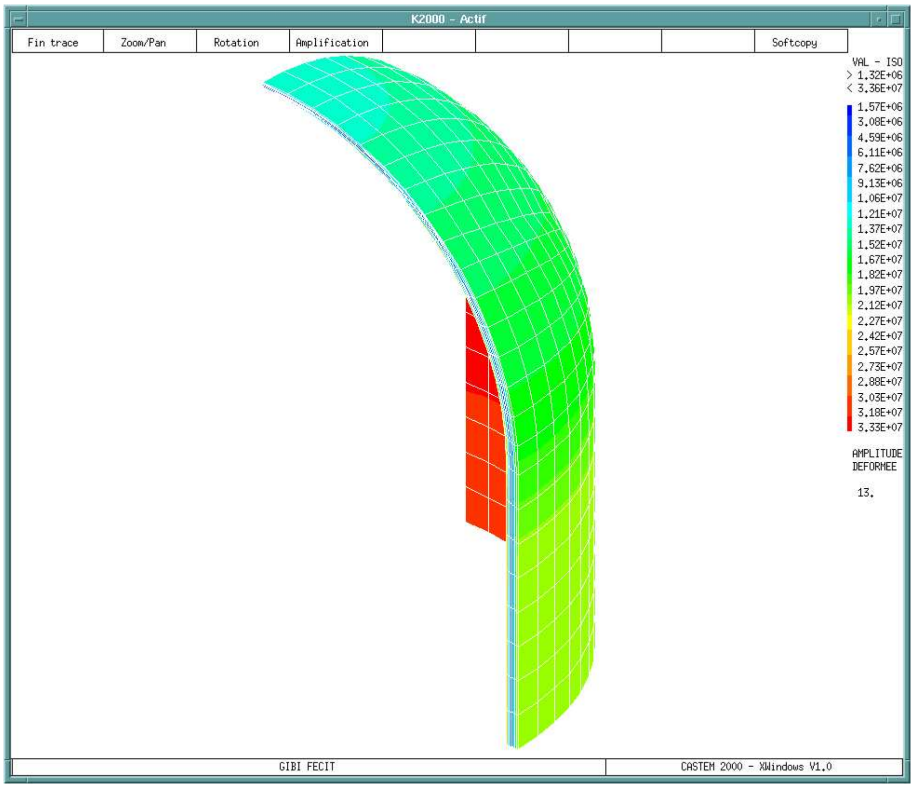

Figure 5: Von Mises equivalent stress

A Program 1

```
OPTI LANG ANGLAIS;
OPTI DIME 3 ELEM CUB8;
RI = 200.E-3;
RE = 210.E-3;
DELTA = -5.E-3;
LAMBDA = -200.E-3;
VEC1 = 0. 0. (LAMBDA + DELTA);
VEC2 = 0. 0. LAMBDA;
S45 = SIN 45.;
C45 = COS 45.;
S30 = SIN 30.;
C30 = COS 30.;
N1 = 8;
N2 = 6;

A1 = RI 0. 0.;
B1 = (RI * C45) (RI * S45) 0.;
C1 = 0. RI 0.;
D1 = 0. (RI * C45) (RI * S45);
E1 = 0. 0. RI;
F1 = (RI * C45) 0. (RI * S45);
G1 = (RI * C30 * C45) (RI * C30. * S45) (RI * S30);
H1 = RI 0. (LAMBDA + DELTA);
I1 = (RI * C45) (RI * S45) (LAMBDA + DELTA);
L1 = 0. RI (LAMBDA + DELTA);
O1 = 0. 0. 0.;
AB1 = CERCLE N1 A1 O1 B1;
BC1 = CERCLE N1 B1 O1 C1;
CD1 = CERCLE N1 C1 O1 D1;
DE1 = CERCLE N1 D1 O1 E1;
EF1 = CERCLE N1 E1 O1 F1;
FA1 = CERCLE N1 F1 O1 A1;
FG1 = CERCLE N1 F1 O1 G1;
GD1 = CERCLE N1 G1 O1 D1;
GB1 = CERCLE N1 G1 O1 B1;
GF1 = INVE FG1;
DG1 = INVE GD1;
BG1 = INVE GB1;
AH1 = DROITE N1 A1 H1;
LC1 = DROITE N1 L1 C1;
HI1 = AB1 PLUS VEC1;
IL1 = BC1 PLUS VEC1;

SURFI1 = DALLER AB1 BG1 GF1 FA1 SPHER O1;
SURFI2 = DALLER BC1 CD1 DG1 GB1 SPHER O1;
SURFI3 = DALLER DE1 EF1 FG1 GD1 SPHER O1;
SURFI4 = (AB1 ET BC1) TRAN N1 VEC1;


A2 = RE 0. DELTA;
B2 = (RE * C45) (RE * S45) DELTA;
C2 = 0. RE DELTA;
D2 = 0. (RE * C45) ((RE * S45) + DELTA);
E2 = 0. 0. (RE + DELTA);
F2 = (RE * C45) 0. ((RE * S45) + DELTA);
G2 = (RE * C30 * C45) (RE * C30 * S45) ((RE * S30) + DELTA);
H2 = RE 0. (LAMBDA + DELTA);
I2 = (RE * C45) (RE * S45) (LAMBDA + DELTA);
L2 = 0. RE (LAMBDA + DELTA);
O2 = 0. 0. DELTA;
AB2 = CERCLE N1 A2 O2 B2;
BC2 = CERCLE N1 B2 O2 C2;
CD2 = CERCLE N1 C2 O2 D2;
DE2 = CERCLE N1 D2 O2 E2;
```

```
EF2 = CERCLE N1 E2 O2 F2;
FA2 = CERCLE N1 F2 O2 A2;
FG2 = CERCLE N1 F2 O2 G2;
GD2 = CERCLE N1 G2 O2 D2;
GB2 = CERCLE N1 G2 O2 B2;
GF2 = INVE FG2;
DG2 = INVE GD2;
BG2 = INVE GB2;
AH2 = DROITE N1 A2 H2;
LC2 = DROITE N1 L2 C2;
HI2 = AB2 PLUS VEC2;
IL2 = BC2 PLUS VEC2;

SURFE1 = DALLER AB2 BG2 GF2 FA2 SPHER O2;
SURFE2 = DALLER BC2 CD2 DG2 GB2 SPHER O2;
SURFE3 = DALLER DE2 EF2 FG2 GD2 SPHER O2;
SURFE4 = (AB2 ET BC2) TRAN N1 VEC2;

SURFI = SURFI1 ET SURFI2 ET SURFI3 ET SURFI4;
SURFE = SURFE1 ET SURFE2 ET SURFE3 ET SURFE4;
VOL1 = SURFI VOLU N2 SURFE;

SURFYEF = EF1 REGLER N2 EF2;
SURFYFA = FA1 REGLER N2 FA2;
SURFYAH = AH1 REGLER N2 AH2;
SURFXLC = LC1 REGLER N2 LC2;
SURFXCD = CD1 REGLER N2 CD2;
SURFXDE = DE1 REGLER N2 DE2;
SURFZHI = HI1 REGLER N2 HI2;
SURFZIL = IL1 REGLER N2 IL2;
SURFY = SURFYEF ET SURFYFA ET SURFYAH;
SURFX = SURFXLC ET SURFXCD ET SURFXDE;
SURFZ = SURFZHI ET SURFZIL;
ELIM (RI / 10000) SURFX VOL1;
ELIM (RI / 10000) SURFY VOL1;
ELIM (RI / 10000) SURFZ VOL1;

OPTION SORTIR FINVOL;
SORTIR AVS VOL1;
OPTION SORTIR FINSURX;
SORTIR AVS SURFX;
OPTION SORTIR FINSURY;
SORTIR AVS SURFY;
OPTION SORTIR FINSURZ;
SORTIR AVS SURFZ;
OPTION SORTIR FINSURI;
SORTIR AVS SURFI;
OPTION SORTIR FINSURE;
SORTIR AVS SURFE;
```

B Program 2

```
OPTION LANG ANGLAIS;
OPTION DIME 3 ELEM CUB8;

PAR1 = 0.0001;

OPTION LECTURE FINVOL;
TABL = LIRE AVS;
VOL1 = TABL.LEMAILLA;

OPTION LECTURE FINSURI;
TABL = LIRE AVS;
SURFI = TABL.LEMAILLA;

OPTION LECTURE FINSURE;
TABL = LIRE AVS;
SURFE = TABL.LEMAILLA;

OPTION LECTURE FINSURX;
TABL = LIRE AVS;
SURFX = TABL.LEMAILLA;

OPTION LECTURE FINSURY;
TABL = LIRE AVS;
SURFY = TABL.LEMAILLA;

OPTION LECTURE FINSURZ;
TABL = LIRE AVS;
SURFZ = TABL.LEMAILLA;

ELIM PAR1 SURFI VOL1;
ELIM PAR1 SURFE VOL1;
ELIM PAR1 SURFX VOL1;
ELIM PAR1 SURFY VOL1;
ELIM PAR1 SURFZ VOL1;

MOD1 = MODEL VOL1 THERMIQUE ISOTROPE;
MOD2 = MODEL SURFI CONVECTION;
MOD3 = MODEL SURFE CONVECTION;

MAT1 = MATER MOD1 K 52.;
MAT2 = MATER MOD2 H 750.;
MAT3 = MATER MOD3 H 500.;

FLU1 = CONV MOD2 MAT2 T 600.;
FLU2 = CONV MOD3 MAT3 T 100.;

COND1 = COND MOD1 MAT1;
COND2 = COND MOD2 MAT2;
COND3 = COND MOD3 MAT3;
CONDTOT = COND1 ET COND2 ET COND3;

FLUTOT = FLU1 ET FLU2;

TEMP1 = RESOU CONDTOT FLUTOT;

PDV = -1000. 0. 500.;
TRAC CACH PDV VOL1 TEMP1;

OPTION SORTIR FINTEMP;
SORTIR AVS VOL1 TEMP1;
```

C Program 3

```
OPTION LANG ANGLAIS;
OPTION DIME 3 ELEM CUB8;

PAR1 = 0.0001;

OPTION LECTURE FINVOL;
TABL = LIRE AVS;
VOL1 = TABL.LEMAILLA;

OPTION LECTURE FINTEMP;
TABL = LIRE AVS;
CHT1 = TABL.LECHPOIN;
PUNT1 = TABL.MAILSUPP;

OPTION LECTURE FINSURI;
TABL = LIRE AVS;
SURFI = TABL.LEMAILLA;

OPTION LECTURE FINSURE;
TABL = LIRE AVS;
SURFE = TABL.LEMAILLA;

OPTION LECTURE FINSURX;
TABL = LIRE AVS;
SURFX = TABL.LEMAILLA;

OPTION LECTURE FINSURY;
TABL = LIRE AVS;
SURFY = TABL.LEMAILLA;

OPTION LECTURE FINSURZ;
TABL = LIRE AVS;
SURFZ = TABL.LEMAILLA;

ELIM PAR1 SURFI VOL1;
ELIM PAR1 SURFE VOL1;
ELIM PAR1 SURFX VOL1;
ELIM PAR1 SURFY VOL1;
ELIM PAR1 SURFZ VOL1;

MOD1 = MODEL VOL1 MECANIQUE ELASTIQUE ISOTROPE;
MAT1 = MATER MOD1 YOUN 200.E9 NU 0.3 RHO 7860. ALPHA 1.E-5;

SIGT1 = THET MOD1 MAT1 CHT1;
FT1 = BSIGMA MOD1 SIGT1;

CLX = BLOQ SURFX UX;
CLY = BLOQ SURFY UY;
CLZ = BLOQ SURFZ UZ;
CLTOT = CLX ET CLY ET CLZ;

RIG1 = RIGIDITE MOD1 MAT1;
RIGCL = RIG1 ET CLTOT;
DEP1 = RESO RIGCL FT1;

DEF0 = DEFO VOL1 DEP1 0. BLAN;
DEF1 = DEFO VOL1 DEP1 BLEU;

S2 = SIGMA MOD1 MAT1 DEP1;
S1 = S2 - SIGT1;

SEXX = EXCO SMXX S1;
SEYY = EXCO SMYY S1;
SEZZ = EXCO SMZZ S1;
SEXY = EXCO SMXY S1;
SEXZ = EXCO SMXZ S1;
SEYZ = EXCO SMYZ S1;
SPXX = NOMC S (CHANGER CHPO MOD1 SEXX);
SPYY = NOMC S (CHANGER CHPO MOD1 SEYY);
SPZZ = NOMC S (CHANGER CHPO MOD1 SEZZ);
```

```
SPXY = NOMC S (CHANGER CHPO MOD1 SEXY);
SPXZ = NOMC S (CHANGER CHPO MOD1 SEXZ);
SPYZ = NOMC S (CHANGER CHPO MOD1 SEYZ);

SPT1 = ((SPXX - SPYY)**2) + ((SPYY - SPZZ)**2) + ((SPZZ - SPXX)**2);
SPT2 = (SPXY**2) + (SPYZ**2) + (SPXZ**2);
SPVM = ((SPT1 + (6 * SPT2))**0.5)/(2.**0.5);

TRAC CACH DEF1 SPVM;
```

D Program 4

```
OPTION LANG ANGLAIS;
OPTION DIME 3 ELEM CUB8;

PAR1 = 0.0001;

OPTION LECTURE FINVOL;
TABL = LIRE AVS;
VOL1 = TABL.LEMAILLA;

OPTION LECTURE FINTEMP;
TABL = LIRE AVS;
CHT1 = TABL.LECHPOIN;
PUNT1 = TABL.MAILSUPP;

OPTION LECTURE FINSURI;
TABL = LIRE AVS;
SURFI = TABL.LEMAILLA;

OPTION LECTURE FINSURE;
TABL = LIRE AVS;
SURFE = TABL.LEMAILLA;

OPTION LECTURE FINSURX;
TABL = LIRE AVS;
SURFX = TABL.LEMAILLA;

OPTION LECTURE FINSURY;
TABL = LIRE AVS;
SURFY = TABL.LEMAILLA;

OPTION LECTURE FINSURZ;
TABL = LIRE AVS;
SURFZ = TABL.LEMAILLA;

ELIM PAR1 PUNT1 VOL1;
ELIM PAR1 SURFI VOL1;
ELIM PAR1 SURFE VOL1;
ELIM PAR1 SURFX VOL1;
ELIM PAR1 SURFY VOL1;
ELIM PAR1 SURFZ VOL1;

CHPEY1 = 200.E9*(CHT1-(MANU CHPO VOL1 1 T 375.));
CHPEY2 = 100.E9*(CHT1 - (MANU CHPO VOL1 1 T 400.));
CHPEY3 = (CHPEY1 - CHPEY2)/(400. - 375.);

MOD1 = MODEL VOL1 MECANIQUE ELASTIQUE ISOTROPE;

CHMEY1 = CHANGER CHAM (NOMC YOUN CHPEY3) MOD1;

MAT1 = MATER MOD1 YOUN CHMEY1 NU 0.3 RHO 7860. ALPHA 1.E-5;

SIGT1 = THET MOD1 MAT1 CHT1;
FT1 = BSIGMA MOD1 SIGT1;

CLX = BLOQ SURFX UX;
CLY = BLOQ SURFY UY;
CLZ = BLOQ SURFZ UZ;
CLTOT = CLX ET CLY ET CLZ;

RIG1 = RIGIDITE MOD1 MAT1;
RIGCL = RIG1 ET CLTOT;
DEP1 = RESO RIGCL FT1;

DEF0 = DEFO VOL1 DEP1 0. BLAN;
DEF1 = DEFO VOL1 DEP1 BLEU;

S2 = SIGMA MOD1 MAT1 DEP1;
S1 = S2 - SIGT1;

SEXX = EXCO SMXX S1;
SEYY = EXCO SMYY S1;
SEZZ = EXCO SMZZ S1;
```

```
SEXY = EXCO SMXY S1;
SEXZ = EXCO SMXZ S1;
SEYZ = EXCO SMYZ S1;
SPXX = NOMC S (CHANGER CHPO MOD1 SEXX);
SPYY = NOMC S (CHANGER CHPO MOD1 SEYY);
SPZZ = NOMC S (CHANGER CHPO MOD1 SEZZ);
SPXY = NOMC S (CHANGER CHPO MOD1 SEXY);
SPXZ = NOMC S (CHANGER CHPO MOD1 SEXZ);
SPYZ = NOMC S (CHANGER CHPO MOD1 SEYZ);

SPT1 = ((SPXX - SPYY)**2) + ((SPYY - SPZZ)**2) + ((SPZZ - SPXX)**2);
SPT2 = (SPXY**2) + (SPYZ**2) + (SPXZ**2);
SPVM = ((SPT1 + (6 * SPT2))**0.5)/(2.**0.5);

TRAC CACH DEF1 SPVM;
```

References

[1] CASTEM Organization: *http://www.castem.org.8001/*.

[2] F. Casadei, L. Papa, P. Pegon *Use of AVS as a Visualization Tool for the Postprocessing of PLEXIS-3c and CASTEM 2000 Finite Element Results*, Technical note no. I.96.206, Joint Research Centre, Ispra(VA) - Italy Nov. 1996.

[3] Advanced Visual Systems, Inc. : *AVS User's Guide*, Release 4, May 1992.