DOCUMENTATION Abaqus2Matlab

Contents

1. Introduction

Abaqus2Matlab is a Matlab toolbox which is used to retrieve the results of an Abaqus analysis in an easy to handle form. It is developed by George Papazafeiropoulos (gpapazafeiropoulos@yahoo.gr) in an effort to facilitate the process of coupling between Abaqus and Matlab. It is written in MATLAB programming language and is available as source code distributed under a BSD-style license (see License.txt which is included in the toolbox folder).

2. Main features and characteristics

Abaqus2Matlab is an effective tool with the following features:

2.1. It provides linking between Abaqus and Matlab. Abaqus analysis can be conducted through Matlab, without interacting with Abaqus/CAE interface, or even Abaqus/Command.

2.2. It transfers efficiently results from Abaqus to Matlab, in an error-proof way, since every contained external function is verified by its application in reading the results of a corresponding Abaqus analysis. The results of the verification of each function are presented in this toolbox in the form of html files.

2.3. It provides the requested results in a form that enables the user to easily manipulate the data for further postprocessing.

2.4. It can read 24 different kinds of nodal results (results at nodes), 35 different kinds of elemental results (results at the element integration points or results regarding whole elements) and 3 different kinds of analysis results (e.g. node definitions, element connectivity, eigenfrequencies and eigenvalues, etc.)

2.5. A complete documentation package is provided along with the source code in this toolbox.

2.6. It covers most types of Abaqus analyses and results. A sufficient number of functions is included in the toolbox to capture the most usually requested Abaqus results.

3. Setup all files and folders

All files and folders of Abaqus2Matlab toolbox have to be setup in the current folder of Matlab, which must be the folder of the toolbox. This folder should be placed in the Abaqus working directory, although this is not mandatory. In any case, the files generated in Abaqus runs will be placed one level up (outside) from the toolbox folder.

3.1. Find the directory containing this file

```
S = mfilename('fullpath');
namelength=numel('Documentation');
S=S(1:end-1-namelength);
```

3.2. Setup all files and folders inside the directory where Abaqus2Matlab toolbox is found

```
addpath(genpath(S));
cd(S);
savepath
```

4. Source code files

The source code files and folders used in this toolbox are the following:

4.1. A function named Fil2str that converts the contents of the results file into a one-row string to be further used in Matlab. This conversion is necessary because the results file is written as a sequential file, i.e. all words in the results file are of the same length (all rows in the file have the same length). Details

4.2. A folder named **OutputAnalysis** which contains the functions for the processing of the analysis results (e.g. node definitions, element connectivity, eigenfrequencies and eigenvalues, etc). See Analysis result types to find which record key and which function is associated with each of the possible analysis result type and List of functions used for any file output request

4.3. A folder named **OutputNodes** which contains the functions for the processing of the nodal results. See Node result types to find which record key, which output variable identifier and which function is associated with each of the possible nodal result types and List of functions used for any node file output request

4.4. A folder named **OutputElements** which contains the functions for the processing of the element results (results at the element integration points or results regarding whole elements). See Element result types to find which record key, which output variable identifier and which function is associated with each of the possible element result types and List of functions used for any element file output request.

4.5. This script (Documentation.m).

5. Verification files

All the functions provided with this toolbox and associated with obtaining analysis, element or node results are verified to ensure that they work correctly and they are not error-prone. In the verification process a suitable Abaqus input file, in which the option for the extraction of the desired results in an ascii results file (.fil) is specified, is run by Abaqus, after being copied from the **AbaqusInputFiles** folder outside the folder of this toolbox (no matter where it is placed), which must be the Abaqus working directory. After the Abaqus analysis terminates and the results file is created in the Abaqus working directory, it is processed appropriately by Matlab to obtain the requested results. Finally, the results are presented and checked with regard to their class and size. See here for a complete list of the functions verified and the verification results for each function. The verification source codes are contained in the folder named **Verification**.

The verification of this toolbox was made using Abaqus 6.13.

6. Supplementary files

Except for the source code files and folders used in this toolbox other supplementary files and folders are provided, which are the following:

6.1. A folder named **AbaqusInputFiles** which contains the input files which are run by Abaqus. These Abaqus files can be run by opening Abaqus/Command and typing < < abaqus job=X > > where X is the name of the Abaqus input file without the extension (*.inp). Each Abaqus input file is named with a number, let it be Y, which is the record key of the output variable identifier. The Abaqus input file Y.inp is run by Abaqus and produces results which are obtained after Abaqus completes the analysis by the function RecY.m. The Abaqus input files can be opened in any simple text editor, to view the various options specified in them.

6.2. A folder named **help** which contains all the source files which are published in the documentation, and do not include any verification examples. Such source files include the record key tables, function lists, etc.

6.3. A folder named **html** which contains all the html files of the documentation of this toolbox, including all the html files produced by publishing the verification examples of this toolbox. All the verification examples contained in the folder **Verification** and the editing files of the external functions and the Abaqus input files contained in the folder **help** are published by Matlab in this folder and are accessible through the documentation.

7. Demonstration of Abaqus2Matlab toolbox

Follow the instructions below to watch step by step an example verification procedure of the toolbox:

7.1. Ensure that Abaqus license server has started successfully.

7.2. Place the folder of the toolbox in the Abaqus working directory (usually C:\Temp)

7.3. Open the file named < < Documentation.m > > in Matlab and run it (press F5)

7.4. Type in the command window of Matlab the name of the file to be executed (it will be one of the verification files in the **Verification** folder) without its extension. The name of the file is of the form VerifyX, where X is the name of the Abaqus input file (X.inp) which is run by Abaqus to produce the corresponding results file X.fil in the Abaqus working directory. The information contained in X.fil is processed by the external Matlab function RecX.m, to give the requested output. For example by typing Verify8 in the command window of Matlab, the file 8.inp is run by Abaqus, after the analysis the file 8.fil is created in the Abaqus working directory, and the function Rec8.m obtains the requested results.

7.5. After the source code in the file VerifyX.m has run, the results of the Abaqus results file X.fil will appear in the command window. The results of the run can be viewed in the documentation which accompanies this toolbox. A complete list of the verification results for all Abaqus results postprocessing functions can be found here.

8. Instructions for use of Abaqus2Matlab toolbox

Follow the instructions below to run and use the toolbox:

8.1. Ensure that Abaqus license server has started successfully.

8.2. Place the folder of the toolbox in the Abaqus working directory (usually C:\Temp). Usually, this step is not necessary, since Abaqus can run from any directory. This action is suggested, however, to avoid confusion with the large number of files which are created in each Abaqus run.

8.3. Open the file named < < Documentation.m > > in Matlab and run it (press F5)

8.4. The source codes in the matlab verification files (VerifyX.m) can be followed to extract the results of an arbitrary Abaqus input file.

8.5. To extract an arbitrary Abaqus analysis result from an Abaqus results file, initially the record key and the output variable identifier have to be specified. These can be obtained from Analysis result types for an analysis-type output, Element result types for an element-type output, and from Node result types for a node-type output.

8.6. To view the instructions for use of each function, type < < doc RecX > > or < < help RecX > > (where X is the record key found in step 8.5 above) in the Matlab command window. the first option shows the function manual in a matlab browser, whereas the second option shows the function manual in the matlab command window. In the manual of each function the necessary options to be included in the Abaqus input file are shown.

8.7. Construct the relative Abaqus input file, and place it in the Abaqus working directory. It is supposed that until here, the Abaqus input file is ready to be run by Abaqus.

8.8. Run the Abaqus input file in Abaqus, either by opening Abaqus/Command and typing < < abaqus job=X > >, then enter, or by typing in the Matlab command window < < !abaqus job=X > >, then enter. After the analysis terminates, the results file X.fil is automatically generated. This file is then read by Matlab to extract the requested results.

8.9. Place the file X.fil in the same directory with function Fil2str. Type in the Matlab command window < < Rec= Fil2str ('X.fil') > >. The variable Rec is a one-row string containing the information contained in the X.fil file.

8.10. Type in the Matlab command window < < out=RecX(Rec) > >. The variable out contains the requested results, extracted from the X.fil results file. It will be generally a double or cell array. For more information about the identity and/or physical meaning of each element contained in this array, one can refer to the manual of the function RecX.m (mentioned in section 8.6 above) or section 5.1.2 (Results file output format) of the Abaqus Analysis User's Guide

---

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Table of records written for any file output request

Find the record key, the output variable identifier and the function associated with each of the possible analysis result types, sorted in alphabetical order. The following 8 different analysis result types can be used with Abaqus2Matlab toolbox.

RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Abaqus release, etc.	1921	-	Rec1921.m
Active degrees of freedom	1902	-	Rec1902.m
Element definitions	1900	-	Rec1900.m
Increment start record	2000	-	Rec2000.m
Label cross-reference	1940	-	Rec1940.m
Modal	1980	-	Rec1980.m
Node definitions	1901	-	Rec1901.m
Output request definition	1911	-	Rec1911.m

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

List of functions used for any file output request

Details about the external functions used for obtaining results about any file output request. See Analysis result types to view the possible analysis result types that can be obtained using Abaqus2Matlab toolbox.

Rec1900: Element definitions

Rec1901: Node definitions

Rec1902: Active degrees of freedom

Rec1911: Output request definition

Rec1921: Abaqus release, etc.

Rec1940: Label cross-reference

Rec1980: Modal

Rec2000: Increment start record

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Table of records written for any element file output request

Find the record key, the output variable identifier and the function associated with each of the possible element result types, sorted in alphabetical order. The following 46 different element result types can be used with Abaqus2Matlab toolbox.

ELEMENT RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Average Shell Section Stress	83	SSAVG	Rec83.m
Concrete Failure	31	CONF	Rec31.m
Coordinates	8	COORD	Rec8.m
Creep Strain (Including Swelling)	23	CE	Rec23.m
Element Status	61	STATUS	Rec61.m
Energy (Summed over Element)	19	ELEN	Rec19.m
Energy Density	14	ENER	Rec14.m
Equivalent plastic strain components	45	PEQC	Rec45.m
Film	33	FILM	Rec33.m
Gel (Pore Pressure Analysis)	40	GELVR	Rec40.m
Heat Flux Vector	28	HFL	Rec28.m
J-integral	1991	SP	Rec1991.m
Logarithmic Strain	89	LE	Rec89.m
Mass Concentration (Mass Diffusion Analysis)	38	CONC	Rec38.m
Mechanical Strain Rate	91	ER	Rec91.m
Nodal Flux Caused by Heat	10	NFLUX	Rec10.m
Nominal Strain	90	NE	Rec90.m
Plastic Strain	22	PE	Rec22.m
Pore Fluid Effective Velocity Vector	97	FLVEL	Rec97.m
Pore or Acoustic Pressure	18	POR	Rec18.m
Principal elastic strains	408	EEP	Rec408.m
Principal inelastic strains	409	IEP	Rec409.m
Principal logarithmic strains	405	LEP	Rec405.m
Principal mechanical strain rates	406	ERP	Rec406.m
Principal nominal strains	404	NEP	Rec404.m
Principal plastic strains	411	PEP	Rec411.m

Principal strains	403	EP	Rec403.m
Principal stresses	401	SP	Rec401.m
Principal thermal strains	410	THEP	Rec410.m
Principal values of backstress tensor for kinematic hardening plasticity	402	ALPHAP	Rec402.m
Principal values of deformation gradient	407	DGP	Rec407.m
Radiation	34	RAD	Rec34.m
Saturation (Pore Pressure Analysis)	35	SAT	Rec35.m
Section Force and Moment	13	SF	Rec13.m
Section Strain and Curvature	29	SE	Rec29.m
Section Thickness	27	STH	Rec27.m
Strain Jump at Nodes	32	SJP	Rec32.m
Stress	11	S	Rec11.m
Stress Invariant	12	SINV	Rec12.m
Thermal Strain	88	THE	Rec88.m
Total Elastic Strain	25	EE	Rec25.m
Total Fluid Volume Ratio	43	FLUVR	Rec43.m
Total Inelastic Strain	24	IE	Rec24.m
Total Strain	21	E	Rec21.m
Unit Normal to Crack in Concrete	26	CRACK	Rec26.m
Whole Element Volume	78	EVOL	Rec78.m

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

List of functions used for any element file output request

Details about the external functions used for obtaining results about any element file output request. See Element result types to view the possible element result types that can be obtained using Abaqus2Matlab toolbox.

Rec8: Coordinates

Rec10: Nodal Flux Caused By Heat

Rec11: Stress

Rec12: Stress Invariant

Rec13: Section Force And Moment

Rec14: Energy Density

Rec18: Pore Or Acoustic Pressure

Rec19: Energy (Summed over Element)

Rec21: Total Strain

Rec22: Plastic Strain

Rec23: Creep Strain (Including Swelling)

Rec24: Total Inelastic Strain

Rec25: Total Elastic Strain

Rec26: Unit Normal To Crack In Concrete

Rec27: Section Thickness

Rec28: Heat Flux Vector

Rec29: Section Strain And Curvature

Rec31: Concrete Failure

Rec32: Strain Jump At Nodes

Rec33: Film

Rec34: Radiation

Rec35: Saturation (Pore Pressure Analysis)

Rec38: Mass Concentration (Mass Diffusion Analysis)

Rec40: Gel (Pore Pressure Analysis)

Rec43: Total Fluid Volume Ratio

Rec45: Equivalent plastic strain components

Rec61: Element Status

Rec78: Whole Element Volume

Rec83: Average Shell Section Stress

Rec88: Thermal Strain

Rec89: Logarithmic Strain

Rec90: Nominal Strain

Rec91: Mechanical Strain Rate

Rec97: Pore Fluid Effective Velocity Vector

Rec401: Principal Stress

Rec402: Principal values of backstress tensor for kinematic hardening plasticity

Rec403: Principal strains

Rec404: Principal nominal strains

Rec405: Principal logarithmic strains

Rec406: Principal mechanical strain rates

Rec407: Principal values of deformation gradient

Rec408: Principal elastic strains

Rec409: Principal inelastic strains

Rec410: Principal thermal strains

Rec411: Principal plastic strains

Rec1991: J-integral

Table of records written for any node file output request

Find the record key, the output variable identifier and the function associated with each of the possible nodal result types, sorted in alphabetical order. The following 24 different nodal result types can be used with Abaqus2Matlab toolbox.

NODAL RECORD TYPE	RECORD KEY	OUTPUT VARIABLE IDENTIFIER	FUNCTION
Concentrated Electrical Nodal Charge	120	CECHG	Rec120.m
Concentrated Electrical Nodal Current	139	CECUR	Rec139.m
Concentrated Flux	206	CFL	Rec206.m
Electrical Potential	105	EPOT	Rec105.m
Electrical Reaction Charge	119	RCHG	Rec119.m
Electrical Reaction Current	138	RECUR	Rec138.m
Fluid Cavity Pressure	136	PCAV	Rec136.m
Fluid Cavity Volume	137	CVOL	Rec137.m
Internal Flux	214	RFLE	Rec214.m
Motions (in Cavity Radiation Analysis)	237	MOT	Rec237.m
Nodal Acceleration	103	A	Rec103.m
Nodal Coordinate	107	COORD	Rec107.m
Nodal Displacement	101	U	Rec101.m
Nodal Point Load	106	CF	Rec106.m
Nodal Reaction Force	104	RF	Rec104.m
Nodal Velocity	102	V	Rec102.m
Normalized Concentration (Mass Diffusion Analysis)	221	NNC	Rec221.m
Pore or Acoustic Pressure	108	POR	Rec108.m
Reactive Fluid Total Volume	110	RVT	Rec110.m
Reactive Fluid Volume Flux	109	RVF	Rec109.m
Residual Flux	204	RFL	Rec204.m
Temperature	201	NT	Rec201.m
Total Force	146	TF	Rec146.m
Viscous Forces Due to Static Stabilization	145	VF	Rec145.m

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

List of functions used for any node file output request

Details about the external functions used for obtaining results about any node file output request. See Node result types to view the possible node result types that can be obtained using Abaqus2Matlab toolbox.

Rec101: Nodal Displacement

Rec102: Nodal Velocity

Rec103: Nodal Acceleration

Rec104: Nodal Reaction Force

Rec105: Electrical Potential

Rec106: Nodal Point Load

Rec107: Nodal Coordinate

Rec108: Pore Or Acoustic Pressure

Rec109: Reactive Fluid Volume Flux

Rec110: Reactive Fluid Total Volume

Rec119: Electrical Reaction Charge

Rec120: Concentrated Electrical Nodal Charge

Rec136: Fluid Cavity Pressure

Rec137: Fluid Cavity Volume

Rec138: Electrical Reaction Current

Rec139: Concentrated Electrical Nodal Current

Rec145: Viscous Forces Due To Static Stabilization

Rec146: Total Force

Rec201: Temperature

Rec204: Residual Flux

Rec206: Concentrated Flux

Rec214: Internal Flux

Rec221: Normalized Concentration (Mass Diffusion Analysis)

Rec237: Motions (In Cavity Radiation Analysis)

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

List of verification results for the external functions

Verification examples published with Matlab for each external function used for obtaining Abaqus analysis results and included in Abaqus2Matlab toolbox.

Functions used to obtain element results

Rec8

Rec10, Rec11, Rec12, Rec13, Rec14, Rec18, Rec19

Rec21, Rec22, Rec23, Rec24, Rec25, Rec26, Rec27, Rec28, Rec29

Rec31, Rec32, Rec33, Rec34, Rec35, Rec38

Rec40, Rec43, Rec45

Rec61

Rec78

Rec83, Rec88, Rec89

Rec90, Rec91, Rec97

Rec401, Rec402, Rec403, Rec404, Rec405, Rec406, Rec407, Rec408, Rec409

Rec410, Rec411

Rec1991

Functions used to obtain node results

Rec101, Rec102, Rec103, Rec104, Rec105, Rec106, Rec107, Rec108, Rec109

Rec110, Rec119

Rec120

Rec136, Rec137, Rec138, Rec139

Rec145, Rec146

Rec201, Rec204, Rec206

Rec214

Rec221

Rec237

Functions used to obtain analysis results

Rec1900, Rec1901, Rec1902

Rec1911

Rec1921

Rec1940

Rec1980

Rec2000

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

List of Abaqus input files used for the verification of the external functions

Abaqus input files published with Matlab (one for each external function) used for obtaining Abaqus analysis results and included in Abaqus2Matlab toolbox.

Functions used to obtain element results

8.inp

10.inp, 11.inp, 12.inp, 13.inp, 14.inp, 18.inp, 19.inp

21.inp, 22.inp, 23.inp, 24.inp, 25.inp, 26.inp, 27.inp, 28.inp, 29.inp

31.inp, 32.inp, 33.inp, 34.inp, 35.inp, 38.inp

40.inp, 43.inp, 45.inp

61.inp

78.inp

83.inp, 88.inp, 89.inp

90.inp, 91.inp, 97.inp

401.inp, 402.inp, 403.inp, 404.inp, 405.inp, 406.inp, 407.inp, 408.inp, 409.inp

410.inp, 411.inp

1991.inp

Functions used to obtain node results

101.inp, 102.inp, 103.inp, 104.inp, 105.inp, 106.inp, 107.inp, 108.inp, 109.inp

110.inp, 119.inp

120.inp

136.inp, 137.inp, 138.inp, 139.inp

145.inp, 146.inp

201.inp, 204.inp, 206.inp

214.inp

221.inp

237.inp

Functions used to obtain analysis results

1900.inp, 1901.inp, 1902.inp

1911.inp

1921.inp

1940.inp

1980.inp

2000.inp

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

```
help Fil2str
```

```
Assembly of the information in the ABAQUS results file

Syntax
    #Rec# = Fil2str(#ResultsFileName#);

Description
    Assemble the information contained in an ABAQUS results (*.fil) file
    in ASCII format into a string that has one row.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #ResultsFileName# (string) is a string containing the name of the
        ABAQUS results (*.fil) file, along with its extension. The
        results file is generated by Abaqus after the analysis has been
        completed.

Output parameters
    #Rec# ([1 x #m#]) is a string containing the information of the
        Abaqus results file assembled in one row.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec8
```

```
ABAQUS coordinate output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec8(#Rec#)

Description
    Read coordinate output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for coordinate output is
    8 (only in ABAQUS/Standard). See section < < Results file output format > >
    in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain coordinate results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        COORD
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 8 as follows:
        Column  1  –  First coordinate.
        Column  2  –  Etc.
        where #n# is the number of elements multiplied by the number of
        increments, and #m# is the number of coordinates of each element.
        If the results file does not contain the desired output, #out#
        will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec10
```

```
ABAQUS nodal flux caused by heat output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec10(#Rec#)

Description
    Read nodal flux (caused by heat) output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for nodal
    flux output is 10 (only in ABAQUS/Standard). See section < < Results
    file output format > > in ABAQUS Analysis User's manual for more
    details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain nodal flux results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        NFLUX
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 10 as follows:
        Column  1  –  Node number.
        Column  2  –  First flux component.
        Column  3  –  Etc.
        where #n# is the number of nodes and #m# is the number of nodal
        flux components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec11
```

```
ABAQUS stress output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec11(#Rec#)

Description
    Read stress output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for stress output is 11. See
    section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain stress results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        S
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 11 as follows:
        Column  1  –  First stress component.
        Column  2  –  Second stress component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of stress components
        of each element. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec12
```

```
ABAQUS stress invariant output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec12(#Rec#)

Description
    Read stress invariant output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for stress invariant
    output is 12 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain stress invariant results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SINV
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 7]) is a double array containing the attributes of
        the record key 12 as follows:
        Column  1  –  Mises stress.
        Column  2  –  Tresca stress.
        Column  3  –  Hydrostatic pressure.
        Column  4  –  Currently not used.
        Column  5  –  Currently not used.
        Column  6  –  Currently not used.
        Column  7  –  Third stress invariant.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments. If the results file does not contain the
        desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec13
```

```
ABAQUS section force and moment output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec13(#Rec#)

Description
    Read section force and moment output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    section force and moment output is 13. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain section force and moment results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 13 as follows:
        Column  1  –  First section force.
        Column  2  –  Second section force.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a description of which section forces are available
        for each beam or shell element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments, and
        #m# is the number of section force components of each element. If
        the results file does not contain the desired output, #out# will
        be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec14
```

```
ABAQUS energy density output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec14(#Rec#)

Description
    Read energy density output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for energy density
    output is 14. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain energy density results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        ENER
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 7]) is a double array containing the attributes of
        the record key 14 as follows:
        Column  1  –  Strain energy.
        Column  2  –  Plastic dissipation.
        Column  3  –  Creep dissipation.
        Column  4  –  Viscous dissipation.
        Column  5  –  Electrostatic energy.
        Column  6  –  Energy dissipated due to electrical conduction.
        Column  7  –  Damage dissipation.
        where #n# is the number of elements multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec18
```

```
ABAQUS pore or acoustic pressure output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec18(#Rec#)

Description
    Read pore or acoustic pressure output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for pore
    or acoustic pressure output is 18 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain pore or acoustic pressure results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        POR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 18 as follows:
        Column  1  –  Liquid pressure.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec19
```

```
ABAQUS energy (summed over element) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec19(#Rec#)

Description
    Read energy (summed over element) output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    energy (summed over element) output is 19. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain energy (summed over element) results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        ELEN
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 10]) is a double array containing the attributes of
        the record key 19 as follows:
        Column  1  –  Kinetic energy.
        Column  2  –  Strain energy.
        Column  3  –  Plastic dissipation.
        Column  4  –  Creep dissipation.
        Column  5  –  Viscous dissipation, not including dissipation due
        to stabilization.
        Column  6  –  Static dissipation (due to stabilization).
        Column  7  –  Artificial strain energy.
        Column  8  –  Electrostatic energy.
        Column  9  –  Electrical energy dissipated in a conductor.
        Column  10  –  Damage dissipation.
        where #n# is the number of elements multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec21
```

```
ABAQUS total strain output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec21(#Rec#)

Description
    Read total strain output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for total strain output
    is 21. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain total strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        E
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 21 as follows:
        Column  1  –  First strain component.
        Column  2  –  Second strain component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a description of the components for a given element
        type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of strain components
        of each element. If the results file does not contain the desired
        output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec22
```

```
ABAQUS plastic strain output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec22(#Rec#)

Description
    Read plastic strain output from the results (*.fil) file generated
    from the ABAQUS finite element software (ABAQUS/Explicit). The
    asterisk (*) is replaced by the name of the results file. The record
    key for plastic strain output is 22. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain plastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        PE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 6]) is a double array containing the attributes of
        the record key 22 as follows:
        Column  1  –  First plastic strain component.
        Column  2  –  Second plastic strain component.
        Column  3  –  Etc; followed by the equivalent plastic strain,
        actively yielding flag (yes or no, A8 format), and magnitude of
        plastic strain in Abaqus/Standard; followed by “0.0, UNUSED, 0.0”
        in Abaqus/Explicit for consistency with the length of the
        Abaqus/Standard record.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments. If the results file does not contain the
        desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec23
```

```
ABAQUS creep strain (including swelling) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec23(#Rec#)

Description
    Read creep strain output (including swelling) from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for creep strain output is 23 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain creep strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        CE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 23 as follows:
        Column  1  –  First creep strain component.
        Column  2  –  Second creep strain component.
        Column  3  –  Etc; followed by the equivalent creep strain,
        volumetric swelling strain, and magnitude of creep strain.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of creep strain
        results of each element. If the results file does not contain the
        desired output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec24
```

```
ABAQUS total inelastic strain output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec24(#Rec#)

Description
    Read total inelastic strain output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for total
    inelastic strain output is 24 (only in ABAQUS/Standard). See section
    < < Results file output format > > in ABAQUS Analysis User's manual for
    more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain total inelastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        IE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 24 as follows:
        Column  1  –  First inelastic strain component.
        Column  2  –  Second inelastic strain component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of total inelastic
        strain components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec25
```

```
ABAQUS total elastic strain output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec25(#Rec#)

Description
    Read total elastic strain output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for total
    elastic strain output is 25 (only in ABAQUS/Standard). See section
    < < Results file output format > > in ABAQUS Analysis User's manual for
    more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain total elastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        EE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 25 as follows:
        Column  1  –  First elastic strain component.
        Column  2  –  Second elastic strain component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of total elastic
        strain components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec26
```

```
ABAQUS unit normal to crack in concrete output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec26(#Rec#)

Description
    Read unit normal to crack in concrete output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    unit normal to crack in concrete output is 26. See section < < Results
    file output format > > in ABAQUS Analysis User's manual for more
    details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain unit normal to crack in concrete results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        CRACK
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 9]) is a double array containing the attributes of
        the record key 26 as follows:
        Column  1  –  11-component (if a 1D, 2D, or 3D analysis).
        Column  2  –  12-component (if a 2D or 3D analysis).
        Column  3  –  13-component (if a 3D analysis).
        Column  4  –  21-component (if a 2D or 3D analysis).
        Column  5  –  22-component (if a 2D or 3D analysis).
        Column  6  –  23-component (if a 3D analysis).
        Column  7  –  31-component (if a 3D analysis).
        Column  8  –  32-component (if a 3D analysis).
        Column  9  –  33-component (if a 3D analysis).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec27
```

```
  ABAQUS section thickness output to MATLAB

  Syntax
      #Rec# = Fil2str('*.fil');
      #out# = Rec27(#Rec#)

  Description
      Read section thickness output from the results (*.fil) file generated
      from the ABAQUS finite element software. The asterisk (*) is replaced
      by the name of the results file. The record key for section thickness
      output is 27. See section < < Results file output format > > in ABAQUS
      Analysis User's manual for more details.
      The following options with parameters have to be specified in the
      ABAQUS input file for the results (*.fil) file to be created and to
      contain section thickness results:
          ...
          *FILE FORMAT, ASCII
          *EL FILE
          STH
          ...
      NOTE: The results file (*.fil) must be placed in the same directory
      with the MATLAB source files in order to be processed.

  Input parameters
      #Rec# (string) is an one-row string containing the ASCII code of the
          ABAQUS results (*.fil) file. It is generated by the function
          Fil2str.m.

  Output parameters
      #out# ([#n# x 1]) is a double array containing the attributes of
          the record key 27 as follows:
          Column  1  –  Current section thickness for membranes and
          finite-strain shells in Abaqus/Standard and for plane stress
          elements, membranes, and all shells in Abaqus/Explicit.
          where #n# is the number of elements multiplied by the number of
          integration points multiplied by the number of increments. If the
          results file does not contain the desired output, #out# will be
          an empty array

  _______________________________________________________________________
  Abaqus2Matlab - www.abaqus2matlab.com
  Copyright (c) 2016 by George Papazafeiropoulos

  If using this toolbox for research or industrial purposes, please cite:
  G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
  Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec28
```

```
ABAQUS heat flux vector output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec28(#Rec#)

Description
    Read heat flux vector output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for heat flux vector
    output is 28. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain heat flux vector results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        HFL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 28 as follows:
        Column  1  –  Magnitude.
        Column  2  –  First component.
        Column  3  –  Second component.
        Column  4  –  Etc.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments, and
        #m# is the number of heat flux vector components of each element.
        If the results file does not contain the desired output, #out#
        will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec29
```

```
ABAQUS section strain and curvature output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec29(#Rec#)

Description
    Read section strain and curvature output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    section strain and curvature output is 29. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain section strain and curvature results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 29 as follows:
        Column  1  –  First section strain.
        Column  2  –  Second section strain.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments, and
        #m# is the number of section strain and curvature components of
        each element. If the results file does not contain the desired
        output, #out# will be an empty array
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec31
```

```
ABAQUS concrete failure output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec31(#Rec#)

Description
    Read concrete failure output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for concrete failure
    output is 31 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain concrete failure results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        CONF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a cell array containing the attributes of
        the record key 31 as follows:
        Column  1  -  Serial number (increasing from 1 to #n#)
        Column  2  –  Summary of the state of a concrete material point.
        This is the number of cracks or –1 if the concrete has crushed.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec32
```

```
ABAQUS strain jump at nodes output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec32(#Rec#)

Description
    Read strain jump at nodes output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    strain jump at nodes output is 32 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain strain jump at nodes results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SJP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 32 as follows:
        Column  1  –  First strain jump component.
        Column  2  –  Second strain jump component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of strain jump at
        nodes components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec33
```

```
ABAQUS film output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec33(#Rec#)

Description
    Read film output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for film output is 33 (only
    in ABAQUS/Standard). See section < < Results file output format > > in
    ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain film results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        FILM
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 3]) is a cell array containing the attributes of
        the record key 33 as follows:
        Column  1  –  Type.
        Column  2  –  Sink temperature.
        Column  3  –  Film coefficient.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec34
```

```
ABAQUS radiation output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec34(#Rec#)

Description
    Read radiation output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for radiation output is
    34 (only in ABAQUS/Standard). See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain radiation results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        RAD
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 3]) is a cell array containing the attributes of
        the record key 34 as follows:
        Column  1  –  Type.
        Column  2  –  Sink temperature.
        Column  3  –  Radiation constant.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec35
```

```
ABAQUS saturation (pore pressure analysis) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec35(#Rec#)

Description
    Read saturation (pore pressure analysis) output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for saturation output is 35 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain saturation results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SAT
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 35 as follows:
        Column  1  –  Saturation.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec38
```

```
ABAQUS mass concentration (mass diffusion analysis) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec38(#Rec#)

Description
    Read mass concentration (mass diffusion analysis) output from the
    results (*.fil) file generated from the ABAQUS finite element
    software. The asterisk (*) is replaced by the name of the results
    file. The record key for mass concentration output is 38 (only in
    ABAQUS/Standard). See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain mass concentration results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        CONC
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 38 as follows:
        Column  1  –  Concentration.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec40
```

```
ABAQUS gel (pore pressure analysis) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec40(#Rec#)

Description
    Read gel (pore pressure analysis) output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    gel output is 40 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain gel results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        GELVR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 40 as follows:
        Column  1  –  Gel volume ratio.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec43
```

```
ABAQUS total fluid volume ratio output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec43(#Rec#)

Description
    Read total fluid volume ratio output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for total
    fluid volume ratio output is 43 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain total fluid volume ratio results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        FLUVR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 43 as follows:
        Column  1  –  Total fluid volume ratio.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments. If the
        results file does not contain the desired output, #out# will be
        an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec45
```

```
ABAQUS equivalent plastic strain components output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec45(#Rec#)

Description
    Read equivalent plastic strain components output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for equivalent plastic strain components output is 45. See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain equivalent plastic strain component results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        PEQC
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 8]) is a double array containing the attributes of
        the record key 45 as follows:
        Column  1  –  Equivalent plastic strain for Drucker-Prager
        failure surface.
        Column  2  –  Actively yielding flag (yes or no, A8 format) for
        Drucker-Prager failure surface.
        Column  3  –  Equivalent plastic strain for cap surface.
        Column  4  –  Actively yielding flag (yes or no, A8 format) for
        cap surface.
        Column  5  –  Equivalent plastic strain for transition surface.
        Column  6  –  Actively yielding flag (yes or no, A8 format) for
        transition surface.
        Column  7  –  Total volumetric inelastic strain.
        Column  8  –  Actively yielding flag (yes or no, A8 format).
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments. If the results file does
        not contain the desired output, #out# will be an empty array.
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec61
```

```
ABAQUS element status output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec61(#Rec#)

Description
    Read element status output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for element status
    output is 61 (only in ABAQUS/Explicit). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain element status results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        STATUS
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 1]) is a double array containing the attributes of
        the record key 61 as follows:
        Column  1  –  Status of element (shear failure model, tensile
        failure model, porous failure criterion, brittle failure model,
        Johnson-Cook plasticity model, and VUMAT). The status of an
        element is 1.0 if the element is active, 0.0 if the element is
        not.
        where #n# is the number of elements multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec78
```

```
ABAQUS whole element volume output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec78(#Rec#)

Description
    Read whole element volume output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for whole
    element volume output is 78. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain whole element volume results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        EVOL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 78 as follows:
        Column  1  -  Current element volume.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #m# is the number of
        stress components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016
    Emilio Martinez-Paneda

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec83
```

```
ABAQUS average shell section stress output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec83(#Rec#)

Description
    Read average shell section stress output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    average shell section stress output is 83 (only in ABAQUS/Standard).
    See section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain average shell section stress results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SSAVG
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 83 as follows:
        Column  1  –  First section stress.
        Column  2  –  Second section stress.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a description of which section stresses are available
        for each shell element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of increments, and
        #m# is the number of average shell section stress components of
        each element. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec88
```

```
ABAQUS thermal strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec88(#Rec#)

Description
    Read thermal strain output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for thermal strain
    output is 88 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain thermal strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        THE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 88 as follows:
        Column  1  –  First thermal strain component.
        Column  2  –  Second thermal strain component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of thermal strain
        components of each element. If the results file does not contain
        the desired output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec89
```

```
  ABAQUS logarithmic strain output to MATLAB

  Syntax
      #Rec# = Fil2str('*.fil');
      #out# = Rec89(#Rec#)

  Description
      Read logarithmic strain output from the results (*.fil) file
      generated from the ABAQUS finite element software. The asterisk (*)
      is replaced by the name of the results file. The record key for
      logarithmic strain output is 89. See section < < Results file output
      format > > in ABAQUS Analysis User's manual for more details.
      The following options with parameters have to be specified in the
      ABAQUS input file for the results (*.fil) file to be created and to
      contain logarithmic strain results:
          ...
          *FILE FORMAT, ASCII
          *EL FILE
          LE
          ...
      NOTE: The results file (*.fil) must be placed in the same directory
      with the MATLAB source files in order to be processed.

  Input parameters
      #Rec# (string) is an one-row string containing the ASCII code of the
          ABAQUS results (*.fil) file. It is generated by the function
          Fil2str.m.

  Output parameters
      #out# ([#n# x #m#]) is a double array containing the attributes of
          the record key 89 as follows:
          Column  1  –  First logarithmic strain component.
          Column  2  –  Second logarithmic strain component.
          Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
          Manual for a definition of the number and type of the components
          for the element type).
          where #n# is the number of elements multiplied by the number of
          integration points multiplied by the number of section points
          (for shell, beam, or layered solid elements) multiplied by the
          number of increments, and #m# is the number of logarithmic strain
          components of each element. If the results file does not contain
          the desired output, #out# will be an empty array

  _______________________________________________________________________
  Abaqus2Matlab - www.abaqus2matlab.com
  Copyright (c) 2016 by George Papazafeiropoulos

  If using this toolbox for research or industrial purposes, please cite:
  G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
  Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec90
```

```
ABAQUS nominal strain output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec90(#Rec#)

Description
    Read nominal strain output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for nominal strain
    output is 90. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain nominal strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        NE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 90 as follows:
        Column  1  –  First nominal strain component.
        Column  2  –  Second nominal strain component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of nominal strain
        components of each element. If the results file does not contain
        the desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec91
```

```
ABAQUS mechanical strain rate output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec91(#Rec#)

Description
    Read mechanical strain rate output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    mechanical strain rate output is 91 (only in ABAQUS/Standard). See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain mechanical strain rate results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        ER
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 91 as follows:
        Column  1  –  First strain rate component.
        Column  2  –  Second strain rate component.
        Column  3  –  Etc. (See “Elements” in Abaqus Analysis User's
        Manual for a definition of the number and type of the components
        for the element type).
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of mechanical strain
        rate components of each element. If the results file does not
        contain the desired output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec97
```

```
ABAQUS pore fluid effective velocity vector output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec97(#Rec#)

Description
    Read pore fluid effective velocity vector output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for pore fluid effective velocity vector output is 97 (only in
    ABAQUS/Standard). See section < < Results file output format > > in
    ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain pore fluid effective velocity vector results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        FLVEL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 97 as follows:
        Column  1  –  Magnitude.
        Column  2  –  First component.
        Column  3  –  Second component.
        Column  4  –  Etc.
        where #n# is the number of elements multiplied by the number of
        integration points multiplied by the number of section points
        (for shell, beam, or layered solid elements) multiplied by the
        number of increments, and #m# is the number of pore fluid
        effective velocity vector components of each element. If the
        results file does not contain the desired output, #out# will be
        an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
```



```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec101
```

```
ABAQUS displacement output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec101(#Rec#)

Description
    Read displacement output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for displacement output
    is 101. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain displacement results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        U
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 101 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of displacement.
        Column  3  –  Second component of displacement.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of displacements per node. If
        the results file does not contain the desired output, #out# will
        be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec102
```

```
ABAQUS velocity output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec102(#Rec#)

Description
    Read velocity output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for velocity output is 102.
    See section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain velocity results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        V
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 102 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of velocity.
        Column  3  –  Second component of velocity.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of velocities per node. If the
        results file does not contain the desired output, #out# will be
        an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec103
```

```
ABAQUS acceleration output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec103(#Rec#)

Description
    Read acceleration output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for acceleration output
    is 103. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain acceleration results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        A
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 103 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of acceleration.
        Column  3  –  Second component of acceleration.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of accelerations per node. If
        the results file does not contain the desired output, #out# will
        be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec104
```

```
ABAQUS reaction force output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec104(#Rec#)

Description
    Read reaction force output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for reaction force
    output is 104. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain reaction force results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 104 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of reaction force.
        Column  3  –  Second component of reaction force.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of reaction forces per node.
        If the results file does not contain the desired output, #out#
        will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec105
```

```
ABAQUS electrical potential output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec105(#Rec#)

Description
    Read electrical potential output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    electrical potential output is 105. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain electrical potential results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        EPOT
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 105 as follows:
        Column  1  –  Node number.
        Column  2  –  Magnitude.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec106
```

```
ABAQUS point loads, moments, fluxes output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec106(#Rec#)

Description
    Read point loads, moments, fluxes output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    point loads, moments, fluxes output is 106. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain point loads, moments, fluxes results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        CF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 106 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of load or flux.
        Column  3  –  Second component of load or flux.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of loads or fluxes per node.
        If the results file does not contain the desired output, #out#
        will be an empty array
```

```
help Rec107
```

```
ABAQUS coordinate output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec107(#Rec#)

Description
    Read coordinate output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for coordinate output is
    107. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain coordinate results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        COORD
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 107 as follows:
        Column  1  –  Node number.
        Column  2  –  First coordinate.
        Column  3  –  Second coordinate.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of coordinates per node. If
        the results file does not contain the desired output, #out# will
        be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec108
```

```
ABAQUS pore or acoustic pressure output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec108(#Rec#)

Description
    Read pore or acoustic pressure output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for pore
    or acoustic pressure output is 108. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain pore or acoustic pressure results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        POR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 108 as follows:
        Column  1  –  Node number.
        Column  2  –  Pressure.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec109
```

```
ABAQUS reactive fluid volume flux output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec109(#Rec#)

Description
    Read reactive fluid volume flux output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    reactive fluid volume flux output is 109 (only in ABAQUS/Standard).
    See section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain reactive fluid volume flux results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RVF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 109 as follows:
        Column  1  –  Node number.
        Column  2  –  Reaction fluid volume flux.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec110
```

```
  ABAQUS reactive fluid total volume output to MATLAB

  Syntax
      #Rec# = Fil2str('*.fil');
      #out# = Rec110(#Rec#)

  Description
      Read reactive fluid total volume output from the results (*.fil) file
      generated from the ABAQUS finite element software. The asterisk (*)
      is replaced by the name of the results file. The record key for
      reactive fluid total volume output is 110 (only in ABAQUS/Standard).
      See section < < Results file output format > > in ABAQUS Analysis User's
      manual for more details.
      The following options with parameters have to be specified in the
      ABAQUS input file for the results (*.fil) file to be created and to
      contain reactive fluid total volume results:
          ...
          *FILE FORMAT, ASCII
          *NODE FILE
          RVT
          ...
      NOTE: The results file (*.fil) must be placed in the same directory
      with the MATLAB source files in order to be processed.

  Input parameters
      #Rec# (string) is an one-row string containing the ASCII code of the
          ABAQUS results (*.fil) file. It is generated by the function
          Fil2str.m.

  Output parameters
      #out# ([#n# x 2]) is a double array containing the attributes of
          the record key 110 as follows:
          Column  1  –  Node number.
          Column  2  –  Reaction fluid total volume.
          where #n# is the number of nodes multiplied by the number of
          increments. If the results file does not contain the desired
          output, #out# will be an empty array

  _______________________________________________________________________
  Abaqus2Matlab - www.abaqus2matlab.com
  Copyright (c) 2016 by George Papazafeiropoulos

  If using this toolbox for research or industrial purposes, please cite:
  G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
  Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec119
```

```
ABAQUS electrical reaction charge output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec119(#Rec#)

Description
    Read electrical reaction charge output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    electrical reaction charge output is 119 (only in ABAQUS/Standard).
    See section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain electrical reaction charge results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RCHG
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 119 as follows:
        Column  1  –  Node number.
        Column  2  –  Charge scalar value.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec120
```

```
ABAQUS concentrated electrical nodal charge output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec120(#Rec#)

Description
    Read concentrated electrical nodal charge output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for concentrated electrical nodal charge output is 120 (only in
    ABAQUS/Standard). See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain concentrated electrical nodal charge results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        CECHG
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 120 as follows:
        Column  1  –  Node number.
        Column  2  –  Current scalar value.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec136
```

```
ABAQUS fluid cavity pressure output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec136(#Rec#)

Description
    Read fluid cavity pressure output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for fluid
    cavity pressure output is 136. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain fluid cavity pressure results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        PCAV
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 136 as follows:
        Column  1  –  Fluid cavity reference node number.
        Column  2  –  Pressure.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec137
```

```
ABAQUS fluid cavity volume output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec137(#Rec#)

Description
    Read fluid cavity volume output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for fluid
    cavity volume output is 137. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain fluid cavity volume results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        CVOL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 137 as follows:
        Column  1  –  Fluid cavity reference node number.
        Column  2  –  Volume.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec138
```

```
ABAQUS electrical reaction current output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec138(#Rec#)

Description
    Read electrical reaction current output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    electrical reaction current output is 138. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain electrical reaction current results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RECUR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 138 as follows:
        Column  1  –  Node number.
        Column  2  –  Electrical current.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec139
```

```
ABAQUS concentrated electrical nodal current output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec139(#Rec#)

Description
    Read concentrated electrical nodal current output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for concentrated electrical nodal current output is 139. See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain concentrated electrical nodal current results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        CECUR
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 139 as follows:
        Column  1  –  Node number.
        Column  2  –  Electrical current.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec145
```

```
ABAQUS viscous forces due to static stabilization output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec145(#Rec#)

Description
    Read viscous forces due to static stabilization output from the
    results (*.fil) file generated from the ABAQUS finite element
    software. The asterisk (*) is replaced by the name of the results
    file. The record key for viscous forces due to static stabilization
    output is 145 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain viscous forces due to static stabilization results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        VF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 145 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of viscous force.
        Column  3  –  Second component of viscous force.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of components of viscous force
        per node. If the results file does not contain the desired
        output, #out# will be an empty array
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec146
```

```
ABAQUS total force output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec146(#Rec#)

Description
    Read total force output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for total force output
    is 146 (only in ABAQUS/Standard). See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain total force results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        TF
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 146 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of total force.
        Column  3  –  Second component of total force.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of components of total forces
        per node. If the results file does not contain the desired
        output, #out# will be an empty array
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec201
```

```
ABAQUS temperature output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec201(#Rec#)

Description
    Read temperature output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for temperature output
    is 201. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain temperature results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        NT
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 201 as follows:
        Column  1  –  Node number.
        Column  2  –  Temperature.
        Column  3  –  Etc (for heat shells)
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of temperatures per node. If
        the results file does not contain the desired output, #out# will
        be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec204
```

```
ABAQUS residual flux output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec204(#Rec#)

Description
    Read residual flux output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for residual flux
    output is 204. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain residual flux results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RFL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 204 as follows:
        Column  1  –  Node number.
        Column  2  –  Residual flux.
        Column  3  –  Etc (for heat shells).
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of residual fluxes per node.
        If the results file does not contain the desired output, #out#
        will be an empty array

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec206
```

```
ABAQUS concentrated flux output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec206(#Rec#)

Description
    Read concentrated flux output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for concentrated flux
    output is 206 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain concentrated flux results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        CFL
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 206 as follows:
        Column  1  –  Node number.
        Column  2  –  Concentrated flux.
        Column  3  –  Etc (for heat shells)
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of concentrated fluxes per
        node. If the results file does not contain the desired output,
        #out# will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec214
```

```
ABAQUS internal flux output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec214(#Rec#)

Description
    Read internal flux output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for internal flux
    output is 214 (only in ABAQUS/Standard). See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain internal flux results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        RFLE
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 214 as follows:
        Column  1  –  Node number.
        Column  2  –  Flux, excluding external flux.
        Column  3  –  Etc (for heat shells)
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of internal fluxes per node.
        If the results file does not contain the desired output, #out#
        will be an empty array

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec221
```

```
ABAQUS normalized concentration (mass diffusion analysis) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec221(#Rec#)

Description
    Read normalized concentration output from the results (*.fil) file
    generated from the ABAQUS finite element software (mass diffusion
    analysis). The asterisk (*) is replaced by the name of the results
    file. The record key for normalized concentration output is 221 (only
    in ABAQUS/Standard). See section < < Results file output format > > in
    ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain normalized concentration results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        NNC
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a double array containing the attributes of
        the record key 221 as follows:
        Column  1  –  Node number.
        Column  2  –  Concentration.
        where #n# is the number of nodes multiplied by the number of
        increments. If the results file does not contain the desired
        output, #out# will be an empty array
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec237
```

```
ABAQUS motions (in cavity radiation analysis) output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec237(#Rec#)

Description
    Read motion output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for motion output is 237
    (only in ABAQUS/Standard). See section < < Results file output format > >
    in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain motion results:
        ...
        *FILE FORMAT, ASCII
        *NODE FILE
        MOT
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 237 as follows:
        Column  1  –  Node number.
        Column  2  –  First component of motion.
        Column  3  –  Second component of motion.
        Column  4  –  Etc
        where #n# is the number of nodes multiplied by the number of
        increments and #m#-1 is the number of components of motion per
        node. If the results file does not contain the desired output,
        #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec401
```

```
ABAQUS principal stresses output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec401(#Rec#)

Description
    Read principal stresses output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal stresses output is 401. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal stress results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        SP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 401 as follows:
        Column  1  -  Minimum principal stress.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016
    Emilio Martinez-Paneda

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec402
```

```
ABAQUS principal values of backstress tensor for kinematic hardening
plasticity output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec402(#Rec#)

Description
    Read principal values of backstress tensor for kinematic hardening
    plasticity output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for principal values of
    backstress tensor for kinematic hardening plasticity output is 402.
    See section < < Results file output format > > in ABAQUS Analysis User's
    manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal values of backstress tensor for kinematic hardening
    plasticity results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        ALPHAP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 402 as follows:
        Column  1  -  Minimum principal value of backstress tensor for
        kinematic hardening plasticity.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec403
```

```
ABAQUS principal strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec403(#Rec#)

Description
    Read principal strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal strains output is 403. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        EP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 403 as follows:
        Column  1  -  Minimum principal strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec404
```

```
ABAQUS principal nominal strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec404(#Rec#)

Description
    Read principal nominal strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal nominal strains output is 404. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal nominal strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        NEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 404 as follows:
        Column  1  -  Minimum principal nominal strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec405
```

```
ABAQUS principal logarithmic strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec405(#Rec#)

Description
    Read principal logarithmic strains output from the results (*.fil)
    file generated from the ABAQUS finite element software. The asterisk
    (*) is replaced by the name of the results file. The record key for
    principal logarithmic strains output is 405. See section < < Results
    file output format > > in ABAQUS Analysis User's manual for more
    details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal logarithmic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        LEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 405 as follows:
        Column  1  -  Minimum principal logarithmic strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

Published with MATLAB® R2015a

```
help Rec406
```

```
ABAQUS principal mechanical strain rates output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec406(#Rec#)

Description
    Read principal mechanical strain rates output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for principal mechanical strain rates output is 406. See section
    < < Results file output format > > in ABAQUS Analysis User's manual for
    more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal mechanical strain rate results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        ERP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 406 as follows:
        Column  1  -  Minimum principal mechanical strain rate.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec407
```

```
ABAQUS principal values of deformation gradient output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec407(#Rec#)

Description
    Read principal values of deformation gradient output from the results
    (*.fil) file generated from the ABAQUS finite element software. The
    asterisk (*) is replaced by the name of the results file. The record
    key for principal values of deformation gradient output is 407. See
    section < < Results file output format > > in ABAQUS Analysis User's manual
    for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal values of deformation gradient results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        DGP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 407 as follows:
        Column  1  -  Minimum principal value of deformation gradient.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec408
```

```
ABAQUS principal elastic strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec408(#Rec#)

Description
    Read principal elastic strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal elastic strains output is 408. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal elastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        EEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 408 as follows:
        Column  1  -  Minimum principal elastic strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

Published with MATLAB® R2015a

```
help Rec409
```

```
ABAQUS principal inelastic strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec409(#Rec#)

Description
    Read principal inelastic strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal inelastic strains output is 409. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal inelastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        IEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 409 as follows:
        Column  1  -  Minimum principal inelastic strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec410
```

```
ABAQUS principal thermal strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec410(#Rec#)

Description
    Read principal thermal strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal thermal strains output is 410. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal thermal strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        THEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 410 as follows:
        Column  1  -  Minimum principal thermal strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec411
```

```
ABAQUS principal plastic strains output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec411(#Rec#)

Description
    Read principal plastic strains output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    principal plastic strains output is 411. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain principal plastic strain results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        PEP
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #NDI#]) is a double array containing the attributes of
        the record key 411 as follows:
        Column  1  -  Minimum principal plastic strain.
        Column  2  -  Etc.
        (See < < Elements > > in Abaqus Analysis User's Manual for a
        definition of the number and type of the components for the
        element type), where #n# is the number of elements multiplied by
        the number of integration points multiplied by the number of
        section points (for shell, beam, or layered solid elements)
        multiplied by the number of increments, and #NDI# is the number
        of direct stresses at each point. If the results file does not
        contain the desired output, #out# will be an empty array.

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1900
```

```
ABAQUS element definitions output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1900(#Rec#)

Description
    Read element definitions output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    element definitions output is 1900. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a cell array containing the attributes of
        the record key 1900 as follows:
        Column  1  –  Element number.
        Column  2  –  Element type (characters, A8 format, left
        justified).
        Column  3  –  First node on the element.
        Column  4  –  Second node on the element.
        Column  5  –  Etc.
        where #n# is the number of elements and #m#-2 is the number of
        nodes per element.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1901
```

```
ABAQUS node definition output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1901(#Rec#)

Description
    Read node definition output from the results (*.fil) file generated
    from the ABAQUS finite element software. The asterisk (*) is replaced
    by the name of the results file. The record key for node definition
    output is 1901. See section < < Results file output format > > in ABAQUS
    Analysis User's manual for more details.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a double array containing the attributes of
        the record key 1901 as follows:
        Column  1  –  Node number.
        Column  2  –  First coordinate.
        Column  3  –  Second coordinate.
        Column  4  –  etc.
        where #n# is the number of nodes and #m#-1 is the number of
        coordinates per node. If the results file does not contain the
        desired output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1902
```

```
ABAQUS active degrees of freedom output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1902(#Rec#)

Description
    Read active degrees of freedom output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    active degrees of freedom output is 1902. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details. The
    following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([1 x 30]) is an array containing the attributes of the record
        key 1902 as follows:
        Column  1  -  Location in nodal arrays of degree of freedom 1 (0
        if DOF 1 is not active in the model).
        Column  2  -  Location in nodal arrays of degree of freedom 2 (0
        if DOF 2 is not active in the model).
        Column  3  -  etc.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1911
```

```
  ABAQUS output request definition output to MATLAB

  Syntax
      #Rec# = Fil2str('*.fil');
      #out# = Rec1911(#Rec#)

  Description
      Read output request definition from the results (*.fil) file
      generated from the ABAQUS finite element software. The asterisk (*)
      is replaced by the name of the results file. The record key for
      output request definition is 1911. See section < < Results file output
      format > > in ABAQUS Analysis User's manual for more details.
      The following option with parameter has to be specified in the ABAQUS
      input file for the results (*.fil) file to be created:
          ...
          *FILE FORMAT, ASCII
          ...
      NOTE: The results file (*.fil) must be placed in the same directory
      with the MATLAB source files in order to be processed.

  Input parameters
      #Rec# (string) is an one-row string containing the ASCII code of the
          ABAQUS results (*.fil) file. It is generated by the function
          Fil2str.m.

  Output parameters
      #out# ([#n# x 3]) is a cell array containing the attributes of
          the record key 1911 as follows:
          Column  1  -  Flag for element-based output (0), nodal output
          (1), modal output (2), or element set energy output (3).
          Column  2  -  Set name (node or element set) used in the request
          (A8 format). This attribute is blank if no set was specified.
          Column  3  -  Element type (only for element output, A8 format).
          where #n# is the number of output request definitions.

  _________________________________________________________________________
  Abaqus2Matlab - www.abaqus2matlab.com
  Copyright (c) 2016 by George Papazafeiropoulos

  If using this toolbox for research or industrial purposes, please cite:
  G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
  Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1921
```

```
ABAQUS analysis information output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1921(#Rec#)

Description
    Read analysis information output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    analysis information output is 1921. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([1 x #m#]) is a cell array containing the attributes of
        the record key 1921 as follows:
        Column  1  –  Abaqus release number (A8 format).
        Column  2  –  Date (2A8 format).
        Column  3  –  Date continued.
        Column  4  –  Time (A8 format).
        Column  5  –  Number of elements in the model.
        Column  6  –  Number of nodes in the model.
        Column  7  –  Typical element length in the model.
        where #m# is the length of the record.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1940
```

```
ABAQUS label cross-reference output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1940(#Rec#)

Description
    Read label cross-reference output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for label
    cross-reference output is 1940. See section < < Results file output
    format > > in ABAQUS Analysis User's manual for more details.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 2]) is a cell array containing the attributes of
        the record key 1940 as follows:
        Column  1  –  Integer reference.
        Column  2  –  Label (10A8 format).
        where #n# is the number of elements.

_________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec1980
```

```
ABAQUS modal output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1980(#Rec#)

Description
    Read modal output from the results (*.fil) file generated from the
    ABAQUS finite element software. The asterisk (*) is replaced by the
    name of the results file. The record key for modal output is 1980 (in
    ABAQUS/Standard) and is written once per eigenvalue in a natural
    frequency extraction step. See section < < Results file output format > >
    in ABAQUS Analysis User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and
    contain frequency analysis results (#n# is the number of requested
    eigenvalues):
        ...
        *FREQUENCY
        #n#
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 16]) is a double array containing the attributes of the
        record key 1980 as follows:
        Column  1  –  Eigenvalue number.
        Column  2  –  Eigenvalue.
        Column  3  –  Generalized mass.
        Column  4  –  Composite damping.
        Column  5  –  Participation factor for degree of freedom 1.
        Column  6  –  Effective mass for degree of freedom 1.
        Column  7  –  Participation factor for degree of freedom 2.
        Column  8  –  Effective mass for degree of freedom 2.
        Column  9  –  Participation factor for degree of freedom 3.
        Column 10  –  Effective mass for degree of freedom 3.
        Column 11  –  Participation factor for degree of freedom 4.
        Column 12  –  Effective mass for degree of freedom 4.
        Column 13  –  Participation factor for degree of freedom 5.
        Column 14  –  Effective mass for degree of freedom 5.
        Column 15  –  Participation factor for degree of freedom 6.
        Column 16  –  Effective mass for degree of freedom 6.
        where #n# is the number of requested eigenvalues in the frequency
        extraction step. If the results file does not contain the desired
        output, #out# will be an empty array
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)

```
help Rec1991
```

```
ABAQUS J-integral output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec1991(#Rec#)

Description
    Read J-integral output from the results (*.fil) file generated from
    the ABAQUS finite element software. The asterisk (*) is replaced by
    the name of the results file. The record key for J-integral output is
    1991. See section < < Results file output format > > in ABAQUS Analysis
    User's manual for more details.
    The following options with parameters have to be specified in the
    ABAQUS input file for the results (*.fil) file to be created and to
    contain J-integral results:
        ...
        *FILE FORMAT, ASCII
        *EL FILE
        JK
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x #m#]) is a cell array containing the attributes of
        the record key 1991 as follows:
        Column  1  –  Crack number.
        Column  2  –  Node set (A8 format).
        Column  3  -  Number of contours.
        Column  4  –  J-integral value estimated by first contour.
        Column  5  –  J-integral value estimated by second contour.
        Column  6  –  Etc.
        where #n# is the number of cracks multiplied by the crack front
        locations. If the results file does not contain the desired
        output, #out# will be an empty array

_______________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

```
help Rec2000
```

```
ABAQUS increment start record output to MATLAB

Syntax
    #Rec# = Fil2str('*.fil');
    #out# = Rec2000(#Rec#)

Description
    Read increment start record output from the results (*.fil) file
    generated from the ABAQUS finite element software. The asterisk (*)
    is replaced by the name of the results file. The record key for
    increment start record output is 2000. See section < < Results file
    output format > > in ABAQUS Analysis User's manual for more details.
    The following option with parameter has to be specified in the ABAQUS
    input file for the results (*.fil) file to be created:
        ...
        *FILE FORMAT, ASCII
        ...
    NOTE: The results file (*.fil) must be placed in the same directory
    with the MATLAB source files in order to be processed.

Input parameters
    #Rec# (string) is an one-row string containing the ASCII code of the
        ABAQUS results (*.fil) file. It is generated by the function
        Fil2str.m.

Output parameters
    #out# ([#n# x 21]) is a cell array containing the attributes of
        the record key 2000 as follows:
        Column  1  –  Total time.
        Column  2  –  Step time.
        Column  3  –  Maximum creep strain-rate ratio (control of
        solution-dependent amplitude) in Abaqus/Standard; currently not
        used in Abaqus/Explicit.
        Column  4  –  Solution-dependent amplitude in Abaqus/Standard;
        currently not used in Abaqus/Explicit.
        Column  5  –  Procedure type: gives a key to the step type. See
        Table 5.1.2–1 at the end of this section.
        Column  6  –  Step number.
        Column  7  –  Increment number.
        Column  8  –  Linear perturbation flag in Abaqus/Standard: 0 if
        general step, 1 if linear perturbation step; currently not used
        in Abaqus/Explicit.
        Column  9  –  Load proportionality factor: nonzero only in static
        Riks steps; currently not used in Abaqus/Explicit.
        Column  10  –  Frequency (cycles/time) in a steady-state dynamic
        response analysis or steady-state transport angular velocity
        (rad/time) in a steady-state transport analysis; currently not
        used in Abaqus/Explicit.
        Column  11  –  Time increment.
        Columns  12-21  –  The step subheading entered as the first data
        line of the *STEP option (A8 format). Equivalent to the step
        description in Abaqus/CAE.
        where #n# is the number of increments.
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing (submitted)
```

COORDINATE output from Abaqus to Matlab (Record key 8)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\8.inp'],[S(1:a(end)-1),'\8.inp'],'f')
```

Run the input file 8.inp with Abaqus

```
!abaqus job=8
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('8.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('8.fil');
```

Obtain the desired output data

```
out = Rec8(Rec)
```

```
out =

  436.0770   76.0770         0
  643.9230   76.0770         0
  436.0770  283.9230         0
  643.9230  283.9230         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

NODAL FLUX CAUSED BY HEAT output from Abaqus to Matlab (Record key 10)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\10.inp'],[S(1:a(end)-1),'\10.inp'],'f')
```

Run the input file 10.inp with Abaqus

```
!abaqus job=10
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('10.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('10.fil');
```

Obtain the desired output data

```
out = Rec10(Rec)
```

```
out =

    1.0000   -0.0485
   17.0000   -0.0291
    3.0000    0.0585
    9.0000   -0.1280
   10.0000   -0.1081
    2.0000    0.2552
    3.0000   -0.0000
   19.0000   -0.0549
    5.0000    0.0972
   11.0000   -0.2670
   12.0000   -0.0819
    4.0000    0.3067
    5.0000    0.0000
   21.0000   -0.0632
    7.0000    0.1051
   13.0000   -0.3624
   14.0000   -0.0500
    6.0000    0.3705
   15.0000    0.0394
   31.0000    0.0059
   17.0000   -0.0029
   23.0000    0.2088
   24.0000   -0.2479
   16.0000   -0.0034
   17.0000    0.0320
   33.0000   -0.0325
   19.0000    0.0386
   25.0000   -0.0261
   26.0000   -0.1484
   18.0000    0.1364
   19.0000    0.0163
   35.0000   -0.0363
   21.0000    0.0632
   27.0000   -0.1960
   28.0000   -0.0500
   20.0000    0.2029
   29.0000   -0.0473
   45.0000   -0.0141
   31.0000   -0.0900
   37.0000   -0.0391
   38.0000   -0.2927
   30.0000    0.4831
   31.0000    0.0840
   47.0000   -0.0172
   33.0000    0.0040
   39.0000    0.0437
   40.0000   -0.1791
   32.0000    0.0646
   33.0000    0.0284
   49.0000   -0.0174
   35.0000    0.0363
   41.0000   -0.0927
   42.0000   -0.0500
   34.0000    0.0953
   43.0000    0.0971
   59.0000   -0.0000
   45.0000   -0.0496
```

51.0000	0.2743
52.0000	-0.3163
44.0000	-0.0056
45.0000	0.0636
61.0000	0.0000
47.0000	-0.0153
53.0000	0.1143
54.0000	-0.1807
46.0000	0.0181
47.0000	0.0325
63.0000	-0.0000
49.0000	0.0174
55.0000	-0.0195
56.0000	-0.0500
48.0000	0.0196
1.0000	-0.0000
15.0000	0.0394
17.0000	-0.0320
8.0000	0.0612
16.0000	-0.0966
9.0000	0.0280
3.0000	0.0747
17.0000	0.0029
19.0000	-0.0163
10.0000	0.0081
18.0000	-0.2364
11.0000	0.1670
5.0000	0.0972
19.0000	-0.0386
21.0000	-0.0000
12.0000	-0.0181
20.0000	-0.3029
13.0000	0.2624
15.0000	0.0000
29.0000	-0.0473
31.0000	-0.1420
22.0000	0.1544
30.0000	0.3436
23.0000	-0.3088
17.0000	0.0291
31.0000	0.0900
33.0000	-0.0284
24.0000	0.1479
32.0000	-0.1646
25.0000	-0.0739
19.0000	0.0549
33.0000	-0.0040
35.0000	0.0000
26.0000	0.0484
34.0000	-0.1953
27.0000	0.0960
29.0000	0.0860
43.0000	0.1830
45.0000	-0.0636
36.0000	-0.0500
44.0000	-0.0944
37.0000	-0.0609
31.0000	0.0520
45.0000	0.0496
47.0000	-0.0325
38.0000	0.1927

```
   46.0000   -0.1181
   39.0000   -0.1437
   33.0000    0.0325
   47.0000    0.0153
   49.0000   -0.0000
   40.0000    0.0791
   48.0000   -0.1196
   41.0000   -0.0073
   43.0000    0.0000
   57.0000    0.1082
   59.0000   -0.0666
   50.0000    0.3827
   58.0000   -0.0500
   51.0000   -0.3743
   45.0000    0.0141
   59.0000    0.0666
   61.0000   -0.0327
   52.0000    0.2163
   60.0000   -0.0500
   53.0000   -0.2143
   47.0000    0.0172
   61.0000    0.0327
   63.0000   -0.0000
   54.0000    0.0807
   62.0000   -0.0500
   55.0000   -0.0805
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

   144
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

STRESS output from Abaqus to Matlab (Record key 11)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\11.inp'],[S(1:a(end)-1),'\11.inp'],'f')
```

Run the input file 11.inp with Abaqus

```
!abaqus job=11
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('11.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('11.fil');
```

Obtain the desired output data

```
out = Rec11(Rec)
```

```
out =

   1.0e-03 *

    0.9509   -0.9509   -0.0000   -0.9509
   -0.9509    0.9509   -0.0000   -0.9509
    0.9509   -0.9509   -0.0000    0.9509
   -0.9509    0.9509   -0.0000    0.9509
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     4
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

STRESS INVARIANT output from Abaqus to Matlab (Record key 12)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\12.inp'],[S(1:a(end)-1),'\12.inp'],'f')
```

Run the input file 12.inp with Abaqus

```
!abaqus job=12
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('12.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('12.fil');
```

Obtain the desired output data

```
out = Rec12(Rec)
```

```
out =

  406.1556  407.3403 -141.4196    4.8461    7.2261  412.1865  406.1399
  406.1556  407.3403 -141.4196    4.8461    7.2261  412.1865  406.1399
  406.1556  407.3403 -141.4196    4.8461    7.2261  412.1865  406.1399
  406.1556  407.3403 -141.4196    4.8461    7.2261  412.1865  406.1399
  319.3311  320.8367 -141.4196   33.4631   36.4958  354.2998  319.2987
  319.3311  320.8367 -141.4196   33.4631   36.4958  354.2998  319.2987
  319.3311  320.8367 -141.4196   33.4631   36.4958  354.2998  319.2987
  319.3311  320.8367 -141.4196   33.4631   36.4958  354.2998  319.2987
   64.9397   65.1517  -19.8075   -2.0517   -1.6257   63.1000   64.9366
   64.9397   65.1517  -19.8075   -2.0517   -1.6257   63.1000   64.9366
   64.9397   65.1517  -19.8075   -2.0517   -1.6257   63.1000   64.9366
   64.9397   65.1517  -19.8075   -2.0517   -1.6257   63.1000   64.9366
   50.2762   50.5497  -19.8075    2.7738    3.3253   53.3235   50.2694
   50.2762   50.5497  -19.8075    2.7738    3.3253   53.3235   50.2694
   50.2762   50.5497  -19.8075    2.7738    3.3253   53.3235   50.2694
   50.2762   50.5497  -19.8075    2.7738    3.3253   53.3235   50.2694
  321.4689  322.3693 -116.7404    8.6811   10.4897  331.0505  321.4574
  321.4689  322.3693 -116.7404    8.6811   10.4897  331.0505  321.4574
  321.4689  322.3693 -116.7404    8.6811   10.4897  331.0505  321.4574
  321.4689  322.3693 -116.7404    8.6811   10.4897  331.0505  321.4574
  254.0940  255.2323 -116.7404   30.8990   33.1911  286.1312  254.0707
  254.0940  255.2323 -116.7404   30.8990   33.1911  286.1312  254.0707
  254.0940  255.2323 -116.7404   30.8990   33.1911  286.1312  254.0707
  254.0940  255.2323 -116.7404   30.8990   33.1911  286.1312  254.0707
  140.4671  140.9399  -41.1221   -6.1747   -5.2242  134.7652  140.4598
  140.4671  140.9399  -41.1221   -6.1747   -5.2242  134.7652  140.4598
  140.4671  140.9399  -41.1221   -6.1747   -5.2242  134.7652  140.4598
  140.4671  140.9399  -41.1221   -6.1747   -5.2242  134.7652  140.4598
  108.2716  108.8843  -41.1221    4.4153    5.6514  113.2996  108.2557
  108.2716  108.8843  -41.1221    4.4153    5.6514  113.2996  108.2557
  108.2716  108.8843  -41.1221    4.4153    5.6514  113.2996  108.2557
  108.2716  108.8843  -41.1221    4.4153    5.6514  113.2996  108.2557
  260.8741  261.5708  -99.2264   11.5698   12.9689  273.1406  260.8656
  260.8741  261.5708  -99.2264   11.5698   12.9689  273.1406  260.8656
  260.8741  261.5708  -99.2264   11.5698   12.9689  273.1406  260.8656
  260.8741  261.5708  -99.2264   11.5698   12.9689  273.1406  260.8656
  207.4592  208.3346  -99.2264   29.1942   30.9563  237.5288  207.4423
  207.4592  208.3346  -99.2264   29.1942   30.9563  237.5288  207.4423
  207.4592  208.3346  -99.2264   29.1942   30.9563  237.5288  207.4423
  207.4592  208.3346  -99.2264   29.1942   30.9563  237.5288  207.4423
  202.8851  203.5826  -57.6645  -10.6638   -9.2614  192.9188  202.8742
  202.8851  203.5826  -57.6645  -10.6638   -9.2614  192.9188  202.8742
  202.8851  203.5826  -57.6645  -10.6638   -9.2614  192.9188  202.8742
  202.8851  203.5826  -57.6645  -10.6638   -9.2614  192.9188  202.8742
  155.9041  156.8108  -57.6645    4.7845    6.6139  161.5952  155.8799
  155.9041  156.8108  -57.6645    4.7845    6.6139  161.5952  155.8799
  155.9041  156.8108  -57.6645    4.7845    6.6139  161.5952  155.8799
  155.9041  156.8108  -57.6645    4.7845    6.6139  161.5952  155.8799
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     7
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    48
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

SECTION FORCE AND MOMENT output from Abaqus to Matlab (Record key 13)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\13.inp'],[S(1:a(end)-1),'\13.inp'],'f')
```

Run the input file 13.inp with Abaqus

```
!abaqus job=13
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('13.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('13.fil');
```

Obtain the desired output data

```
out = Rec13(Rec)
```

```
out =

   84.2742         0         0
   84.2742         0         0
   84.2742         0         0
   -8.4125         0         0
   -8.4125         0         0
   -8.4125         0         0
  -95.4174         0         0
  -95.4174         0         0
  -95.4174         0         0
  -22.6873         0         0
  -22.6873         0         0
  -22.6873         0         0
   63.4216         0         0
   63.4216         0         0
   63.4216         0         0
   72.0687         0         0
   72.0687         0         0
   72.0687         0         0
  118.5554         0         0
  118.5554         0         0
  118.5554         0         0
  -92.4162         0         0
  -92.4162         0         0
  -92.4162         0         0
   55.0484         0         0
   55.0484         0         0
   55.0484         0         0
   -2.8531         0         0
   -2.8531         0         0
   -2.8531         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    30
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

ENERGY DENSITY output from Abaqus to Matlab (Record key 14)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\14.inp'],[S(1:a(end)-1),'\14.inp'],'f')
```

Run the input file 14.inp with Abaqus

```
!abaqus job=14
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('14.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('14.fil');
```

Obtain the desired output data

```
out = Rec14(Rec)
```

```
out =

    0.0145         0    0.0000         0         0         0         0
    0.0150         0    0.0000         0         0         0         0
    0.0140         0    0.0000         0         0         0         0
    0.0147         0    0.0000         0         0         0         0
    0.0461         0    0.0000         0         0         0         0
    0.0479    0.0005    0.0000         0         0         0         0
    0.0463         0    0.0000         0         0         0         0
    0.0485         0    0.0000         0         0         0         0
    0.0828    0.0023    0.0000         0         0         0         0
    0.0870    0.0044    0.0000         0         0         0         0
    0.0834         0    0.0000         0         0         0         0
    0.0877    0.0002    0.0000         0         0         0         0
    0.1305    0.0065    0.0000         0         0         0         0
    0.1418    0.0123    0.0000         0         0         0         0
    0.1307    0.0002    0.0000         0         0         0         0
    0.1372    0.0018    0.0000         0         0         0         0
    0.1837    0.0125    0.0000         0         0         0         0
    0.2140    0.0281    0.0000         0         0         0         0
    0.1854    0.0002    0.0000         0         0         0         0
    0.1895    0.0034    0.0000         0         0         0         0
    0.2490    0.0230    0.0001         0         0         0         0
    0.3211    0.0586    0.0001         0         0         0         0
    0.2563    0.0002    0.0000         0         0         0         0
    0.2495    0.0059    0.0000         0         0         0         0
    0.2970    0.0411    0.0001         0         0         0         0
    0.4303    0.1026    0.0001         0         0         0         0
    0.3070    0.0002    0.0000         0         0         0         0
    0.2930    0.0085    0.0001         0         0         0         0
    0.3102    0.0670    0.0001         0         0         0         0
    0.5037    0.1524    0.0001         0         0         0         0
    0.3152    0.0002    0.0000         0         0         0         0
    0.3089    0.0105    0.0001         0         0         0         0
    0.3118    0.0912    0.0001         0         0         0         0
    0.5534    0.1993    0.0001         0         0         0         0
    0.3239    0.0002    0.0000         0         0         0         0
    0.3261    0.0125    0.0001         0         0         0         0
    0.3108    0.1174    0.0001         0         0         0         0
    0.5934    0.2462    0.0001         0         0         0         0
    0.3312    0.0002    0.0000         0         0         0         0
    0.3489    0.0147    0.0001         0         0         0         0
    0.3091    0.1462    0.0002         0         0         0         0
    0.6270    0.2941    0.0002         0         0         0         0
    0.3359    0.0002    0.0000         0         0         0         0
    0.3748    0.0172    0.0002         0         0         0         0
    0.3074    0.1772    0.0002         0         0         0         0
    0.6550    0.3430    0.0002         0         0         0         0
    0.3380    0.0002    0.0000         0         0         0         0
    0.4022    0.0199    0.0002         0         0         0         0
    0.3060    0.2100    0.0002         0         0         0         0
    0.6778    0.3929    0.0002         0         0         0         0
    0.3383    0.0002    0.0000         0         0         0         0
    0.4302    0.0228    0.0002         0         0         0         0
    0.3054    0.2257    0.0002         0         0         0         0
    0.6873    0.4165    0.0002         0         0         0         0
    0.3382    0.0002    0.0000         0         0         0         0
    0.4436    0.0242    0.0002         0         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     7
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    56
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```


```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

PORE OR ACOUSTIC PRESSURE output from Abaqus to Matlab (Record key 18)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\18.inp'],[S(1:a(end)-1),'\18.inp'],'f')
```

Run the input file 18.inp with Abaqus

```
!abaqus job=18
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('18.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('18.fil');
```

Obtain the desired output data

```
out = Rec18(Rec)
```

```
out =

         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
   10.5396
   10.5396
    2.8241
    2.8241
    2.6627
    9.9373
    2.6627
    9.9373
    2.8241
    2.8241
   10.5396
   10.5396
    9.9315
    2.6611
    9.9370
    2.6626
  -11.8301
  -11.8301
   -3.1699
   -3.1699
   -1.9019
   -7.0981
   -1.9019
   -7.0981
   -3.1699
   -3.1699
  -11.8301
  -11.8301
   -7.0981
```

```
   -1.9019
   -7.0981
   -1.9019
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    60
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

ENERGY (SUMMED OVER ELEMENT) output from Abaqus to Matlab (Record key 19)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\19.inp'],[S(1:a(end)-1),'\19.inp'],'f')
```

Run the input file 19.inp with Abaqus

```
!abaqus job=19
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('19.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('19.fil');
```

Obtain the desired output data

```
out = Rec19(Rec)
```

```
out =

  Columns 1 through 7

         0   10.4933         0    0.0004         0         0         0
         0   33.9706    0.0849    0.0028         0         0         0
         0   61.3704    1.2326    0.0065         0         0         0
         0   97.2563    3.7412    0.0127         0         0         0
         0  139.0694    7.9590    0.0215         0         0         0
         0  193.6649   15.7786    0.0332         0         0         0
         0  238.9220   27.4158    0.0456         0         0         0
         0  258.8365   41.4062    0.0579         0         0         0
         0  272.7242   54.5571    0.0703         0         0         0
         0  285.1731   68.1412    0.0827         0         0         0
         0  296.4056   82.3816    0.0951         0         0         0
         0  306.4604   97.2428    0.1075         0         0         0
         0  315.4203  112.6445    0.1199         0         0         0
         0  319.4233  119.9770    0.1258         0         0         0

  Columns 8 through 10

         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

    10
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =
```

```
    14
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

____

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

TOTAL STRAIN output from Abaqus to Matlab (Record key 21)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\21.inp'],[S(1:a(end)-1),'\21.inp'],'f')
```

Run the input file 21.inp with Abaqus

```
!abaqus job=21
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('21.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('21.fil');
```

Obtain the desired output data

```
out = Rec21(Rec)
```

```
out =

   -0.0012   -0.0012    0.0041    0.0000   -0.0006   -0.0006
   -0.0012   -0.0012    0.0041    0.0000    0.0006   -0.0006
   -0.0012   -0.0012    0.0041   -0.0000   -0.0006    0.0006
   -0.0012   -0.0012    0.0041   -0.0000    0.0006    0.0006
   -0.0008   -0.0008    0.0033    0.0000   -0.0006   -0.0006
   -0.0008   -0.0008    0.0033    0.0000    0.0006   -0.0006
   -0.0008   -0.0008    0.0033   -0.0000   -0.0006    0.0006
   -0.0008   -0.0008    0.0033   -0.0000    0.0006    0.0006
   -0.0002   -0.0002    0.0006   -0.0000   -0.0001   -0.0001
   -0.0002   -0.0002    0.0006   -0.0000    0.0001   -0.0001
   -0.0002   -0.0002    0.0006    0.0000   -0.0001    0.0001
   -0.0002   -0.0002    0.0006    0.0000    0.0001    0.0001
   -0.0001   -0.0001    0.0005   -0.0000   -0.0001   -0.0001
   -0.0001   -0.0001    0.0005   -0.0000    0.0001   -0.0001
   -0.0001   -0.0001    0.0005    0.0000   -0.0001    0.0001
   -0.0001   -0.0001    0.0005    0.0000    0.0001    0.0001
   -0.0009   -0.0009    0.0032    0.0000   -0.0004   -0.0004
   -0.0009   -0.0009    0.0032   -0.0000    0.0004   -0.0004
   -0.0009   -0.0009    0.0032    0.0000   -0.0004    0.0004
   -0.0009   -0.0009    0.0032   -0.0000    0.0004    0.0004
   -0.0006   -0.0006    0.0026    0.0000   -0.0004   -0.0004
   -0.0006   -0.0006    0.0026   -0.0000    0.0004   -0.0004
   -0.0006   -0.0006    0.0026    0.0000   -0.0004    0.0004
   -0.0006   -0.0006    0.0026   -0.0000    0.0004    0.0004
   -0.0004   -0.0004    0.0014    0.0000   -0.0002   -0.0002
   -0.0004   -0.0004    0.0014    0.0000    0.0002   -0.0002
   -0.0004   -0.0004    0.0014    0.0000   -0.0002    0.0002
   -0.0004   -0.0004    0.0014    0.0000    0.0002    0.0002
   -0.0003   -0.0003    0.0011    0.0000   -0.0002   -0.0002
   -0.0003   -0.0003    0.0011    0.0000    0.0002   -0.0002
   -0.0003   -0.0003    0.0011    0.0000   -0.0002    0.0002
   -0.0003   -0.0003    0.0011    0.0000    0.0002    0.0002
   -0.0007   -0.0007    0.0026    0.0000   -0.0004   -0.0004
   -0.0007   -0.0007    0.0026   -0.0000    0.0004   -0.0004
   -0.0007   -0.0007    0.0026    0.0000   -0.0004    0.0004
   -0.0007   -0.0007    0.0026   -0.0000    0.0004    0.0004
   -0.0005   -0.0005    0.0022    0.0000   -0.0004   -0.0004
   -0.0005   -0.0005    0.0022   -0.0000    0.0004   -0.0004
   -0.0005   -0.0005    0.0022    0.0000   -0.0004    0.0004
   -0.0005   -0.0005    0.0022   -0.0000    0.0004    0.0004
   -0.0006   -0.0006    0.0020   -0.0000   -0.0003   -0.0003
   -0.0006   -0.0006    0.0020   -0.0000    0.0003   -0.0003
   -0.0006   -0.0006    0.0020    0.0000   -0.0003    0.0003
   -0.0006   -0.0006    0.0020    0.0000    0.0003    0.0003
   -0.0004   -0.0004    0.0016   -0.0000   -0.0003   -0.0003
   -0.0004   -0.0004    0.0016   -0.0000    0.0003   -0.0003
   -0.0004   -0.0004    0.0016    0.0000   -0.0003    0.0003
   -0.0004   -0.0004    0.0016    0.0000    0.0003    0.0003
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    48
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

PLASTIC STRAIN output from Abaqus to Matlab (Record key 22)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\22.inp'],[S(1:a(end)-1),'\22.inp'],'f')
```

Run the input file 22.inp with Abaqus

```
!abaqus job=22
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('22.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('22.fil');
```

Obtain the desired output data

```
out = Rec22(Rec)
```

```
out =

         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
    0.0000   -0.0000    0.0000   -0.0000    0.0000    1.0000    0.0000
         0         0         0         0         0         0         0
         0         0         0         0         0         0         0
    0.0001   -0.0001    0.0001   -0.0000    0.0001    1.0000    0.0002
    0.0002   -0.0002    0.0002   -0.0000    0.0002    1.0000    0.0003
         0         0         0         0         0         0         0
    0.0000   -0.0000    0.0000    0.0000    0.0000    1.0000    0.0000
    0.0003   -0.0003    0.0003   -0.0000    0.0003    1.0000    0.0004
    0.0007   -0.0006    0.0004   -0.0001    0.0006    1.0000    0.0008
    0.0000   -0.0000    0.0000    0.0000    0.0000    1.0000    0.0000
    0.0001   -0.0001    0.0001    0.0000    0.0001    1.0000    0.0001
    0.0006   -0.0006    0.0005   -0.0001    0.0006    1.0000    0.0008
    0.0016   -0.0014    0.0008   -0.0002    0.0014    1.0000    0.0018
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0001   -0.0002    0.0002    0.0000    0.0002    1.0000    0.0002
    0.0012   -0.0011    0.0007   -0.0001    0.0012    1.0000    0.0015
    0.0036   -0.0027    0.0013   -0.0005    0.0030    1.0000    0.0038
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0003   -0.0003    0.0003    0.0001    0.0003    1.0000    0.0004
    0.0024   -0.0020    0.0011   -0.0003    0.0021    1.0000    0.0027
    0.0066   -0.0045    0.0018   -0.0010    0.0052    1.0000    0.0067
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0004   -0.0004    0.0004    0.0001    0.0004    1.0000    0.0006
    0.0041   -0.0031    0.0014   -0.0008    0.0034    1.0000    0.0044
    0.0100   -0.0065    0.0021   -0.0018    0.0078    1.0000    0.0099
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0005   -0.0005    0.0004    0.0002    0.0005    1.0000    0.0007
    0.0058   -0.0040    0.0016   -0.0012    0.0047    1.0000    0.0059
    0.0133   -0.0082    0.0023   -0.0026    0.0101    1.0000    0.0130
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0006   -0.0006    0.0005    0.0002    0.0006    1.0000    0.0008
    0.0076   -0.0050    0.0017   -0.0018    0.0060    1.0000    0.0076
    0.0165   -0.0099    0.0024   -0.0035    0.0125    1.0000    0.0160
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0007   -0.0007    0.0006    0.0003    0.0007    1.0000    0.0010
    0.0096   -0.0060    0.0018   -0.0025    0.0074    1.0000    0.0095
    0.0199   -0.0116    0.0025   -0.0044    0.0149    1.0000    0.0191
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0008   -0.0008    0.0007    0.0004    0.0009    1.0000    0.0011
    0.0118   -0.0070    0.0018   -0.0032    0.0090    1.0000    0.0114
    0.0233   -0.0133    0.0026   -0.0054    0.0174    1.0000    0.0222
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0009   -0.0010    0.0008    0.0004    0.0010    1.0000    0.0013
    0.0140   -0.0081    0.0018   -0.0041    0.0107    1.0000    0.0135
    0.0268   -0.0151    0.0027   -0.0064    0.0199    1.0000    0.0255
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0011   -0.0011    0.0009    0.0005    0.0012    1.0000    0.0015
    0.0151   -0.0087    0.0018   -0.0046    0.0115    1.0000    0.0146
    0.0284   -0.0159    0.0027   -0.0068    0.0211    1.0000    0.0270
    0.0000   -0.0000    0.0000    0.0000    0.0000         0    0.0000
    0.0011   -0.0012    0.0009    0.0006    0.0012    1.0000    0.0016
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     7
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    56
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

CREEP STRAIN (INCLUDING SWELLING) output from Abaqus to Matlab (Record key 23)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\23.inp'],[S(1:a(end)-1),'\23.inp'],'f')
```

Run the input file 23.inp with Abaqus

```
!abaqus job=23
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('23.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('23.fil');
```

Obtain the desired output data

```
out = Rec23(Rec)
```

```
out =

  Columns 1 through 7

   -0.0017   -0.0017   -0.0015         0         0         0    0.0001
   -0.0066   -0.0066   -0.0050         0         0         0    0.0010
   -0.0149   -0.0149   -0.0098         0         0         0    0.0034
   -0.0274   -0.0274   -0.0155         0         0         0    0.0079
   -0.0453   -0.0453   -0.0222         0         0         0    0.0154

  Columns 8 through 9

         0    0.0023
         0    0.0086
         0    0.0190
         0    0.0340
         0    0.0554
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     9
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     5
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

TOTAL INELASTIC STRAIN output from Abaqus to Matlab (Record key 24)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\24.inp'],[S(1:a(end)-1),'\24.inp'],'f')
```

Run the input file 24.inp with Abaqus

```
!abaqus job=24
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('24.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('24.fil');
```

Obtain the desired output data

```
out = Rec24(Rec)
```

```
out =

   -0.0017   -0.0017   -0.0015         0         0         0
   -0.0066   -0.0066   -0.0050         0         0         0
   -0.0149   -0.0149   -0.0098         0         0         0
   -0.0274   -0.0274   -0.0155         0         0         0
   -0.0453   -0.0453   -0.0222         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     5
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Published with MATLAB® R2015a

TOTAL ELASTIC STRAIN output from Abaqus to Matlab (Record key 25)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\25.inp'],[S(1:a(end)-1),'\25.inp'],'f')
```

Run the input file 25.inp with Abaqus

```
!abaqus job=25
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('25.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('25.fil');
```

Obtain the desired output data

```
out = Rec25(Rec)
```

```
out =

    0.0045    0.0045   -0.0394         0         0         0
    0.0081    0.0081   -0.0784         0         0         0
    0.0106    0.0106   -0.1180         0         0         0
    0.0118    0.0118   -0.1588         0         0         0
    0.0112    0.0112   -0.2010         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     5
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

UNIT NORMAL TO CRACK IN CONCRETE output from Abaqus to Matlab (Record key 26)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\26.inp'],[S(1:a(end)-1),'\26.inp'],'f')
```

Run the input file 26.inp with Abaqus

```
!abaqus job=26
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('26.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('26.fil');
```

Obtain the desired output data

```
out = Rec26(Rec)
```

```
out =

     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
     0     0     1     0     0     0     0     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     9
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing

```
(submitted)
```

Published with MATLAB® R2015a

SECTION THICKNESS output from Abaqus to Matlab (Record key 27)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\27.inp'],[S(1:a(end)-1),'\27.inp'],'f')
```

Run the input file 27.inp with Abaqus

```
!abaqus job=27
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('27.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('27.fil');
```

Obtain the desired output data

```
out = Rec27(Rec)
```

```
out =

    2.9549
    2.9549
    2.9549
    2.8000
    2.8000
    2.8000
    2.6451
    2.6451
    2.6451
    2.5549
    2.5549
    2.5549
    2.4000
    2.4000
    2.4000
    2.2451
    2.2451
    2.2451
    2.1549
    2.1549
    2.1549
    2.0000
    2.0000
    2.0000
    1.8451
    1.8451
    1.8451
    1.7549
    1.7549
    1.7549
    1.6000
    1.6000
    1.6000
    1.4451
    1.4451
    1.4451
    1.3549
    1.3549
    1.3549
    1.2000
    1.2000
    1.2000
    1.0451
    1.0451
    1.0451
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =
```

```
     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    45
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```



```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

HEAT FLUX VECTOR output from Abaqus to Matlab (Record key 28)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\28.inp'],[S(1:a(end)-1),'\28.inp'],'f')
```

Run the input file 28.inp with Abaqus

```
!abaqus job=28
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('28.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('28.fil');
```

Obtain the desired output data

```
out = Rec28(Rec)
```

```
out =

   172.8815         0 -172.8815
   172.8815         0 -172.8815
   172.8815   -0.0000 -172.8815
   172.8815   -0.0000 -172.8815
   335.3638         0 -335.3638
   335.3638         0 -335.3638
   335.3638         0 -335.3638
   335.3638         0 -335.3638
   437.4658   -0.0000 -437.4658
   437.4658   -0.0000 -437.4658
   437.4658         0 -437.4658
   437.4658         0 -437.4658
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    12
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

SECTION STRAIN AND CURVATURE output from Abaqus to Matlab (Record key 29)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\29.inp'],[S(1:a(end)-1),'\29.inp'],'f')
```

Run the input file 29.inp with Abaqus

```
!abaqus job=29
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('29.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('29.fil');
```

Obtain the desired output data

```
out = Rec29(Rec)
```

```
out =

   1.0e-03 *

    0.0000    0.0001   -0.0000
   -0.0000   -0.0000   -0.0000
   -0.0000   -0.1319   -0.0000
   -0.0000   -0.0930    0.0000
    0.1322    0.0002    0.0000
    0.2255    0.0001    0.0000
    0.0661   -0.0659    0.0000
   -0.0000    0.0002   -0.0000
    0.1127   -0.1125    0.0000
    0.0661    0.0662   -0.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

CONCRETE FAILURE output from Abaqus to Matlab (Record key 31)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\31.inp'],[S(1:a(end)-1),'\31.inp'],'f')
```

Run the input file 31.inp with Abaqus

```
!abaqus job=31
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('31.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('31.fil');
```

Obtain the desired output data

```
out = Rec31(Rec)
```

```
out =

    [ 1]    '1 CRACK'
    [ 2]    '1 CRACK'
    [ 3]    '1 CRACK'
    [ 4]    '1 CRACK'
    [ 5]    '1 CRACK'
    [ 6]    '1 CRACK'
    [ 7]    '1 CRACK'
    [ 8]    '1 CRACK'
    [ 9]    '1 CRACK'
    [10]    '1 CRACK'
    [11]    '1 CRACK'
    [12]    '1 CRACK'
    [13]    '1 CRACK'
    [14]    '1 CRACK'
    [15]    '1 CRACK'
    [16]    '1 CRACK'
    [17]    '1 CRACK'
    [18]    '1 CRACK'
    [19]    '1 CRACK'
    [20]    '1 CRACK'
    [21]    '1 CRACK'
    [22]    '1 CRACK'
    [23]    '1 CRACK'
    [24]    '1 CRACK'
    [25]    '1 CRACK'
    [26]    '1 CRACK'
    [27]    '1 CRACK'
    [28]    '1 CRACK'
    [29]    '1 CRACK'
    [30]    '1 CRACK'
    [31]    '1 CRACK'
    [32]    '1 CRACK'
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =
```

```
    32
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

STRAIN JUMP AT NODES output from Abaqus to Matlab (Record key 32)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\32.inp'],[S(1:a(end)-1),'\32.inp'],'f')
```

Run the input file 32.inp with Abaqus

```
!abaqus job=32
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('32.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('32.fil');
```

Obtain the desired output data

```
out = Rec32(Rec)
```

```
out =

         0         0         0
    0.0125    0.0000    0.0000
    0.0125    0.0000    0.0000
         0         0         0
    0.0000    0.0002    0.0000
    0.0375    0.0354    0.0167
    0.0375    0.0354    0.0167
    0.0000    0.0002    0.0000
    0.0000    0.0002    0.0000
    0.0375    0.0354    0.0167
    0.0375    0.0354    0.0167
         0    0.0002    0.0000
         0         0         0
    0.0125    0.0000    0.0000
    0.0125    0.0000    0.0000
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
    0.0125    0.0000    0.0000
    0.0125    0.0000    0.0000
         0         0         0
    0.0000    0.0002    0.0000
    0.0375    0.0354    0.0167
    0.0375    0.0354    0.0167
    0.0000    0.0002    0.0000
    0.0000    0.0002    0.0000
    0.0375    0.0354    0.0167
    0.0375    0.0354    0.0167
         0    0.0002    0.0000
         0         0         0
    0.0125    0.0000         0
    0.0125    0.0000    0.0000
         0         0         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```


```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

FILM output from Abaqus to Matlab (Record key 33)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\33.inp'],[S(1:a(end)-1),'\33.inp'],'f')
```

Run the input file 33.inp with Abaqus

```
!abaqus job=33
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('33.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('33.fil');
```

Obtain the desired output data

```
out = Rec33(Rec)
```

```
out =

    'F3'    [102.7778]    [10.1004]
    'F3'    [105.5556]    [10.1642]
    'F3'    [108.3333]    [10.2122]
    'F3'    [111.1111]    [10.2523]
    'F3'    [113.8889]    [10.2880]
    'F3'    [116.6667]    [10.3209]
    'F3'    [119.4444]    [10.3519]
    'F3'    [122.2222]    [10.3815]
    'F3'    [     125]    [10.4100]
    'F3'    [127.7778]    [10.4377]
    'F3'    [130.5556]    [10.4649]
    'F3'    [133.3333]    [10.4915]
    'F3'    [136.1111]    [10.5178]
    'F3'    [138.8889]    [10.5438]
    'F3'    [141.6667]    [10.5695]
    'F3'    [144.4444]    [10.5950]
    'F3'    [147.2222]    [10.6204]
    'F3'    [     150]    [10.6457]
    'F3'    [152.7778]    [10.6710]
    'F3'    [155.5556]    [10.6962]
    'F3'    [158.3333]    [10.7213]
    'F3'    [161.1111]    [10.7465]
    'F3'    [163.8889]    [10.7717]
    'F3'    [166.6667]    [10.7969]
    'F3'    [169.4444]    [10.8221]
    'F3'    [172.2222]    [10.8474]
    'F3'    [     175]    [10.8728]
    'F3'    [177.7778]    [10.8982]
    'F3'    [180.5556]    [10.9238]
    'F3'    [183.3333]    [10.9494]
    'F3'    [186.1111]    [10.9751]
    'F3'    [188.8889]    [11.0010]
    'F3'    [191.6667]    [11.0269]
    'F3'    [194.4444]    [11.0530]
    'F3'    [197.2222]    [11.0792]
    'F3'    [     200]    [11.1056]
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    36
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

RADIATION output from Abaqus to Matlab (Record key 34)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\34.inp'],[S(1:a(end)-1),'\34.inp'],'f')
```

Run the input file 34.inp with Abaqus

```
!abaqus job=34
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('34.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('34.fil');
```

Obtain the desired output data

```
out = Rec34(Rec)
```

```
out =

    'R1'    [75]    [5.0000e-14]
    'R2'    [75]    [5.0000e-14]
    'R3'    [75]    [5.0000e-14]
    'R4'    [75]    [5.0000e-14]
    'R5'    [75]    [5.0000e-14]
    'R6'    [75]    [5.0000e-14]
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

SATURATION (PORE PRESSURE ANALYSIS) output from Abaqus to Matlab (Record key 35)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\35.inp'],[S(1:a(end)-1),'\35.inp'],'f')
```

Run the input file 35.inp with Abaqus

```
!abaqus job=35
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('35.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('35.fil');
```

Obtain the desired output data

```
out = Rec35(Rec)
```

```
out =

    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
    0.0500
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

MASS CONCENTRATION (MASS DIFFUSION ANALYSIS) output from Abaqus to Matlab (Record key 38)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\38.inp'],[S(1:a(end)-1),'\38.inp'],'f')
```

Run the input file 38.inp with Abaqus

```
!abaqus job=38
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('38.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('38.fil');
```

Obtain the desired output data

```
out = Rec38(Rec)
```

```
out =

   1.0e+04 *

         0
         0
    4.8740
         0
    4.8740
    1.6247
    2.7851
    2.7851
    0.0000
    2.7851
    0.0000
    0.0000
         0
    5.5703
    5.5703
         0
    5.5703
         0
         0
         0
    2.7851
         0
    2.7851
    2.7851
    5.5703
         0
         0
    5.5703
         0
    5.5703
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =
```

```
    30
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

GEL (PORE PRESSURE ANALYSIS) output from Abaqus to Matlab (Record key 40)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\40.inp'],[S(1:a(end)-1),'\40.inp'],'f')
```

Run the input file 40.inp with Abaqus

```
!abaqus job=40
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('40.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('40.fil');
```

Obtain the desired output data

```
out = Rec40(Rec)
```

```
out =

    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
    0.0681
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

TOTAL FLUID VOLUME RATIO output from Abaqus to Matlab (Record key 43)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\43.inp'],[S(1:a(end)-1),'\43.inp'],'f')
```

Run the input file 43.inp with Abaqus

```
!abaqus job=43
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('43.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('43.fil');
```

Obtain the desired output data

```
out = Rec43(Rec)
```

```
out =

    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
    0.1097
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

EQUIVALENT PLASTIC STRAIN COMPONENTS output from Abaqus to Matlab (Record key 45)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\45.inp'],[S(1:a(end)-1),'\45.inp'],'f')
```

Run the input file 45.inp with Abaqus

```
!abaqus job=45
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('45.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('45.fil');
```

Obtain the desired output data

```
out = Rec45(Rec)
```

```
out =

  Columns 1 through 7

    1.1128    1.0000         0         0         0         0    0.2982
         0         0         0         0         0         0         0

  Column 8

    1.0000
         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     8
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     2
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

ELEMENT STATUS output from Abaqus to Matlab (Record key 61)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\61.inp'],[S(1:a(end)-1),'\61.inp'],'f')
```

Run the input file 61.inp with Abaqus

```
!abaqus job=61
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('61.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('61.fil');
```

Obtain the desired output data

```
out = Rec61(Rec)
```

```
out =

     1
     1
     1
     1
     1
     1
     1
     1
     1
     1
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

WHOLE ELEMENT VOLUME output from Abaqus to Matlab (Record key 78)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\78.inp'],[S(1:a(end)-1),'\78.inp'],'f')
```

Run the input file 78.inp with Abaqus

```
!abaqus job=78
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('78.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('78.fil');
```

Obtain the desired output data

```
out = Rec78(Rec)
```

```
out =

    1.0000
    1.0000
    1.0000
    1.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     1
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

AVERAGE SHELL SECTION STRESS output from Abaqus to Matlab (Record key 83)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\83.inp'],[S(1:a(end)-1),'\83.inp'],'f')
```

Run the input file 83.inp with Abaqus

```
!abaqus job=83
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('83.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('83.fil');
```

Obtain the desired output data

```
out = Rec83(Rec)
```

```
out =

   1.0e+03 *

   -1.0000   -1.0000   -1.0000    0.0000   -0.0000   -0.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     1
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

THERMAL STRAIN output from Abaqus to Matlab (Record key 88)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\88.inp'],[S(1:a(end)-1),'\88.inp'],'f')
```

Run the input file 88.inp with Abaqus

```
!abaqus job=88
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('88.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('88.fil');
```

Obtain the desired output data

```
out = Rec88(Rec)
```

```
out =

   1.0e-03 *

    1.0000    1.0000         0
    1.0000    1.0000         0
    1.0000    1.0000         0
    1.0000    1.0000         0
         0         0         0
         0         0         0
         0         0         0
         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

LOGARITHMIC STRAIN output from Abaqus to Matlab (Record key 89)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\89.inp'],[S(1:a(end)-1),'\89.inp'],'f')
```

Run the input file 89.inp with Abaqus

```
!abaqus job=89
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('89.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('89.fil');
```

Obtain the desired output data

```
out = Rec89(Rec)
```

```
out =

   -0.0012   -0.0012    0.0041    0.0000   -0.0006   -0.0006
   -0.0012   -0.0012    0.0041   -0.0000    0.0006   -0.0006
   -0.0012   -0.0012    0.0041   -0.0000   -0.0006    0.0006
   -0.0012   -0.0012    0.0041    0.0000    0.0006    0.0006
   -0.0008   -0.0008    0.0033    0.0000   -0.0006   -0.0006
   -0.0008   -0.0008    0.0033   -0.0000    0.0006   -0.0006
   -0.0008   -0.0008    0.0033   -0.0000   -0.0006    0.0006
   -0.0008   -0.0008    0.0033    0.0000    0.0006    0.0006
   -0.0002   -0.0002    0.0006    0.0000   -0.0001   -0.0001
   -0.0002   -0.0002    0.0006   -0.0000    0.0001   -0.0001
   -0.0002   -0.0002    0.0006   -0.0000   -0.0001    0.0001
   -0.0002   -0.0002    0.0006    0.0000    0.0001    0.0001
   -0.0001   -0.0001    0.0005    0.0000   -0.0001   -0.0001
   -0.0001   -0.0001    0.0005   -0.0000    0.0001   -0.0001
   -0.0001   -0.0001    0.0005   -0.0000   -0.0001    0.0001
   -0.0001   -0.0001    0.0005    0.0000    0.0001    0.0001
   -0.0009   -0.0009    0.0033    0.0000   -0.0005   -0.0005
   -0.0009   -0.0009    0.0033   -0.0000    0.0005   -0.0005
   -0.0009   -0.0009    0.0033   -0.0000   -0.0005    0.0005
   -0.0009   -0.0009    0.0033    0.0000    0.0005    0.0005
   -0.0006   -0.0006    0.0027    0.0000   -0.0005   -0.0005
   -0.0006   -0.0006    0.0027   -0.0000    0.0005   -0.0005
   -0.0006   -0.0006    0.0027   -0.0000   -0.0005    0.0005
   -0.0006   -0.0006    0.0027    0.0000    0.0005    0.0005
   -0.0004   -0.0004    0.0013    0.0000   -0.0002   -0.0002
   -0.0004   -0.0004    0.0013   -0.0000    0.0002   -0.0002
   -0.0004   -0.0004    0.0013   -0.0000   -0.0002    0.0002
   -0.0004   -0.0004    0.0013    0.0000    0.0002    0.0002
   -0.0003   -0.0003    0.0010    0.0000   -0.0002   -0.0002
   -0.0003   -0.0003    0.0010   -0.0000    0.0002   -0.0002
   -0.0003   -0.0003    0.0010   -0.0000   -0.0002    0.0002
   -0.0003   -0.0003    0.0010    0.0000    0.0002    0.0002
   -0.0007   -0.0007    0.0026    0.0000   -0.0004   -0.0004
   -0.0007   -0.0007    0.0026   -0.0000    0.0004   -0.0004
   -0.0007   -0.0007    0.0026   -0.0000   -0.0004    0.0004
   -0.0007   -0.0007    0.0026    0.0000    0.0004    0.0004
   -0.0005   -0.0005    0.0022    0.0000   -0.0004   -0.0004
   -0.0005   -0.0005    0.0022   -0.0000    0.0004   -0.0004
   -0.0005   -0.0005    0.0022   -0.0000   -0.0004    0.0004
   -0.0005   -0.0005    0.0022    0.0000    0.0004    0.0004
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

NOMINAL STRAIN output from Abaqus to Matlab (Record key 90)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\90.inp'],[S(1:a(end)-1),'\90.inp'],'f')
```

Run the input file 90.inp with Abaqus

```
!abaqus job=90
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('90.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('90.fil');
```

Obtain the desired output data

```
out = Rec90(Rec)
```

```
out =

   -0.0012   -0.0012    0.0041    0.0000   -0.0006   -0.0006
   -0.0012   -0.0012    0.0041   -0.0000    0.0006   -0.0006
   -0.0012   -0.0012    0.0041   -0.0000   -0.0006    0.0006
   -0.0012   -0.0012    0.0041    0.0000    0.0006    0.0006
   -0.0008   -0.0008    0.0033    0.0000   -0.0006   -0.0006
   -0.0008   -0.0008    0.0033   -0.0000    0.0006   -0.0006
   -0.0008   -0.0008    0.0033   -0.0000   -0.0006    0.0006
   -0.0008   -0.0008    0.0033    0.0000    0.0006    0.0006
   -0.0002   -0.0002    0.0006    0.0000   -0.0001   -0.0001
   -0.0002   -0.0002    0.0006   -0.0000    0.0001   -0.0001
   -0.0002   -0.0002    0.0006   -0.0000   -0.0001    0.0001
   -0.0002   -0.0002    0.0006    0.0000    0.0001    0.0001
   -0.0001   -0.0001    0.0005    0.0000   -0.0001   -0.0001
   -0.0001   -0.0001    0.0005   -0.0000    0.0001   -0.0001
   -0.0001   -0.0001    0.0005   -0.0000   -0.0001    0.0001
   -0.0001   -0.0001    0.0005    0.0000    0.0001    0.0001
   -0.0009   -0.0009    0.0033    0.0000   -0.0005   -0.0005
   -0.0009   -0.0009    0.0033   -0.0000    0.0005   -0.0005
   -0.0009   -0.0009    0.0033   -0.0000   -0.0005    0.0005
   -0.0009   -0.0009    0.0033    0.0000    0.0005    0.0005
   -0.0006   -0.0006    0.0027    0.0000   -0.0005   -0.0005
   -0.0006   -0.0006    0.0027   -0.0000    0.0005   -0.0005
   -0.0006   -0.0006    0.0027   -0.0000   -0.0005    0.0005
   -0.0006   -0.0006    0.0027    0.0000    0.0005    0.0005
   -0.0004   -0.0004    0.0013    0.0000   -0.0002   -0.0002
   -0.0004   -0.0004    0.0013   -0.0000    0.0002   -0.0002
   -0.0004   -0.0004    0.0013   -0.0000   -0.0002    0.0002
   -0.0004   -0.0004    0.0013    0.0000    0.0002    0.0002
   -0.0003   -0.0003    0.0010    0.0000   -0.0002   -0.0002
   -0.0003   -0.0003    0.0010   -0.0000    0.0002   -0.0002
   -0.0003   -0.0003    0.0010   -0.0000   -0.0002    0.0002
   -0.0003   -0.0003    0.0010    0.0000    0.0002    0.0002
   -0.0007   -0.0007    0.0026    0.0000   -0.0004   -0.0004
   -0.0007   -0.0007    0.0026   -0.0000    0.0004   -0.0004
   -0.0007   -0.0007    0.0026   -0.0000   -0.0004    0.0004
   -0.0007   -0.0007    0.0026    0.0000    0.0004    0.0004
   -0.0005   -0.0005    0.0022    0.0000   -0.0004   -0.0004
   -0.0005   -0.0005    0.0022   -0.0000    0.0004   -0.0004
   -0.0005   -0.0005    0.0022   -0.0000   -0.0004    0.0004
   -0.0005   -0.0005    0.0022    0.0000    0.0004    0.0004
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

MECHANICAL STRAIN RATE output from Abaqus to Matlab (Record key 91)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\91.inp'],[S(1:a(end)-1),'\91.inp'],'f')
```

Run the input file 91.inp with Abaqus

```
!abaqus job=91
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('91.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('91.fil');
```

Obtain the desired output data

```
out = Rec91(Rec)
```

```
out =

   -0.0600   -0.0600    0.2000    0.0000    0.0000         0
   -0.0600   -0.0600    0.2000    0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000         0         0
   -0.0600   -0.0600    0.2000         0    0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000         0
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000         0    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000         0    0.0000
   -0.0600   -0.0600    0.2000   -0.0000         0    0.0000
   -0.0600   -0.0600    0.2000    0.0000   -0.0000         0
   -0.0600   -0.0600    0.2000    0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000         0   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000         0    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000         0   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000   -0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000   -0.0000   -0.0000
   -0.0600   -0.0600    0.2000         0    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000   -0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0600   -0.0600    0.2000   -0.0000         0    0.0000
   -0.0600   -0.0600    0.2000    0.0000    0.0000    0.0000
   -0.0619   -0.0619    0.2000    0.0000    0.0000   -0.0000
   -0.0619   -0.0619    0.2000    0.0000    0.0000   -0.0000
   -0.0619   -0.0619    0.2000    0.0000   -0.0000   -0.0000
   -0.0619   -0.0619    0.2000    0.0000   -0.0000   -0.0000
   -0.0619   -0.0619    0.2000   -0.0000    0.0000   -0.0000
   -0.0619   -0.0619    0.2000   -0.0000    0.0000    0.0000
   -0.0619   -0.0619    0.2000    0.0000   -0.0000   -0.0000
   -0.0619   -0.0619    0.2000   -0.0000   -0.0000   -0.0000
   -0.0770   -0.0770    0.2000   -0.0000   -0.0000    0.0000
   -0.0770   -0.0770    0.2000    0.0000   -0.0000    0.0000
   -0.0770   -0.0770    0.2000   -0.0000    0.0000    0.0000
   -0.0770   -0.0770    0.2000   -0.0000    0.0000    0.0000
   -0.0770   -0.0770    0.2000    0.0000   -0.0000    0.0000
   -0.0770   -0.0770    0.2000    0.0000         0    0.0000
   -0.0770   -0.0770    0.2000   -0.0000    0.0000    0.0000
   -0.0770   -0.0770    0.2000    0.0000    0.0000    0.0000
   -0.0890   -0.0890    0.2000   -0.0000   -0.0000    0.0000
```

-0.0890	-0.0890	0.2000	-0.0000	-0.0000	-0.0000
-0.0890	-0.0890	0.2000	0.0000	0	0
-0.0890	-0.0890	0.2000	-0.0000	-0.0000	-0.0000
-0.0890	-0.0890	0.2000	0.0000	-0.0000	0
-0.0890	-0.0890	0.2000	-0.0000	-0.0000	-0.0000
-0.0890	-0.0890	0.2000	0.0000	0	0.0000
-0.0890	-0.0890	0.2000	-0.0000	-0.0000	-0.0000
-0.0932	-0.0932	0.2000	0.0000	0.0000	0
-0.0932	-0.0932	0.2000	0.0000	0.0000	0.0000
-0.0932	-0.0932	0.2000	-0.0000	-0.0000	0
-0.0932	-0.0932	0.2000	0.0000	-0.0000	0.0000
-0.0932	-0.0932	0.2000	-0.0000	0.0000	0.0000
-0.0932	-0.0932	0.2000	-0.0000	0.0000	0.0000
-0.0932	-0.0932	0.2000	-0.0000	0	0.0000
-0.0932	-0.0932	0.2000	-0.0000	-0.0000	0.0000
-0.0944	-0.0944	0.2000	-0.0000	0.0000	0
-0.0944	-0.0944	0.2000	-0.0000	0.0000	0.0000
-0.0944	-0.0944	0.2000	0.0000	0.0000	-0.0000
-0.0944	-0.0944	0.2000	-0.0000	0.0000	0.0000
-0.0944	-0.0944	0.2000	0.0000	0.0000	-0.0000
-0.0944	-0.0944	0.2000	-0.0000	0.0000	0
-0.0944	-0.0944	0.2000	0.0000	0	0
-0.0944	-0.0944	0.2000	0.0000	0	0.0000
-0.0947	-0.0947	0.2000	-0.0000	-0.0000	0.0000
-0.0947	-0.0947	0.2000	0.0000	-0.0000	-0.0000
-0.0947	-0.0947	0.2000	-0.0000	0.0000	0.0000
-0.0947	-0.0947	0.2000	-0.0000	0.0000	-0.0000
-0.0947	-0.0947	0.2000	-0.0000	-0.0000	0
-0.0947	-0.0947	0.2000	0.0000	-0.0000	-0.0000
-0.0947	-0.0947	0.2000	-0.0000	0.0000	0.0000
-0.0947	-0.0947	0.2000	-0.0000	0.0000	-0.0000
-0.0953	-0.0953	0.2000	-0.0000	0.0000	-0.0000
-0.0953	-0.0953	0.2000	-0.0000	0.0000	0.0000
-0.0953	-0.0953	0.2000	0.0000	0	-0.0000
-0.0953	-0.0953	0.2000	-0.0000	-0.0000	0.0000
-0.0953	-0.0953	0.2000	0.0000	0.0000	0
-0.0953	-0.0953	0.2000	-0.0000	0.0000	0.0000
-0.0953	-0.0953	0.2000	0.0000	0	-0.0000
-0.0953	-0.0953	0.2000	0.0000	0	0.0000
-0.0987	-0.0987	0.2000	0.0000	-0.0000	0.0000
-0.0987	-0.0987	0.2000	0.0000	-0.0000	0.0000
-0.0987	-0.0987	0.2000	-0.0000	-0.0000	0.0000
-0.0987	-0.0987	0.2000	0.0000	-0.0000	-0.0000
-0.0987	-0.0987	0.2000	-0.0000	-0.0000	0.0000
-0.0987	-0.0987	0.2000	0.0000	-0.0000	-0.0000
-0.0987	-0.0987	0.2000	-0.0000	-0.0000	0.0000
-0.0987	-0.0987	0.2000	-0.0000	-0.0000	-0.0000
-0.0996	-0.0996	0.2000	-0.0000	-0.0000	0.0000
-0.0996	-0.0996	0.2000	-0.0000	-0.0000	0.0000
-0.0996	-0.0996	0.2000	-0.0000	0.0000	0.0000
-0.0996	-0.0996	0.2000	-0.0000	0.0000	0.0000
-0.0996	-0.0996	0.2000	0	-0.0000	0.0000
-0.0996	-0.0996	0.2000	-0.0000	-0.0000	0.0000
-0.0996	-0.0996	0.2000	0.0000	0.0000	0.0000
-0.0996	-0.0996	0.2000	0.0000	0.0000	0.0000
-0.0999	-0.0999	0.2000	0.0000	0.0000	0.0000
-0.0999	-0.0999	0.2000	0.0000	0.0000	-0.0000
-0.0999	-0.0999	0.2000	0.0000	0	0.0000
-0.0999	-0.0999	0.2000	0.0000	0.0000	-0.0000
-0.0999	-0.0999	0.2000	-0.0000	0.0000	0.0000
-0.0999	-0.0999	0.2000	-0.0000	0.0000	-0.0000

```
   -0.0999   -0.0999    0.2000   -0.0000         0    0.0000
   -0.0999   -0.0999    0.2000   -0.0000         0   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000         0   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000         0   -0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000         0
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000         0   -0.0000         0
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000         0
   -0.1000   -0.1000    0.2000         0   -0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000    0.0000   -0.0000
   -0.1000   -0.1000    0.2000    0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000    0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000         0
   -0.1000   -0.1000    0.2000    0.0000    0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000         0    0.0000   -0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000         0
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000         0    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000   -0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000   -0.0000         0
   -0.1000   -0.1000    0.2000   -0.0000    0.0000    0.0000
   -0.1000   -0.1000    0.2000    0.0000    0.0000    0.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

   160
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

PORE FLUID EFFECTIVE VELOCITY VECTOR output from Abaqus to Matlab (Record key 97)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\97.inp'],[S(1:a(end)-1),'\97.inp'],'f')
```

Run the input file 97.inp with Abaqus

```
!abaqus job=97
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('97.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('97.fil');
```

Obtain the desired output data

```
out = Rec97(Rec)
```

```
out =

    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000    0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000    0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000    0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000    0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000    0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000         0   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000   -0.0000   -5.0000
    5.0000    0.0000   -5.0000
    5.0000   -0.0000   -5.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000    0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000    0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000    0.0000  -10.0000
   10.0000         0  -10.0000
```

```
   10.0000    0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000    0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000         0  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000   -0.0000  -10.0000
   10.0000    0.0000  -10.0000
   10.0000   -0.0000  -10.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    78
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

DISPLACEMENT output from Abaqus to Matlab (Record key 101)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\101.inp'],[S(1:a(end)-1),'\101.inp'],'f')
```

Run the input file 101.inp with Abaqus

```
!abaqus job=101
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('101.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('101.fil');
```

Obtain the desired output data

```
out = Rec101(Rec)
```

```
out =

    1.0000    2.7310  -18.3425         0         0
    2.0000   -4.2518  -20.9370         0         0
    3.0000    3.0339   -9.6878         0         0
    4.0000   -3.4350  -11.9710         0         0
    5.0000    0.0000   -0.0000         0         0
    6.0000   -0.0000   -0.0000         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     5
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

VELOCITY output from Abaqus to Matlab (Record key 102)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\102.inp'],[S(1:a(end)-1),'\102.inp'],'f')
```

Run the input file 102.inp with Abaqus

```
!abaqus job=102
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('102.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('102.fil');
```

Obtain the desired output data

```
out = Rec102(Rec)
```

```
out =

    1.0000    5.4497  -36.6018         0         0
    2.0000   -8.4843  -42.0057         0         0
    3.0000    6.0540  -19.3317         0         0
    4.0000   -6.8545  -24.1145         0         0
    5.0000         0         0         0         0
    6.0000         0         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     5
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Published with MATLAB® R2015a

ACCELERATION output from Abaqus to Matlab (Record key 103)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\103.inp'],[S(1:a(end)-1),'\103.inp'],'f')
```

Run the input file 103.inp with Abaqus

```
!abaqus job=103
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('103.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('103.fil');
```

Obtain the desired output data

```
out = Rec103(Rec)
```

```
out =

    1.0000    9.9085  -66.5487         0         0
    2.0000  -15.4259    5.4441         0         0
    3.0000   11.0072  -35.1486         0         0
    4.0000  -12.4627   37.9736         0         0
    5.0000         0         0         0         0
    6.0000         0         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     5
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

REACTION FORCE output from Abaqus to Matlab (Record key 104)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\104.inp'],[S(1:a(end)-1),'\104.inp'],'f')
```

Run the input file 104.inp with Abaqus

```
!abaqus job=104
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('104.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('104.fil');
```

Obtain the desired output data

```
out = Rec104(Rec)
```

```
out =

    1.0000         0         0         0         0
    2.0000         0         0         0         0
    3.0000         0         0         0         0
    4.0000         0         0         0         0
    5.0000 -159.7002   79.6397         0         0
    6.0000  152.7273   62.0807         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     5
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

ELECTRICAL POTENTIAL output from Abaqus to Matlab (Record key 105)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\105.inp'],[S(1:a(end)-1),'\105.inp'],'f')
```

Run the input file 105.inp with Abaqus

```
!abaqus job=105
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('105.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('105.fil');
```

Obtain the desired output data

```
out = Rec105(Rec)
```

```
out =

   1.0e+06 *

    0.0000   -0.0000
    0.0000   -0.0000
    0.0000   -0.0000
    0.0000   -0.0000
    0.0000    1.0000
    0.0000    1.0000
    0.0000    1.0000
    0.0000    1.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:

G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

POINT LOADS, MOMENTS, FLUXES output from Abaqus to Matlab (Record key 106)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\106.inp'],[S(1:a(end)-1),'\106.inp'],'f')
```

Run the input file 106.inp with Abaqus

```
!abaqus job=106
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('106.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('106.fil');
```

Obtain the desired output data

```
out = Rec106(Rec)
```

```
out =

     1     0     0     0     0
     2     0  -100     0     0
     3     0     0     0     0
     4     0  -100     0     0
     5     0     0     0     0
     6     0     0     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     5
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

COORDINATE output from Abaqus to Matlab (Record key 107)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\107.inp'],[S(1:a(end)-1),'\107.inp'],'f')
```

Run the input file 107.inp with Abaqus

```
!abaqus job=107
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('107.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('107.fil');
```

Obtain the desired output data

```
out = Rec107(Rec)
```

```
out =

     1   720   360
     2   720     0
     3   360   360
     4   360     0
     5     0   360
     6     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

PORE OR ACOUSTIC PRESSURE output from Abaqus to Matlab (Record key 108)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\108.inp'],[S(1:a(end)-1),'\108.inp'],'f')
```

Run the input file 108.inp with Abaqus

```
!abaqus job=108
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('108.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('108.fil');
```

Obtain the desired output data

```
out = Rec108(Rec)
```

```
out =

    1.0000   13.3636
    3.0000   13.3636
    7.0000         0
    9.0000         0
    1.0000         0
    3.0000   12.6000
    7.0000         0
    9.0000   12.6000
    1.0000         0
    3.0000         0
    7.0000   13.3636
    9.0000   13.3636
    1.0000   12.5902
    3.0000         0
    7.0000   12.6021
    9.0000         0
    1.0000  -15.0000
    3.0000  -15.0000
    7.0000         0
    9.0000         0
    1.0000         0
    3.0000   -9.0000
    7.0000         0
    9.0000   -9.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    24
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

REACTIVE FLUID VOLUME FLUX output from Abaqus to Matlab (Record key 109)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\109.inp'],[S(1:a(end)-1),'\109.inp'],'f')
```

Run the input file 109.inp with Abaqus

```
!abaqus job=109
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('109.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('109.fil');
```

Obtain the desired output data

```
out = Rec109(Rec)
```

```
out =

     1.0000    0.1225
     2.0000   -0.1225
     3.0000   -0.1225
     4.0000    0.1225
     5.0000    0.1225
     6.0000   -0.1225
     7.0000   -0.1225
     8.0000    0.1225
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
(submitted)
```

Published with MATLAB® R2015a

REACTIVE FLUID TOTAL VOLUME output from Abaqus to Matlab (Record key 110)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\110.inp'],[S(1:a(end)-1),'\110.inp'],'f')
```

Run the input file 110.inp with Abaqus

```
!abaqus job=110
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('110.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('110.fil');
```

Obtain the desired output data

```
out = Rec110(Rec)
```

```
out =

           1           0
           3           0
           5           0
           7           0
           9           0
          11           0
          15           0
          19           0
          21           0
          23           0
          25           0
          27           0
          29           0
        1201           0
        1203           0
        1205           0
        1207           0
        1209           0
        1211           0
        1215           0
        1219           0
        1221           0
        1223           0
        1225           0
        1227           0
        1229           0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    26
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

ELECTRICAL REACTION CHARGE output from Abaqus to Matlab (Record key 119)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\119.inp'],[S(1:a(end)-1),'\119.inp'],'f')
```

Run the input file 119.inp with Abaqus

```
!abaqus job=119
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('119.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('119.fil');
```

Obtain the desired output data

```
out = Rec119(Rec)
```

```
out =

   1.0e+03 *

    0.0010   -1.0000
    0.0020   -1.0000
    0.0030   -1.0000
    0.0040   -1.0000
    0.0050         0
    0.0060         0
    0.0070         0
    0.0080         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

CONCENTRATED ELECTRICAL NODAL CHARGE output from Abaqus to Matlab (Record key 120)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\120.inp'],[S(1:a(end)-1),'\120.inp'],'f')
```

Run the input file 120.inp with Abaqus

```
!abaqus job=120
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('120.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('120.fil');
```

Obtain the desired output data

```
out = Rec120(Rec)
```

```
out =

           1       -2000
           2       -2000
           3       -2000
           4       -2000
           5       -1000
           6       -1000
           7       -1000
           8       -1000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

FLUID CAVITY PRESSURE output from Abaqus to Matlab (Record key 136)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\136.inp'],[S(1:a(end)-1),'\136.inp'],'f')
```

Run the input file 136.inp with Abaqus

```
!abaqus job=136
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('136.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('136.fil');
```

Obtain the desired output data

```
out = Rec136(Rec)
```

```
out =

    1.0000   99.4886
    1.0000  179.6653
    1.0000  283.8729
    1.0000  376.9086
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

FLUID CAVITY VOLUME output from Abaqus to Matlab (Record key 137)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\137.inp'],[S(1:a(end)-1),'\137.inp'],'f')
```

Run the input file 137.inp with Abaqus

```
!abaqus job=137
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('137.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('137.fil');
```

Obtain the desired output data

```
out = Rec137(Rec)
```

```
out =

    1.0000    0.9999
    1.0000    0.9999
    1.0000    0.9999
    1.0000    1.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

ELECTRICAL REACTION CURRENT output from Abaqus to Matlab (Record key 138)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\138.inp'],[S(1:a(end)-1),'\138.inp'],'f')
```

Run the input file 138.inp with Abaqus

```
!abaqus job=138
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('138.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('138.fil');
```

Obtain the desired output data

```
out = Rec138(Rec)
```

```
out =

   1.0e+08 *

    9.0000   -0.0007
    9.0000    0.0007
    9.0000   -0.0007
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     3
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

CONCENTRATED ELECTRICAL NODAL CURRENT output from Abaqus to Matlab (Record key 139)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\139.inp'],[S(1:a(end)-1),'\139.inp'],'f')
```

Run the input file 139.inp with Abaqus

```
!abaqus job=139
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('139.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('139.fil');
```

Obtain the desired output data

```
out = Rec139(Rec)
```

```
out =

   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000009           0
   900000010           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000009           0
   900000010       65800
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    20
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

VISCOUS FORCES DUE TO STATIC STABILIZATION output from Abaqus to Matlab (Record key 145)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\145.inp'],[S(1:a(end)-1),'\145.inp'],'f')
```

Run the input file 145.inp with Abaqus

```
!abaqus job=145
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('145.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('145.fil');
```

Obtain the desired output data

```
out = Rec145(Rec)
```

```
out =

    2.0000   -0.6707         0
    2.0000   -0.5820         0
    2.0000   -0.5212         0
    2.0000   -0.4665         0
    2.0000   -0.4315         0
    2.0000   -0.3991         0
    2.0000   -0.3690         0
    2.0000   -0.3483         0
    2.0000   -0.3288         0
    2.0000   -0.3201         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

TOTAL FORCE output from Abaqus to Matlab (Record key 146)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\146.inp'],[S(1:a(end)-1),'\146.inp'],'f')
```

Run the input file 146.inp with Abaqus

```
!abaqus job=146
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('146.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('146.fil');
```

Obtain the desired output data

```
out = Rec146(Rec)
```

```
out =

   1.0e+10 *

    0.0000    0.0000   -0.6557
    0.0000         0   -0.6557
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    0.6557
    0.0000         0    0.6557
    0.0000    1.3115         0
    0.0000    0.0000   -0.7584
    0.0000         0   -0.7584
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    0.7584
    0.0000         0    0.7584
    0.0000    1.5167         0
    0.0000    0.0000   -0.9090
    0.0000         0   -0.9090
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    0.9090
    0.0000         0    0.9090
    0.0000    1.8179         0
    0.0000    0.0000   -1.1277
    0.0000         0   -1.1277
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    1.1277
    0.0000         0    1.1277
    0.0000    2.2554         0
    0.0000    0.0000   -1.4405
    0.0000         0   -1.4405
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    1.4405
    0.0000         0    1.4405
    0.0000    2.8811         0
    0.0000    0.0000   -1.5188
    0.0000         0   -1.5188
    0.0000    0.0000         0
    0.0000         0         0
    0.0000    0.0000    1.5188
    0.0000         0    1.5188
    0.0000    3.0377         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =
```

```
    3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    42
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

TEMPERATURE output from Abaqus to Matlab (Record key 201)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\201.inp'],[S(1:a(end)-1),'\201.inp'],'f')
```

Run the input file 201.inp with Abaqus

```
!abaqus job=201
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('201.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('201.fil');
```

Obtain the desired output data

```
out = Rec201(Rec)
```

```
out =

   1.0e+08 *

    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000         0
    9.0000         0
    9.0000    0.0000
    9.0000         0
    9.0000         0
    9.0000         0
    9.0000         0
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =
```

```
    30
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

RESIDUAL FLUX output from Abaqus to Matlab (Record key 204)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\204.inp'],[S(1:a(end)-1),'\204.inp'],'f')
```

Run the input file 204.inp with Abaqus

```
!abaqus job=204
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('204.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('204.fil');
```

Obtain the desired output data

```
out = Rec204(Rec)
```

```
out =

   1.0e+08 *

    9.0000   -0.0001
    9.0000   -0.0000
    9.0000    0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000    0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0001
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     9
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

CONCENTRATED FLUX output from Abaqus to Matlab (Record key 206)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\206.inp'],[S(1:a(end)-1),'\206.inp'],'f')
```

Run the input file 206.inp with Abaqus

```
!abaqus job=206
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('206.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('206.fil');
```

Obtain the desired output data

```
out = Rec206(Rec)
```

```
out =

   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
   900000002           0
   900000003           0
   900000004           0
   900000005           0
   900000006           0
   900000007           0
   900000008           0
   900000001           0
```

900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0

900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0

900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0

900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0

900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0

900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0

900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0

900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	0
900000002	0
900000003	0
900000004	0
900000005	0
900000006	0
900000007	0
900000008	0
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3

900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3

900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3

900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3

900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3
900000008	3
900000001	3
900000002	3
900000003	3
900000004	3
900000005	3
900000006	3
900000007	3

```
   900000008           3
   900000001           3
   900000002           3
   900000003           3
   900000004           3
   900000005           3
   900000006           3
   900000007           3
   900000008           3
   900000001           3
   900000002           3
   900000003           3
   900000004           3
   900000005           3
   900000006           3
   900000007           3
   900000008           3
   900000001           3
   900000002           3
   900000003           3
   900000004           3
   900000005           3
   900000006           3
   900000007           3
   900000008           3
   900000001           3
   900000002           3
   900000003           3
   900000004           3
   900000005           3
   900000006           3
   900000007           3
   900000008           3
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

   816
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

INTERNAL FLUX output from Abaqus to Matlab (Record key 214)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\214.inp'],[S(1:a(end)-1),'\214.inp'],'f')
```

Run the input file 214.inp with Abaqus

```
!abaqus job=214
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('214.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('214.fil');
```

Obtain the desired output data

```
out = Rec214(Rec)
```

```
out =

   1.0e+08 *

    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
    9.0000   -0.0000
```

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

```
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
9.0000   -0.0000
```

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000

9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	-0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000

9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000

9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000

9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000

9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000
9.0000	0.0000

```
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
    9.0000    0.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =
```

```
816
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

NORMALIZED CONCENTRATION (MASS DIFFUSION ANALYSIS) output from Abaqus to Matlab (Record key 221)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\221.inp'],[S(1:a(end)-1),'\221.inp'],'f')
```

Run the input file 221.inp with Abaqus

```
!abaqus job=221
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('221.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('221.fil');
```

Obtain the desired output data

```
out = Rec221(Rec)
```

```
out =

   1.0e+04 *

    0.0001         0
    0.0002         0
    0.0003    4.8740
    0.0004    1.6247
    0.0001    2.7851
    0.0002    2.7851
    0.0003    0.0000
    0.0004    0.0000
    0.0001         0
    0.0002    5.5703
    0.0003    5.5703
    0.0004         0
    0.0001         0
    0.0002         0
    0.0003    2.7851
    0.0004    2.7851
    0.0001    5.5703
    0.0002         0
    0.0003         0
    0.0004    5.5703
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    20
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

MOTIONS (IN CAVITY RADIATION ANALYSIS) output from Abaqus to Matlab (Record key 237)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\237.inp'],[S(1:a(end)-1),'\237.inp'],'f')
```

Run the input file 237.inp with Abaqus

```
!abaqus job=237
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('237.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('237.fil');
```

Obtain the desired output data

```
out = Rec237(Rec)
```

```
out =

     1     0     0
     2     0     0
     3     0     0
     4     0     0
     5     0     0
     6     0     0
     7     0     0
     8     0     0
     9     0     0
    10     0     0
    11     0     0
    12     0     0
    13     0     0
    14     0     0
    15     0     0
    16     0     0
    17     0     0
    18     0     0
    21     0     0
    22     0     0
    23     0     0
    24     0     0
    25     0     0
    26     0     0
    27     0     0
    28     0     0
    29     0     0
    30     0     0
    31     0     0
    32     0     0
    33     0     0
    34     0     0
    35     0     0
    36     0     0
    37     0     0
    38     0     0
    41     0     0
    42     0     0
    43     0     0
    44     0     0
    45     0     0
    46     0     0
    47     0     0
    48     0     0
    49     0     0
    50     0     0
    51     0     0
    52     0     0
    53     0     0
    54     0     0
    55     0     0
    56     0     0
    57     0     0
    58     0     0
    61     0     0
    62     0     0
```

63	0	0
64	0	0
65	0	0
66	0	0
67	0	0
68	0	0
69	0	0
70	0	0
71	0	0
72	0	0
73	0	0
74	0	0
75	0	0
76	0	0
77	0	0
78	0	0
81	0	0
82	0	0
83	0	0
84	0	0
85	0	0
86	0	0
87	0	0
88	0	0
89	0	0
90	0	0
91	0	0
92	0	0
93	0	0
94	0	0
95	0	0
96	0	0
97	0	0
98	0	0
101	0	0
102	0	0
103	0	0
104	0	0
105	0	0
106	0	0
107	0	0
108	0	0
109	0	0
110	0	0
111	0	0
112	0	0
113	0	0
114	0	0
115	0	0
116	0	0
117	0	0
118	0	0
121	0	0
122	0	0
123	0	0
124	0	0
125	0	0
126	0	0
127	0	0
128	0	0
129	0	0

130	0	0
131	0	0
132	0	0
133	0	0
134	0	0
135	0	0
136	0	0
137	0	0
138	0	0
141	0	0
142	0	0
143	0	0
144	0	0
145	0	0
146	0	0
147	0	0
148	0	0
149	0	0
150	0	0
151	0	0
152	0	0
153	0	0
154	0	0
155	0	0
156	0	0
157	0	0
158	0	0
161	0	0
162	0	0
163	0	0
164	0	0
165	0	0
166	0	0
167	0	0
168	0	0
169	0	0
170	0	0
171	0	0
172	0	0
173	0	0
174	0	0
175	0	0
176	0	0
177	0	0
178	0	0
181	0	0
182	0	0
183	0	0
184	0	0
185	0	0
186	0	0
187	0	0
188	0	0
189	0	0
190	0	0
191	0	0
192	0	0
193	0	0
194	0	0
195	0	0

196	0	0
197	0	0
198	0	0
201	0	0
202	0	0
203	0	0
204	0	0
205	0	0
206	0	0
207	0	0
208	0	0
209	0	0
210	0	0
211	0	0
212	0	0
213	0	0
214	0	0
215	0	0
216	0	0
217	0	0
218	0	0
221	0	0
222	0	0
223	0	0
224	0	0
225	0	0
226	0	0
227	0	0
228	0	0
229	0	0
230	0	0
231	0	0
232	0	0
233	0	0
234	0	0
235	0	0
236	0	0
237	0	0
238	0	0
241	0	0
242	0	0
243	0	0
244	0	0
245	0	0
246	0	0
247	0	0
248	0	0
249	0	0
250	0	0
251	0	0
252	0	0
253	0	0
254	0	0
255	0	0
256	0	0
257	0	0
258	0	0
261	0	0
262	0	0
263	0	0
264	0	0

265	0	0
266	0	0
267	0	0
268	0	0
269	0	0
270	0	0
271	0	0
272	0	0
273	0	0
274	0	0
275	0	0
276	0	0
277	0	0
278	0	0
281	0	0
282	0	0
283	0	0
284	0	0
285	0	0
286	0	0
287	0	0
288	0	0
289	0	0
290	0	0
291	0	0
292	0	0
293	0	0
294	0	0
295	0	0
296	0	0
297	0	0
298	0	0
301	0	0
302	0	0
303	0	0
304	0	0
305	0	0
306	0	0
307	0	0
308	0	0
309	0	0
310	0	0
311	0	0
312	0	0
313	0	0
314	0	0
315	0	0
316	0	0
317	0	0
318	0	0
321	0	0
322	0	0
323	0	0
324	0	0
325	0	0
326	0	0
327	0	0
328	0	0
329	0	0
330	0	0

331	0	0
332	0	0
333	0	0
334	0	0
335	0	0
336	0	0
337	0	0
338	0	0
341	0	0
342	0	0
343	0	0
344	0	0
345	0	0
346	0	0
347	0	0
348	0	0
349	0	0
350	0	0
351	0	0
352	0	0
353	0	0
354	0	0
355	0	0
356	0	0
357	0	0
358	0	0
361	0	0
362	0	0
363	0	0
364	0	0
365	0	0
366	0	0
367	0	0
368	0	0
369	0	0
370	0	0
371	0	0
372	0	0
373	0	0
374	0	0
375	0	0
376	0	0
377	0	0
378	0	0
381	0	0
382	0	0
383	0	0
384	0	0
385	0	0
386	0	0
387	0	0
388	0	0
389	0	0
390	0	0
391	0	0
392	0	0
393	0	0
394	0	0
395	0	0
396	0	0
397	0	0

```
    398     0     0
    401     0     0
    402     0     0
    403     0     0
    404     0     0
    405     0     0
    406     0     0
    407     0     0
    408     0     0
    409     0     0
    410     0     0
    411     0     0
    412     0     0
    413     0     0
    414     0     0
    415     0     0
    416     0     0
    417     0     0
    418     0     0
    421     0     0
    422     0     0
    423     0     0
    424     0     0
    425     0     0
    426     0     0
    427     0     0
    428     0     0
    429     0     0
    430     0     0
    431     0     0
    432     0     0
    433     0     0
    434     0     0
    435     0     0
    436     0     0
    437     0     0
    438     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

   396
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

PRINCIPAL STRESSES output from Abaqus to Matlab (Record key 401)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\401.inp'],[S(1:a(end)-1),'\401.inp'],'f')
```

Run the input file 401.inp with Abaqus

```
!abaqus job=401
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('401.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('401.fil');
```

Obtain the desired output data

```
out = Rec401(Rec)
```

```
out =

   1.0e+05 *

    0.1202    0.3865    1.1682
   -0.1967    0.0403    0.3310
   -1.0819   -0.4451   -0.4019
   -0.2044    0.0982    0.5318
   -0.2064   -0.0009    0.2034
   -0.8011   -0.2392    0.0038
   -0.0740    0.0808    0.3433
    0.0408    0.2288    0.7220
   -0.2700    0.2377    1.0622
    0.2404    0.7731    2.3365
   -0.3933    0.0806    0.6620
   -2.1637   -0.8903   -0.8039
   -0.4087    0.1964    1.0635
   -0.4127   -0.0018    0.4069
   -1.6021   -0.4784    0.0076
   -0.1480    0.1616    0.6866
    0.0816    0.4577    1.4440
   -0.5400    0.4753    2.1245
    0.3606    1.1596    3.5047
   -0.5900    0.1209    0.9930
   -3.2456   -1.3354   -1.2058
   -0.6131    0.2947    1.5953
   -0.6191   -0.0026    0.6103
   -2.4032   -0.7175    0.0114
   -0.2220    0.2424    1.0300
    0.1223    0.6865    2.1660
   -0.8100    0.7130    3.1867
    0.4809    1.5461    4.6729
   -0.7866    0.1612    1.3239
   -4.3274   -1.7805   -1.6077
   -0.8175    0.3929    2.1270
   -0.8254   -0.0035    0.8137
   -3.2042   -0.9567    0.0152
   -0.2960    0.3232    1.3733
    0.1631    0.9153    2.8880
   -1.0800    0.9507    4.2490
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    36
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
________________________________________________________________________
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

PRINCIPAL VALUES OF BACKSTRESS TENSOR FOR KINEMATIC HARDENING PLASTICITY output from Abaqus to Matlab (Record key 402)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\402.inp'],[S(1:a(end)-1),'\402.inp'],'f')
```

Run the input file 402.inp with Abaqus

```
!abaqus job=402
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('402.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('402.fil');
```

Obtain the desired output data

```
out = Rec402(Rec)
```

```
out =

   -0.0000   18.4964
   -0.0000   18.4964
   -0.0000   18.4964
   -0.0000   18.4964
   -0.0000   31.9066
   -0.0000   31.9066
   -0.0000   31.9066
   -0.0000   31.9066
   -0.0000   39.5186
   -0.0000   39.5186
   -0.0000   39.5186
   -0.0000   39.5186
   -0.0000   43.5951
   -0.0000   43.5951
   -0.0000   43.5951
   -0.0000   43.5951
   -0.0000   45.7018
   -0.0000   45.7018
   -0.0000   45.7018
   -0.0000   45.7018
   -0.0000   46.7692
   -0.0000   46.7692
   -0.0000   46.7692
   -0.0000   46.7692
   -0.0000   47.3042
   -0.0000   47.3042
   -0.0000   47.3042
   -0.0000   47.3042
   -0.0000   47.5704
   -0.0000   47.5704
   -0.0000   47.5704
   -0.0000   47.5704
   -0.0000   47.7029
   -0.0000   47.7029
   -0.0000   47.7029
   -0.0000   47.7029
   -0.0000   47.7690
   -0.0000   47.7690
   -0.0000   47.7690
   -0.0000   47.7690
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```



```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

PRINCIPAL STRAINS output from Abaqus to Matlab (Record key 403)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\403.inp'],[S(1:a(end)-1),'\403.inp'],'f')
```

Run the input file 403.inp with Abaqus

```
!abaqus job=403
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('403.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('403.fil');
```

Obtain the desired output data

```
out = Rec403(Rec)
```

```
out =

   -0.0004    0.0010
   -0.0004    0.0010
   -0.0004    0.0010
   -0.0004    0.0010
   -0.0009    0.0020
   -0.0009    0.0020
   -0.0009    0.0020
   -0.0009    0.0020
   -0.0014    0.0030
   -0.0014    0.0030
   -0.0014    0.0030
   -0.0014    0.0030
   -0.0019    0.0040
   -0.0019    0.0040
   -0.0019    0.0040
   -0.0019    0.0040
   -0.0024    0.0050
   -0.0024    0.0050
   -0.0024    0.0050
   -0.0024    0.0050
   -0.0029    0.0060
   -0.0029    0.0060
   -0.0029    0.0060
   -0.0029    0.0060
   -0.0034    0.0070
   -0.0034    0.0070
   -0.0034    0.0070
   -0.0034    0.0070
   -0.0039    0.0080
   -0.0039    0.0080
   -0.0039    0.0080
   -0.0039    0.0080
   -0.0044    0.0090
   -0.0044    0.0090
   -0.0044    0.0090
   -0.0044    0.0090
   -0.0049    0.0100
   -0.0049    0.0100
   -0.0049    0.0100
   -0.0049    0.0100
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

PRINCIPAL NOMINAL STRAINS output from Abaqus to Matlab (Record key 404)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\404.inp'],[S(1:a(end)-1),'\404.inp'],'f')
```

Run the input file 404.inp with Abaqus

```
!abaqus job=404
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('404.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('404.fil');
```

Obtain the desired output data

```
out = Rec404(Rec)
```

```
out =

     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```



```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

PRINCIPAL LOGARITHMIC STRAINS output from Abaqus to Matlab (Record key 405)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\405.inp'],[S(1:a(end)-1),'\405.inp'],'f')
```

Run the input file 405.inp with Abaqus

```
!abaqus job=405
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('405.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('405.fil');
```

Obtain the desired output data

```
out = Rec405(Rec)
```

```
out =

     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    40
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
```



```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```


Published with MATLAB® R2015a

PRINCIPAL MECHANICAL STRAIN RATES output from Abaqus to Matlab (Record key 406)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\406.inp'],[S(1:a(end)-1),'\406.inp'],'f')
```

Run the input file 406.inp with Abaqus

```
!abaqus job=406
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('406.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('406.fil');
```

Obtain the desired output data

```
out = Rec406(Rec)
```

```
out =

   -0.0115   -0.0115    0.0399
   -0.0132   -0.0132    0.0398
   -0.0101   -0.0101    0.0396
   -0.0137   -0.0137    0.0394
   -0.0102   -0.0102    0.0393
   -0.0129   -0.0129    0.0391
   -0.0112   -0.0112    0.0390
   -0.0116   -0.0116    0.0388
   -0.0121   -0.0121    0.0387
   -0.0107   -0.0107    0.0385
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

PRINCIPAL VALUES OF DEFORMATION GRADIENT output from Abaqus to Matlab (Record key 407)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\407.inp'],[S(1:a(end)-1),'\407.inp'],'f')
```

Run the input file 407.inp with Abaqus

```
!abaqus job=407
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('407.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('407.fil');
```

Obtain the desired output data

```
out = Rec407(Rec)
```

```
out =

    0.7500    1.0000    1.0000
    0.5000    1.0000    1.0000
    0.2500    1.0000    1.0000
    0.2000    1.0000    1.0000
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     4
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```


If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

PRINCIPAL ELASTIC STRAINS output from Abaqus to Matlab (Record key 408)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\408.inp'],[S(1:a(end)-1),'\408.inp'],'f')
```

Run the input file 408.inp with Abaqus

```
!abaqus job=408
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('408.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('408.fil');
```

Obtain the desired output data

```
out = Rec408(Rec)
```

```
out =

   -0.0021   -0.0021    0.0112
   -0.4546   -0.4546    1.2145
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     2
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

PRINCIPAL INELASTIC STRAINS output from Abaqus to Matlab (Record key 409)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\409.inp'],[S(1:a(end)-1),'\409.inp'],'f')
```

Run the input file 409.inp with Abaqus

```
!abaqus job=409
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('409.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('409.fil');
```

Obtain the desired output data

```
out = Rec409(Rec)
```

```
out =

   -0.4525   -0.4525    1.2033
         0         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     2
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

PRINCIPAL THERMAL STRAINS output from Abaqus to Matlab (Record key 410)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\410.inp'],[S(1:a(end)-1),'\410.inp'],'f')
```

Run the input file 410.inp with Abaqus

```
!abaqus job=410
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('410.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('410.fil');
```

Obtain the desired output data

```
out = Rec410(Rec)
```

```
out =

   1.0e-03 *

    1.0000    1.0000
    1.0000    1.0000
    1.0000    1.0000
    1.0000    1.0000
         0         0
         0         0
         0         0
         0         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     8
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

If using this toolbox for research or industrial purposes, please cite:

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

PRINCIPAL PLASTIC STRAINS output from Abaqus to Matlab (Record key 411)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\411.inp'],[S(1:a(end)-1),'\411.inp'],'f')
```

Run the input file 411.inp with Abaqus

```
!abaqus job=411
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('411.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('411.fil');
```

Obtain the desired output data

```
out = Rec411(Rec)
```

```
out =

   1.0e-04 *

         0         0         0
         0         0         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
   -0.1277   -0.1277         0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    20
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

ELEMENT DEFINITION output from Abaqus to Matlab (Record key 1900)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1900.inp'],[S(1:a(end)-1),'\1900.inp'],'f')
```

Run the input file 1900.inp with Abaqus

```
!abaqus job=1900
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1900.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1900.fil');
```

Obtain the desired output data

```
out = Rec1900(Rec)
```

```
out = 

    [ 1]    'FRAME2D'    [5]    [3]
    [ 2]    'FRAME2D'    [3]    [1]
    [ 3]    'FRAME2D'    [6]    [4]
    [ 4]    'FRAME2D'    [4]    [2]
    [ 5]    'FRAME2D'    [3]    [4]
    [ 6]    'FRAME2D'    [1]    [2]
    [ 7]    'FRAME2D'    [5]    [4]
    [ 8]    'FRAME2D'    [6]    [3]
    [ 9]    'FRAME2D'    [3]    [2]
    [10]    'FRAME2D'    [4]    [1]
    [11]    'MASS'       [1]    [0]
    [12]    'MASS'       [2]    [0]
    [13]    'MASS'       [3]    [0]
    [14]    'MASS'       [4]    [0]
    [15]    'MASS'       [5]    [0]
    [16]    'MASS'       [6]    [0]
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     4
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    16
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

NODE DEFINITION output from Abaqus to Matlab (Record key 1901)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1901.inp'],[S(1:a(end)-1),'\1901.inp'],'f')
```

Run the input file 1901.inp with Abaqus

```
!abaqus job=1901
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1901.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1901.fil');
```

Obtain the desired output data

```
out = Rec1901(Rec)
```

```
out =

    1.0000  720.0000  360.0000         0   -0.7071    0.7071
    2.0000  720.0000         0         0    0.7071    0.7071
    3.0000  360.0000  360.0000         0    0.7071    0.7071
    4.0000  360.0000         0         0   -0.7071    0.7071
    5.0000         0  360.0000         0    0.7071    0.7071
    6.0000         0         0         0   -0.7071    0.7071
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     6
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     6
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

ACTIVE DEGREES OF FREEDOM output from Abaqus to Matlab (Record key 1902)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1902.inp'],[S(1:a(end)-1),'\1902.inp'],'f')
```

Run the input file 1902.inp with Abaqus

```
!abaqus job=1902
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1902.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1902.fil');
```

Obtain the desired output data

```
out = Rec1902(Rec)
```

```
out =

  Columns 1 through 13

     0     1     2     3     0     0     4     0     0     0     0     0     0

  Columns 14 through 26

     0     0     0     0     0     0     0     0     0     0     0     0     0

  Columns 27 through 30

     0     0     0     0
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

    30
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     1
```

Check class of output

```
cOut=class(out)
```

```
cOut =

double
```

```
If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

OUTPUT REQUEST DEFINITION output from Abaqus to Matlab (Record key 1911)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1911.inp'],[S(1:a(end)-1),'\1911.inp'],'f')
```

Run the input file 1911.inp with Abaqus

```
!abaqus job=1911
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1911.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1911.fil');
```

Obtain the desired output data

```
out = Rec1911(Rec)
```

```
out =

    [0]    ''    'MASS'
    [0]    ''    'MASS'
    [1]    ''    ''
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     3
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

     3
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

ABAQUS ANALYSIS INFORMATION output from Abaqus to Matlab (Record key 1921)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1921.inp'],[S(1:a(end)-1),'\1921.inp'],'f')
```

Run the input file 1921.inp with Abaqus

```
!abaqus job=1921
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1921.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1921.fil');
```

Obtain the desired output data

```
out = Rec1921(Rec)
```

out =

    '6.13-1'   '05-Ιαν-2017'   '12:57:49'   [16]   [6]   [419.6468]

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

nAttr =

     6

Check the number of entries

```
nEntr=size(out,1)
```

nEntr =

     1

Check class of output

```
cOut=class(out)
```

cOut =

cell

```
Abaqus2Matlab - www.abaqus2matlab.com
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

LABEL CROSS-REFERENCE output from Abaqus to Matlab (Record key 1940)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1940.inp'],[S(1:a(end)-1),'\1940.inp'],'f')
```

Run the input file 1940.inp with Abaqus

```
!abaqus job=1940
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1940.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1940.fil');
```

Obtain the desired output data

```
out = Rec1940(Rec)
```

```
out =

    [ 1]    'ASSEMBLY_PART-5-1_MASSES'
    [ 2]    'ASSEMBLY_PART-5-1_SET1'
    [ 3]    'ASSEMBLY_PART-5-1_SET10'
    [ 4]    'ASSEMBLY_PART-5-1_SET2'
    [ 5]    'ASSEMBLY_PART-5-1_SET3'
    [ 6]    'ASSEMBLY_PART-5-1_SET4'
    [ 7]    'ASSEMBLY_PART-5-1_SET5'
    [ 8]    'ASSEMBLY_PART-5-1_SET6'
    [ 9]    'ASSEMBLY_PART-5-1_SET7'
    [10]    'ASSEMBLY_PART-5-1_SET8'
    [11]    'ASSEMBLY_PART-5-1_SET9'
    [12]    'ASSEMBLY_SET21'
    [13]    'ASSEMBLY_SET22'
    [14]    'ANTIALIASING'
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

     2
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    14
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

```
Copyright (c) 2016 by George Papazafeiropoulos

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

MODAL output from Abaqus to Matlab (Record key 1980)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1980.inp'],[S(1:a(end)-1),'\1980.inp'],'f')
```

Run the input file 1980.inp with Abaqus

```
!abaqus job=1980
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1980.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1980.fil');
```

Obtain the desired output data

```
out = Rec1980(Rec)
```

```
out =

  Columns 1 through 6

    [ 1]    [         0]    [2.7590]    [0]    [ 1.6575e-14]    [ 2.4682e-13]
    [ 2]    [         0]    [2.4797]    [0]    [-1.7055e-13]    [-4.6938e-12]
    [ 3]    [8.8818e-16]    [2.1312]    [0]    [ 7.1353e-18]    [-4.8642e-12]
    [ 4]    [6.2172e-15]    [1.3763]    [0]    [-6.8128e-17]    [-5.4767e-13]
    [ 5]    [1.0658e-14]    [1.4879]    [0]    [ 1.8577e-13]    [ 4.4409e-12]
    [ 6]    [5.6843e-14]    [1.7112]    [0]    [-8.8930e-14]    [-5.7335e-12]
    [ 7]    [    1.4164]    [2.5307]    [0]    [ 5.4838e-17]    [     1.1051]
    [ 8]    [   13.1511]    [2.7664]    [0]    [     1.1557]    [ 6.8627e-15]
    [ 9]    [   14.6462]    [3.1014]    [0]    [ 1.9957e-15]    [     0.5323]
    [10]    [   43.2512]    [3.8919]    [0]    [-1.8542e-16]    [    -0.0336]

  Columns 7 through 11

    [-0.0140]    [-248.3574]    [  12.9012]    [ 1.6389e-10]    [7.5800e-28]
    [ 0.0650]    [ 100.9057]    [ -85.1131]    [-2.5271e-09]    [7.2127e-26]
    [ 1.3399]    [ 247.4820]    [-361.7161]    [-3.2899e-09]    [1.0851e-34]
    [ 1.2499]    [ 197.1864]    [-680.0529]    [-3.7039e-10]    [6.3882e-33]
    [-0.0264]    [  32.5043]    [-372.7573]    [ 2.9699e-09]    [5.1346e-26]
    [ 0.0822]    [  67.2987]    [ 301.4158]    [-3.5950e-09]    [1.3533e-26]
    [      0]    [        0]    [        0]    [   747.4092]    [7.6103e-33]
    [      0]    [        0]    [        0]    [  -208.0211]    [    3.6947]
    [      0]    [        0]    [        0]    [    44.9680]    [1.2353e-29]
    [      0]    [        0]    [        0]    [   -28.4812]    [1.3381e-31]

  Columns 12 through 15

    [1.6808e-25]    [5.4289e-04]    [1.7018e+05]    [  459.2179]
    [5.4633e-23]    [    0.0105]    [2.5248e+04]    [1.7964e+04]
    [5.0425e-23]    [    3.8262]    [1.3053e+05]    [2.7885e+05]
    [4.1282e-25]    [    2.1502]    [5.3515e+04]    [6.3652e+05]
    [2.9344e-23]    [    0.0010]    [1.5720e+03]    [2.0674e+05]
    [5.6254e-23]    [    0.0116]    [7.7504e+03]    [1.5547e+05]
    [    3.0903]    [         0]    [         0]    [         0]
    [1.3029e-28]    [         0]    [         0]    [         0]
    [    0.8787]    [         0]    [         0]    [         0]
    [    0.0044]    [         0]    [         0]    [         0]

  Column 16

    [7.4104e-20]
    [1.5836e-17]
    [2.3068e-17]
    [1.8882e-19]
    [1.3124e-17]
    [2.2116e-17]
    [1.4137e+06]
    [1.1971e+05]
    [6.2713e+03]
    [3.1570e+03]
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

    16
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

J-INTEGRAL output from Abaqus to Matlab (Record key 1991)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\1991.inp'],[S(1:a(end)-1),'\1991.inp'],'f')
```

Run the input file 1991.inp with Abaqus

```
!abaqus job=1991
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('1991.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('1991.fil');
```

Obtain the desired output data

```
out = Rec1991(Rec)
```

```
out =

  Columns 1 through 6

    [1]    'T0'     [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T1'     [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T2'     [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T3'     [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T4'     [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T5'     [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T6'     [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T7'     [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T8'     [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T9'     [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T10'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T11'    [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T12'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T13'    [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T14'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T15'    [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T16'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T17'    [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T18'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]
    [1]    'T19'    [8]    [1.3603e-07]    [1.3316e-07]    [1.3367e-07]
    [1]    'T20'    [8]    [1.3615e-07]    [1.3314e-07]    [1.3355e-07]

  Columns 7 through 10

    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]
    [1.3380e-07]    [1.3387e-07]    [1.3394e-07]    [1.3400e-07]
    [1.3356e-07]    [1.3352e-07]    [1.3345e-07]    [1.3336e-07]

  Column 11

    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
```

```
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
    [1.3406e-07]
    [1.3325e-07]
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

    11
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    21
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

```
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)
```

Published with MATLAB® R2015a

INCREMENT START RECORD output from Abaqus to Matlab (Record key 2000)

In this example a simple Abaqus model is analysed and results are retrieved by postprocessing the results *.fil file generated by Abaqus using Matlab. For more information please see the Documentation of Abaqus2Matlab toolbox.

Contents

Run Abaqus model

```
S = which('Documentation.m');
% Change current directory to Abaqus working directory
a = strfind(S,'\');
cd(S(1:a(end)-1))
```

Copy the input file to be run by Abaqus into the Abaqus working directory

```
copyfile([S(1:a(end)-1),'\AbaqusInputFiles\2000.inp'],[S(1:a(end)-1),'\2000.inp'],'f')
```

Run the input file 2000.inp with Abaqus

```
!abaqus job=2000
```

Pause Matlab execution to give Abaqus enough time to create the lck file

```
pause(10)
```

If the lck file exists then halt Matlab execution

```
while exist('2000.lck','file')==2
    pause(0.1)
end
```

Postprocess Abaqus results file with Matlab

Assign all lines of the fil file in an one-row string (after Abaqus analysis terminates)

```
Rec = Fil2str('2000.fil');
```

Obtain the desired output data

```
out = Rec2000(Rec)
```

```
out =

  Columns 1 through 9

    [1.5802]    [1.5802]    [0]    [0]    [21]    [1]    [ 1]    [0]    [0]
    [3.1605]    [3.1605]    [0]    [0]    [21]    [1]    [ 2]    [0]    [0]
    [4.4117]    [4.4117]    [0]    [0]    [21]    [1]    [ 3]    [0]    [0]
    [5.6630]    [5.6630]    [0]    [0]    [21]    [1]    [ 4]    [0]    [0]
    [6.5589]    [6.5589]    [0]    [0]    [21]    [1]    [ 5]    [0]    [0]
    [7.4549]    [7.4549]    [0]    [0]    [21]    [1]    [ 6]    [0]    [0]
    [8.3508]    [8.3508]    [0]    [0]    [21]    [1]    [ 7]    [0]    [0]
    [9.0166]    [9.0166]    [0]    [0]    [21]    [1]    [ 8]    [0]    [0]
    [9.6823]    [9.6823]    [0]    [0]    [21]    [1]    [ 9]    [0]    [0]
    [    10]    [    10]    [0]    [0]    [21]    [1]    [10]    [0]    [0]

  Columns 10 through 12

    [0]    [1.5802]    ' STEP-1'
    [0]    [1.5802]    ' STEP-1'
    [0]    [1.2513]    ' STEP-1'
    [0]    [1.2513]    ' STEP-1'
    [0]    [0.8959]    ' STEP-1'
    [0]    [0.8959]    ' STEP-1'
    [0]    [0.8959]    ' STEP-1'
    [0]    [0.6658]    ' STEP-1'
    [0]    [0.6658]    ' STEP-1'
    [0]    [0.3177]    ' STEP-1'
```

Verify output

Check number of attributes

```
nAttr=size(out,2)
```

```
nAttr =

    12
```

Check the number of entries

```
nEntr=size(out,1)
```

```
nEntr =

    10
```

Check class of output

```
cOut=class(out)
```

```
cOut =

cell
```

Abaqus2Matlab - www.abaqus2matlab.com

If using this toolbox for research or industrial purposes, please cite:
G. Papazafeiropoulos, M. Muniz-Calvente, E. Martinez-Paneda.
Abaqus2Matlab: a suitable tool for finite element post-processing
(submitted)

Published with MATLAB® R2015a

```
type('8.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS COORDINATE OUTPUT TO MATLAB (COORD, RECORD KEY 8)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
*ELEMENT, TYPE=CPE4
 1, 4, 2, 1, 3
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 3
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 1, 4
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 1000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*END STEP
```

```
type('10.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL FLUX CAUSED BY HEAT OUTPUT TO MATLAB (NFLUX, RECORD KEY 10)
*NODE
 1,1.,
 7,1.,3.
 57,5.
 63,5.,3.
*NGEN,NSET=NL
 1,7,1
*NGEN,NSET=NR
 57,63,1
*NFILL,NSET=NALL
 NL,NR,8,7
*ELEMENT,TYPE=DC2D6,ELSET=EALL
 1,1,17,3,9,10,2
 13,1,15,17,8,16,9
*ELGEN,ELSET=EALL
 1,3,2,1,4,14,3
 13,3,2,1,4,14,3
*NSET,NSET=NBOT
 1,8,15,22,29,30,43,50,57
*NSET,NSET=NLFT
 1,2,3,4,5,6,7
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*CONDUCTIVITY
 4.85E-4,
*DENSITY
 0.283,
*SPECIFIC HEAT
 0.116,
*STEP
*HEAT TRANSFER,STEADY STATE
 1.,1.
*DFLUX
 EALL,BF,.3
*BOUNDARY
 NLFT,11,11,200.
 NBOT,11,11,400.
*FILE FORMAT, ASCII
*EL FILE
 NFLUX
*ENDSTEP
```

```
type('11.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS STRESS OUTPUT TO MATLAB (S, RECORD KEY 11)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
*ELEMENT, TYPE=CPE4
 1, 4, 2, 1, 3
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 3
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 1, 4
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 1000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 S
*END STEP
```

```
type('12.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS STRESS INVARIANT OUTPUT TO MATLAB (SINV, RECORD KEY 12)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      0,       0,       0
 2,      1,       0,       0
 3,      1,       1,       0
 4,      0,       1,       0
 5,      0,       0,       1
 6,      1,       0,       1
 7,      1,       1,       1
 8,      0,       1,       1
*ELEMENT, TYPE=C3D8
 1, 1, 2, 3, 4, 5, 6, 7, 8
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
 6,5
 7,6
 8,7
 9,8
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 1,2,3,4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5,6,7,8
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 100000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 3, -100.
*FILE FORMAT, ASCII
*EL FILE
 SINV
*END STEP
```

```
type('13.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS SECTION FORCE AND MOMENT OUTPUT TO MATLAB (SF, RECORD KEY 13)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,      0,      360
 6,      0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 SF
*END STEP
```

```
type('14.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ENERGY DENSITY OUTPUT TO MATLAB (ENER, RECORD KEY 14)
*RESTART,WRITE,FREQUENCY=1
*NODE
 1,
 7,60.
 13,180.
 15,228.
 19,348.
 801,,144.
 807,60.,144.
 813,180.,144.
 815,228.,144.
 819,348.,144.
 20,696.
 820,696.,144.
*NGEN,NSET=BASE
 1,7
 7,13
 13,15
 15,19
*NSET,NSET=F1
 801,
*NSET,NSET=F2,GENERATE
 802,807
*NGEN,NSET=CENTER
 1,801,100
*NGEN,NSET=TOP
 801,807
 807,813
 813,815
 815,819
*NFILL
 BASE,TOP,8,100
*NGEN,NSET=FAR
 20,820,200
*ELEMENT,TYPE=CPE8R
 1,1,3,203,201,2,103,202,101
*ELGEN,ELSET=ALL
 1,4,200,1,9,2,10
*ELSET,ELSET=PRINTEL
 4
*SOLID SECTION,ELSET=ALL,MATERIAL=A1
*MATERIAL,NAME= A1
*ELASTIC
 30000.,0.3
*DRUCKER PRAGER,SHEAR CRITERION=LINEAR
 30.16,1.0,30.16
*DRUCKER PRAGER HARDENING
 19.8,0.
*DRUCKER PRAGER CREEP, LAW=TIME
 2.96E-8,2
*ELEMENT,TYPE=CINPE5R
 101,219,19,20,220,119
*ELGEN,ELSET=FAR
```

```
 101,4,200,1
*SOLID SECTION,ELSET=FAR,MATERIAL=A2
*MATERIAL,NAME= A2
*ELASTIC
 30000.,0.3
*EQUATION
 2,
 F2,2,1.,801,2,-1.
*BOUNDARY
 CENTER,1
 F2,1
 BASE,1,2
*AMPLITUDE, NAME=RAMP
 0.,0.,1.,1.
*STEP,INC=50, UNSYMM=YES
 PRESCRIBE DISPLACEMENT
*VISCO, CETOL=0.01
 .025,1.,,.1
*BOUNDARY, AMP=RAMP
 801,2,,-5.0
*MONITOR,NODE=801,DOF=2
*CONTROLS,ANALYSIS=DISCONTINUOUS
*FILE FORMAT, ASCII
*EL FILE,ELSET=PRINTEL
 ENER
*END STEP
```

```
type('18.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PORE PRESSURE OUTPUT TO MATLAB (POR, RECORD KEY 18)
*NODE
 1, 1.
 3, 4.
 7, 1., 5.
 9, 4., 5.
*NGEN, NSET=SIDE1
 1, 3
*NGEN, NSET=SIDE3
 7, 9
*NGEN, NSET=SIDE4
 1, 7, 3
*NGEN, NSET=SIDE2
 3, 9, 3
*NSET, NSET=NALL, GENERATE
 1, 9
*NSET,NSET=CORNERS1
 1,3
*NSET,NSET=CORNERS2
 3,9
*NSET,NSET=CORNERS3
 7,9
*NSET,NSET=CORNERS4
 1,7
*NSET,NSET=CORNERS
 1,3,7,9
*ELEMENT,TYPE=CPE4P, ELSET=EALL
 1, 1,3,9,7
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC
 1.E8,
*PERMEABILITY,SPECIFIC=1.0
 1.E-5,
*DENSITY
 1.4142,
*INITIAL CONDITIONS,TYPE=RATIO
 NALL,1.
*STEP
*SOILS,CONSOLIDATION
 1. , 1.
*FILE FORMAT, ASCII
*EL FILE
 POR
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS, 8
*DLOAD,OP=NEW
 1,CENTRIF,100.,0.,2.5,0.,0.,0.,1.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1.,1.
*DLOAD, OP=NEW
```

```
 1, BX, 100.
 1, BY, 100.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*DLOAD, OP=NEW
 1, GRAV, 100.,1,1,0
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*DLOAD, OP=NEW
 1, P1, 100.
 1, P3, 100.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*DLOAD, OP=NEW
 1, P2, 100.
 1, P4, 100.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*DLOAD, OP=NEW
 1, HP4, 100., 5., 0.
 1, HP2, 100., 5., 0.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*DLOAD, OP=NEW
 1, HP1, 100., 5., 0.
 1, HP3, 100., 5., 0.
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS3, 8
*DLOAD, OP=NEW
*FLOW, OP=NEW
 1, Q1, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY,OP=NEW
 NALL, 1,2
 CORNERS4, 8
*FLOW, OP=NEW
 1, Q2, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
```

```
 CORNERS1, 8
*FLOW, OP=NEW
 1, Q3, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS2, 8
*FLOW, OP=NEW
 1, Q4, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS3, 8
*FLOW, OP=NEW
*DFLOW, OP=NEW
 1, S1, 3.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS4, 8
*DFLOW, OP=NEW
 1, S2, 3.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS1, 8
*DFLOW, OP=NEW
 1, S3, 3.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS2, 8
*DFLOW, OP=NEW
 1, S4, 3.E-5
*END STEP
```

```
type('19.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ENERGY (SUMMED OVER ELEMENT) OUTPUT TO MATLAB (ELEN, RECORD KEY 19)
*RESTART,WRITE,FREQUENCY=1
*NODE
 1,
 7,60.
 13,180.
 15,228.
 19,348.
 801,,144.
 807,60.,144.
 813,180.,144.
 815,228.,144.
 819,348.,144.
 20,696.
 820,696.,144.
*NGEN,NSET=BASE
 1,7
 7,13
 13,15
 15,19
*NSET,NSET=F1
 801,
*NSET,NSET=F2,GENERATE
 802,807
*NGEN,NSET=CENTER
 1,801,100
*NGEN,NSET=TOP
 801,807
 807,813
 813,815
 815,819
*NFILL
 BASE,TOP,8,100
*NGEN,NSET=FAR
 20,820,200
*ELEMENT,TYPE=CPE8R
 1,1,3,203,201,2,103,202,101
*ELGEN,ELSET=ALL
 1,4,200,1,9,2,10
*ELSET,ELSET=PRINTEL
 4
*SOLID SECTION,ELSET=ALL,MATERIAL=A1
*MATERIAL,NAME= A1
*ELASTIC
 30000.,0.3
*DRUCKER PRAGER,SHEAR CRITERION=LINEAR
 30.16,1.0,30.16
*DRUCKER PRAGER HARDENING
 19.8,0.
*DRUCKER PRAGER CREEP, LAW=TIME
 2.96E-8,2
*ELEMENT,TYPE=CINPE5R
 101,219,19,20,220,119
*ELGEN,ELSET=FAR
```

```
 101,4,200,1
*SOLID SECTION,ELSET=FAR,MATERIAL=A2
*MATERIAL,NAME= A2
*ELASTIC
 30000.,0.3
*EQUATION
 2,
 F2,2,1.,801,2,-1.
*BOUNDARY
 CENTER,1
 F2,1
 BASE,1,2
*AMPLITUDE, NAME=RAMP
 0.,0.,1.,1.
*STEP,INC=50, UNSYMM=YES
 PRESCRIBE DISPLACEMENT
*VISCO, CETOL=0.01
 .025,1.,,.1
*BOUNDARY, AMP=RAMP
 801,2,,-5.0
*MONITOR,NODE=801,DOF=2
*CONTROLS,ANALYSIS=DISCONTINUOUS
*FILE FORMAT, ASCII
*EL FILE,ELSET=PRINTEL
 ELEN
*END STEP
```

```
type('21.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TOTAL STRAIN OUTPUT TO MATLAB (E, RECORD KEY 21)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,     0,      0,      0
 2,     1,      0,      0
 3,     1,      1,      0
 4,     0,      1,      0
 5,     0,      0,      1
 6,     1,      0,      1
 7,     1,      1,      1
 8,     0,      1,      1
*ELEMENT, TYPE=C3D8
 1, 1, 2, 3, 4, 5, 6, 7, 8
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
 6,5
 7,6
 8,7
 9,8
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 1,2,3,4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5,6,7,8
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 100000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 3, -100.
*FILE FORMAT, ASCII
*EL FILE
 E
*END STEP
```

```
type('22.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PLASTIC STRAIN OUTPUT TO MATLAB (PE, RECORD KEY 22)
*RESTART,WRITE,FREQUENCY=1
*NODE
 1,
 7,60.
 13,180.
 15,228.
 19,348.
 801,,144.
 807,60.,144.
 813,180.,144.
 815,228.,144.
 819,348.,144.
 20,696.
 820,696.,144.
*NGEN,NSET=BASE
 1,7
 7,13
 13,15
 15,19
*NSET,NSET=F1
 801,
*NSET,NSET=F2,GENERATE
 802,807
*NGEN,NSET=CENTER
 1,801,100
*NGEN,NSET=TOP
 801,807
 807,813
 813,815
 815,819
*NFILL
 BASE,TOP,8,100
*NGEN,NSET=FAR
 20,820,200
*ELEMENT,TYPE=CPE8R
 1,1,3,203,201,2,103,202,101
*ELGEN,ELSET=ALL
 1,4,200,1,9,2,10
*ELSET,ELSET=PRINTEL
 4
*SOLID SECTION,ELSET=ALL,MATERIAL=A1
*MATERIAL,NAME= A1
*ELASTIC
 30000.,0.3
*DRUCKER PRAGER,SHEAR CRITERION=LINEAR
 30.16,1.0,30.16
*DRUCKER PRAGER HARDENING
 19.8,0.
*DRUCKER PRAGER CREEP, LAW=TIME
 2.96E-8,2
*ELEMENT,TYPE=CINPE5R
 101,219,19,20,220,119
*ELGEN,ELSET=FAR
```

```
 101,4,200,1
*SOLID SECTION,ELSET=FAR,MATERIAL=A2
*MATERIAL,NAME= A2
*ELASTIC
 30000.,0.3
*EQUATION
 2,
 F2,2,1.,801,2,-1.
*BOUNDARY
 CENTER,1
 F2,1
 BASE,1,2
*AMPLITUDE, NAME=RAMP
 0.,0.,1.,1.
*STEP,INC=50, UNSYMM=YES
 PRESCRIBE DISPLACEMENT
*VISCO, CETOL=0.01
 .025,1.,,.1
*BOUNDARY, AMP=RAMP
 801,2,,-5.0
*MONITOR,NODE=801,DOF=2
*CONTROLS,ANALYSIS=DISCONTINUOUS
*FILE FORMAT, ASCII
*EL FILE, ELSET=PRINTEL
 PE
*END STEP
```

```
type('23.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS CREEP STRAIN (INCLUDING SWELLING) OUTPUT TO MATLAB (CE, RECORD KEY 23)
*RESTART,WRITE,FREQ=5
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8R,ELSET=ALLE
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=VVE3
 1.,
*MATERIAL,NAME=VVE3
*HYPERELASTIC,N=1,MODULI=INSTANTANEOUS
 8.,2.,0.1
*VISCOELASTIC,TIME=PRONY
 0.,0.5,3.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
 4,3
*STEP,NLGEOM
*VISCO
 2.,10.,2.,10.,
*BOUNDARY
 5,3,,-.2
 6,3,,-.2
 7,3,,-.2
 8,3,,-.2
*FILE FORMAT, ASCII
*EL FILE
 CE
*END STEP
```

```
type('24.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TOTAL INELASTIC STRAIN OUTPUT TO MATLAB (IE, RECORD KEY 24)
*RESTART,WRITE,FREQ=5
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8R,ELSET=ALLE
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=VVE3
 1.,
*MATERIAL,NAME=VVE3
*HYPERELASTIC,N=1,MODULI=INSTANTANEOUS
 8.,2.,0.1
*VISCOELASTIC,TIME=PRONY
 0.,0.5,3.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
 4,3
*STEP,NLGEOM
*VISCO
 2.,10.,2.,10.,
*BOUNDARY
 5,3,,-.2
 6,3,,-.2
 7,3,,-.2
 8,3,,-.2
*FILE FORMAT, ASCII
*EL FILE
 IE
*END STEP
```

```
type('25.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TOTAL ELASTIC STRAIN OUTPUT TO MATLAB (EE, RECORD KEY 25)
*RESTART,WRITE,FREQ=5
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8R,ELSET=ALLE
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=VVE3
 1.,
*MATERIAL,NAME=VVE3
*HYPERELASTIC,N=1,MODULI=INSTANTANEOUS
 8.,2.,0.1
*VISCOELASTIC,TIME=PRONY
 0.,0.5,3.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
 4,3
*STEP,NLGEOM
*VISCO
 2.,10.,2.,10.,
*BOUNDARY
 5,3,,-.2
 6,3,,-.2
 7,3,,-.2
 8,3,,-.2
*FILE FORMAT, ASCII
*EL FILE
 EE
*END STEP
```

```
type('26.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS UNIT NORMAL TO CRACK IN CONCRETE OUTPUT TO MATLAB (CRACK, RECORD KEY 26)
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8,ELSET=ALLE
 1,1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=ALLE
*MATERIAL,NAME=ALLE
*ELASTIC
 4.65E6,.18
*CONCRETE
 1300.,0.
 2200.,.000027
 3000.,.0001
 3600.,.000225
 4450.,.00055
 4650.,.001
 4200.,.002
 2000.,.0035
*FAILURE RATIOS
 1.18,.15,1.25,.2,0.
 1.18,.05,1.25,.2,40.
*TENSION STIFFENING,DEP=1
 1., 0.,      0., 0.
 0., 3.5E-4,  0., 0.
 1., 0.,     40., 0.
 0., 4.5E-4, 40., 0.
 1., 0.,      0., 2.
 0., 5.5E-4,  0., 2.
 1., 0.,     40., 2.
 0., 6.5E-4, 40., 2.
*SHEAR RETENTION,DEP=1
 1.1,0.,,,20.,0.
 0.9,0.,,,20.,2.
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALLN,20.
*INITIAL CONDITIONS,TYPE=FIELD,VARIABLE=1
 ALLN,1.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
```

```
 4,3
*STEP,INC=20
*STATIC,DIRECT
 10.,20.
*BOUNDARY
 7,3,,.0008
 5,3,,.0008
 6,3,,.0008
 8,3,,.0008
*FILE FORMAT, ASCII
*EL FILE
 CRACK
*END STEP
```

Published with MATLAB® R2015a

```
type('27.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS SECTION THICKNESS OUTPUT TO MATLAB (STH, RECORD KEY 27)
*NODE,NSET=BOTTOM
 1,   0.0,   0.0
 2,  10.0,   0.0
 3,  20.0,   0.0
*NODE,NSET=TOP
 101,   0.0,100.0
 102,  10.0,100.0
 103,  20.0,100.0
*NFILL,NSET=NALL
 BOTTOM,TOP,10,10
*NODAL THICKNESS
 BOTTOM,3.
 TOP,1.
*NODAL THICKNESS, GENERATE
 BOTTOM,TOP,10,10
*ELEMENT,TYPE=M3D8
 1, 1,3,23,21,2,13,22,11
*ELGEN,ELSET=EALL
 1,5,20,20
*MEMBRANE SECTION,MATERIAL=A1,ELSET=EALL,NODAL
 1.0,
*MATERIAL,NAME=A1
*ELASTIC,TYPE=ISOTROPIC
 1000.0,
*BOUNDARY
 BOTTOM,1,2
 NALL,3,3
*STEP
*STATIC
*CLOAD
 101,2,166.66667
 102,2,666.66667
 103,2,166.66667
*FILE FORMAT, ASCII
*EL FILE
 STH
*END STEP
```

```
type('28.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS HEAT FLUX VECTOR OUTPUT TO MATLAB (HFL, RECORD KEY 28)
*RESTART,WRITE,F=100
*NODE
 1, 0.0, 0.0
 2, 0.1, 0.0
 3, 0.1, 0.1
 4, 0.0, 0.1
*NSET,NSET=NALL
 1,2,3,4
*ELEMENT,TYPE=DC2D4
 1, 1,2,3,4
*ELSET,ELSET=EALL
 1
*SOLID SECTION,ELSET=EALL,MATERIAL=MAT
*MATERIAL,NAME=MAT
*CONDUCTIVITY
 1.40,
*SPECIFIC HEAT
 260.0,
*DENSITY
 7800.0,
*FILM PROPERTY,NAME=FILMP
 10.0,  0.0
 16.0,300.0
*AMPLITUDE,NAME=SINK
 0.0,100.0, 3600.0,200.0
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 NALL,0.0
*SURFACE, NAME=SURF1
 1, S3
*SURFACE, NAME=SURF2
 EALL, S2
*SURFACE, NAME=SURF3
 EALL, S4
*SURFACE, NAME=SURF4
 1, S1
*STEP,INC=1000,UNSYMM=YES
*HEATTRANSFER,DELTMX=20.0
 3600.,3600.,,3600.0
*SFILM,AMP=SINK
 SURF1,F,1.0,FILMP
*DSFLUX
 SURF2,S,0.0
 SURF3,S,0.0
 SURF4,S,0.0
*FILE FORMAT, ASCII
*EL FILE
 HFL
*ENDSTEP
```

```
type('29.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS SECTION STRAIN AND CURVATURE OUTPUT TO MATLAB (SE, RECORD KEY 29)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,      0,      360
 6,      0,      0
*ELEMENT, TYPE=B21
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*BEAM SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#2, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#3, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#4, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#5, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#6, MATERIAL=MAT1, SECTION=RECT
```

```
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#7, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#8, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#9, MATERIAL=MAT1, SECTION=RECT
 1,1
*BEAM SECTION, ELSET=_PICKEDSET2_#10, MATERIAL=MAT1, SECTION=RECT
 1,1
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*MATERIAL, NAME=MAT1
*ELASTIC
 230,0.3
*DENSITY
 1
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 SE
*END STEP
```

```
type('31.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS CONCRETE FAILURE OUTPUT TO MATLAB (CONF, RECORD KEY 26)
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8,ELSET=ALLE
 1,1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=ALLE
*MATERIAL,NAME=ALLE
*ELASTIC
 4.65E6,.18
*CONCRETE
 1300.,0.
 2200.,.000027
 3000.,.0001
 3600.,.000225
 4450.,.00055
 4650.,.001
 4200.,.002
 2000.,.0035
*FAILURE RATIOS
 1.18,.15,1.25,.2,0.
 1.18,.05,1.25,.2,40.
*TENSION STIFFENING,DEP=1
 1., 0.,      0., 0.
 0., 3.5E-4,  0., 0.
 1., 0.,     40., 0.
 0., 4.5E-4, 40., 0.
 1., 0.,      0., 2.
 0., 5.5E-4,  0., 2.
 1., 0.,     40., 2.
 0., 6.5E-4, 40., 2.
*SHEAR RETENTION,DEP=1
 1.1,0.,,,20.,0.
 0.9,0.,,,20.,2.
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALLN,20.
*INITIAL CONDITIONS,TYPE=FIELD,VARIABLE=1
 ALLN,1.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
```

```
 4,3
*STEP,INC=20
*STATIC,DIRECT
 1.,4.
*BOUNDARY
 7,3,,.0008
 5,3,,.0008
 6,3,,.0008
 8,3,,.0008
*FILE FORMAT, ASCII
*EL FILE
 CONF
*END STEP
```

```
type('32.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS STRAIN JUMP AT NODES OUTPUT TO MATLAB (SJP, RECORD KEY 32)
*PART,NAME=PART-1
*NODE
 1,  0.0
 2,  1.0
 3,  2.0
 4,  3.0
 5,  0.0, 1.0
 6,  1.0, 1.0
 7,  2.0, 1.0
 8,  3.0, 1.0
 9,  0.0, 2.0
 10,  1.0, 2.0
 11,  2.0, 2.0
 12,  3.0, 2.0
 13,  0.0, 3.0
 14,  1.0, 3.0
 15,  2.0, 3.0
 16,  3.0, 3.0
*ELEMENT,TYPE=CPS4
 1,  1, 2, 6, 5
 2,  2, 3, 7, 6
 3,  3, 4, 8, 7
 4,  5, 6,10, 9
 5,  6, 7,11,10
 6,  7, 8,12,11
 7,  9,10,14,13
 8, 10,11,15,14
 9, 11,12,16,15
*ELSET,ELSET=EELAST, GEN
 1,4
 6,9
*ELSET,ELSET=EDAMAGE
 5,
*ELSET,ELSET=EA
 1,
*SOLID SECTION,ELSET=EELAST,MATERIAL=GLASS_EPOXY,ORIENT=RECT
 1.,
*SOLID SECTION,ELSET=EDAMAGE,MATERIAL=GLASS_EPOXY_DMG,ORIENT=RECT,CONTROLS=SCONT
 1.,
*NSET,NSET=FIX1
 1,5,9,13
*NSET,NSET=FIX2
 1,2,3,4
*NSET,NSET=FIX3,GEN
 1,16
*NSET, NSET=MOVE1
 4,8,12,16
*NSET, NSET=MOVE2
 13,14,15,16
*ORIENTATION,NAME=RECT
 1.0, 0.0, 0.0, 0.0, 1.0, 0.0
 3,0.0
*END PART
```

```
*ASSEMBLY,NAME=ASSEMBLY-1
*INSTANCE,NAME=PART-1-1,PART=PART-1
*END INSTANCE
*END ASSEMBLY
*MATERIAL,NAME=GLASS_EPOXY
*ELASTIC,TYPE=LAMINA
 53.8E9,17.9E9,0.25,8.96E9,8.96E9,6.88E9
*MATERIAL,NAME=GLASS_EPOXY_DMG
*ELASTIC,TYPE=LAMINA
 53.8E9,17.9E9,0.25,8.96E9,8.96E9,6.88E9
*DAMAGE INITIATION,CRITERION=HASHIN,ALPHA=0.0
 1034E6,1034E6,27.6E6,138E6,41.4E6,69E6
*DAMAGE EVOLUTION, TYPE=ENERGY, SOFTENING=LINEAR
 20E6,20E6,4.0E4,1.0E6
*DAMAGE STABILIZATION
 1.E-4,1.E-4,1.E-4,1.E-4
*SECTION CONTROLS, NAME=SCONT, ELEMENT DELETION=NO, MAX DEGRADATION=0.99
*STEP,INC=200
 SMALL DISPLACEMENT ANALYSIS
*STATIC
 1.0,1.0,,1.0
*BOUNDARY
 ASSEMBLY-1.PART-1-1.MOVE1, 1,1, 0.1
 ASSEMBLY-1.PART-1-1.MOVE2, 2,2, 0.1
 ASSEMBLY-1.PART-1-1.FIX1,  1,1
 ASSEMBLY-1.PART-1-1.FIX1,  3,6
 ASSEMBLY-1.PART-1-1.FIX2,  2,2
*FILE FORMAT, ASCII
*EL FILE
 SJP
*END STEP
*STEP,INC=200
 SMALL DISPLACEMENT ANALYSIS
*STATIC
 1.0,1.0,,1.0
*BOUNDARY
 ASSEMBLY-1.PART-1-1.MOVE1, 1,1, -0.1
 ASSEMBLY-1.PART-1-1.MOVE2, 2,2, -0.1
 ASSEMBLY-1.PART-1-1.FIX1,  1,1
 ASSEMBLY-1.PART-1-1.FIX2,  2,2
*END STEP
```

```
type('33.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS FILM OUTPUT TO MATLAB (FILM, RECORD KEY 33)
*RESTART,WRITE,F=100
*NODE
 1, 0.0,  0.0
 11,0.0,  0.1
 12,0.01, 0.0
 22,0.01, 0.1
*NGEN,NSET=NALL
 1,11
 12,22
*ELEMENT,TYPE=DC2D4
 1, 1,12,13,2
*ELGEN,ELSET=EALL
 1, 10, 1
*SOLID SECTION,ELSET=EALL,MATERIAL=MAT
*MATERIAL,NAME=MAT
*CONDUCTIVITY
 1.40,
*SPECIFIC HEAT
 260.0,
*DENSITY
 7800.0,
*FILM PROPERTY,NAME=FILMP
 10.0,  0.0
 16.0,300.0
*AMPLITUDE,NAME=SINK
 0.0,100.0, 3600.0,200.0
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 NALL,0.0
*SURFACE, NAME=SURF1
 10, S3
*SURFACE, NAME=SURF2
 EALL, S2
*SURFACE, NAME=SURF3
 EALL, S4
*SURFACE, NAME=SURF4
 1, S1
*STEP,INC=1000,UNSYMM=YES
*HEATTRANSFER,DELTMX=20.0
 100.,3600.,,100.0
*SFILM,AMP=SINK
 SURF1,F,1.0,FILMP
*DSFLUX
 SURF2,S,0.0
 SURF3,S,0.0
 SURF4,S,0.0
*FILE FORMAT, ASCII
*EL FILE
 FILM
*ENDSTEP
```

Published with MATLAB® R2015a

```
type('34.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS RADIATION OUTPUT TO MATLAB (RAD, RECORD KEY 34)
*RESTART,WRITE
*NODE,NSET=ALL
 1, 0., 0., 0.
 2, 7., 0., 0.
 3, 7., 0., -7.
 4, 0., 0., -7.
 5, 0., 7., 0.
 6, 7., 7., 0.
 7, 7., 7., -7.
 8, 0., 7., -7.
*NSET,NSET=FIX1
 1,2,3,4
*NSET,NSET=FIX2
 5,6,7,8
*NSET,NSET=FIX3
 1,2,6,5
*NSET,NSET=FIX4
 2,3,7,6
*NSET,NSET=FIX5
 3,4,7,8
*NSET,NSET=FIX6
 1,4,5,8
*ELEMENT,TYPE=DC3D8, ELSET=EALL
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*SURFACE,NAME=SIDE
 1,S3
*PHYSICAL CONSTANTS, ABSOLUTE ZERO=-460.,STEFAN BOLTZMANN=5.0E-8
*MATERIAL,NAME=A1
*CONDUCTIVITY
 3.77E-5,
*DENSITY
 82.9,
*SPECIFIC HEAT
 .39,
*BOUNDARY
 FIX1,11
*STEP
*HEAT TRANSFER, STEADY STATE
*DFLUX
 1, BF, .3
*FILE FORMAT, ASCII
*EL FILE
 RAD
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX2, 11
*RADIATE, OP=NEW
*DFLUX, OP=NEW
 1, S1, .3
*END STEP
```

```
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX1, 11
*DFLUX,OP=NEW
 1, S2, .3
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX5, 11
*DFLUX,OP=NEW
 1, S3, .3
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX6, 11
*DFLUX,OP=NEW
 1, S4, .3
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX3, 11
*DFLUX,OP=NEW
 1, S5, .3
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*BOUNDARY, OP=NEW
 FIX4, 11
*DFLUX,OP=NEW
 1, S6, .3
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*FILM,OP=NEW
 1, F1, 75., .103
*BOUNDARY,OP=NEW
 FIX2, 11
*DFLUX,OP=NEW
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*DFLUX,OP=NEW
*FILM, OP=NEW
 1, F2, 75., .103
*BOUNDARY,OP=NEW
 FIX1, 11
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*FILM,OP=NEW
 1, F3, 75., .103
*BOUNDARY,OP=NEW
 FIX5, 11
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*FILM,OP=NEW
```

```
 1, F4, 75., .103
*BOUNDARY,OP=NEW
 FIX6, 11
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*FILM,OP=NEW
 1, F5, 75., .103
*BOUNDARY,OP=NEW
 FIX3, 11
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*FILM,OP=NEW
 1, F6, 75., .103
*BOUNDARY,OP=NEW
 FIX4, 11
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*RADIATE, OP=NEW
 1, R1, 75., 1.E-6
*BOUNDARY,OP=NEW
 FIX2,11
*FILM,OP=NEW
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*FILM, OP=NEW
*RADIATE, OP=NEW
 1, R2, 75., 1.E-6
*BOUNDARY,OP=NEW
 FIX1, 11
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*RADIATE, OP=NEW
 1, R3, 75., 1.E-6
*BOUNDARY,OP=NEW
 FIX5,11
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*RADIATE, OP=NEW
 1, R4, 75., 1.E-6
*BOUNDARY, OP=NEW
 FIX6,11
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*RADIATE, OP=NEW
 1, R5, 75., 1.E-6
*BOUNDARY,OP=NEW
 FIX3,11
*END STEP
*STEP
*HEAT TRANSFER, STEADY STATE
*RADIATE, OP=NEW
 1, R6, 75., 1.E-6
*BOUNDARY, OP=NEW
```

```
  FIX4,11
 *END STEP
```

Published with MATLAB® R2015a

```
type('35.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS SATURATION (PORE PRESSURE ANALYSIS) OUTPUT TO MATLAB (SAT, RECORD KEY 35)
*NODE,NSET=ALLN
 1,0.,0.
 3,.00508,0.
 101,0.,.0508
 103,.00508,.0508
*NGEN,NSET=BOT
 1,3,1
*NGEN,NSET=TOP
 101,103,1
*NFILL,NSET=ALLN
 BOT,TOP,20,5
*NSET,NSET=LHS,GEN
 1,101,5
*NSET,NSET=RHS,GEN
 3,103,5
*ELEMENT,TYPE=CPE8RP,ELSET=BLOCK
 1,1,3,13,11,2,8,12,6
*ELGEN,ELSET=BLOCK
 1,10,10,1
*ELSET,ELSET=OUTE
 1,3,5,7,9
*SOLID SECTION,ELSET=BLOCK,MATERIAL=SEP2
 .02,
*MATERIAL,NAME=SEP2
*ELASTIC
 10000.,0.
*POROUS BULK MODULI
 ,2.E9
*PERMEABILITY,SPECIFIC=10000.
 3.7E-4,
*SORPTION,LAW=TABULAR,TYPE=ABSORPTION
 -100000.,.04
 -10000.,.05
 -4500.,.1
 -3500.,.18
 -2000.,.45
 -1000.,.91
 0.,1.
*SORPTION,LAW=TABULAR,TYPE=EXSORPTION
 -100000.,.09
 -10000.,.1
 -8000.,.11
 -6000.,.18
 -4500.,.33
 -3000.,.79
 -2000.,.91
 0.,1.
*SORPTION,TYPE=SCANNING
 9.45E6,
*GEL
 .0005,.0015,1.E8,500.
*INITIAL CONDITIONS,TYPE=SATURATION
 ALLN,.05
```

```
*NSET,NSET=PORN,GEN
 1,101,10
 3,103,10
*INITIAL CONDITIONS,TYPE=PORE
 PORN,-10000.
*INITIAL CONDITIONS,TYPE=RATIO
 ALLN,5.
*BOUNDARY
 BOT,2
 ALLN,1
*RESTART,WRITE,FREQ=10
*STEP
*SOILS,CONSOLIDATION
 1.E-7,1.E-7
*DLOAD
 10,P3,-500.
*CONTROLS,ANAL=DISC
*FILE FORMAT, ASCII
*EL FILE
 SAT
*END STEP
```

```
type('38.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS MASS CONCENTRATION (MASS DIFFUSION ANALYSIS) OUTPUT TO MATLAB (CONC, RECORD
KEY 38)
*NODE,NSET=ALL
 1, 0.
 2, 7.
 3, 7., 7.
 4, 0., 7.
*NSET,NSET=FIX1
 1,2
*NSET,NSET=FIX2
 2,3
*NSET,NSET=FIX3
 3,4
*NSET,NSET=FIX4
 1,4
*ELEMENT,TYPE=DC2D3, ELSET=EALL
 1,1,2,3
 2,1,3,4
*ORIENTATION,NAME=RECT
 1.0, 0.0, 0.0, 0.0, 1.0, 0.0
 1, 0.0
*SOLID SECTION,MATERIAL=A1, ELSET=EALL,ORIENT=RECT
*MATERIAL,NAME=A1
*DIFFUSIVITY,TYPE=ORTHO,LAW=GENERAL
 3.77E-5,7.54E-5,11.31E-5
*SOLUBILITY
 1.,
*BOUNDARY
 FIX1, 11
*STEP
*MASS DIFFUSION,STEADY STATE
*FILE FORMAT, ASCII
*EL FILE
 CONC
*DFLUX,OP=NEW
 1, BF, .3
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX3,11
*DFLUX,OP=NEW
 1, S1, .3
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX4,11
*DFLUX,OP=NEW
 1, S2, .3
*EL FILE
 CONC
*END STEP
*STEP
```

```
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX1,11
*DFLUX,OP=NEW
 2, S2, .3
*EL FILE
 CONC
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX2,11
*DFLUX,OP=NEW
 2, S3, .3
*END STEP
```

```
type('40.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS GEL (PORE PRESSURE ANALYSIS) OUTPUT TO MATLAB (GELVR, RECORD KEY 40)
*NODE,NSET=ALLN
 1,0.,0.
 3,.00508,0.
 101,0.,.0508
 103,.00508,.0508
*NGEN,NSET=BOT
 1,3,1
*NGEN,NSET=TOP
 101,103,1
*NFILL,NSET=ALLN
 BOT,TOP,20,5
*NSET,NSET=LHS,GEN
 1,101,5
*NSET,NSET=RHS,GEN
 3,103,5
*ELEMENT,TYPE=CPE8RP,ELSET=BLOCK
 1,1,3,13,11,2,8,12,6
*ELGEN,ELSET=BLOCK
 1,10,10,1
*ELSET,ELSET=OUTE
 1,3,5,7,9
*SOLID SECTION,ELSET=BLOCK,MATERIAL=SEP2
 .02,
*MATERIAL,NAME=SEP2
*ELASTIC
 10000.,0.
*POROUS BULK MODULI
 ,2.E9
*PERMEABILITY,SPECIFIC=10000.
 3.7E-4,
*SORPTION,LAW=TABULAR,TYPE=ABSORPTION
 -100000.,.04
 -10000.,.05
 -4500.,.1
 -3500.,.18
 -2000.,.45
 -1000.,.91
 0.,1.
*SORPTION,LAW=TABULAR,TYPE=EXSORPTION
 -100000.,.09
 -10000.,.1
 -8000.,.11
 -6000.,.18
 -4500.,.33
 -3000.,.79
 -2000.,.91
 0.,1.
*SORPTION,TYPE=SCANNING
 9.45E6,
*GEL
 .0005,.0015,1.E8,500.
*INITIAL CONDITIONS,TYPE=SATURATION
 ALLN,.05
```

```
*NSET,NSET=PORN,GEN
 1,101,10
 3,103,10
*INITIAL CONDITIONS,TYPE=PORE
 PORN,-10000.
*INITIAL CONDITIONS,TYPE=RATIO
 ALLN,5.
*BOUNDARY
 BOT,2
 ALLN,1
*RESTART,WRITE,FREQ=10
*STEP
*SOILS,CONSOLIDATION
 1.E-7,1.E-7
*DLOAD
 10,P3,-500.
*CONTROLS,ANAL=DISC
*FILE FORMAT, ASCII
*EL FILE
 GELVR
*END STEP
```

```
type('43.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TOTAL FLUID VOLUME RATIO OUTPUT TO MATLAB (FLUVR, RECORD KEY 43)
*NODE,NSET=ALLN
 1,0.,0.
 3,.00508,0.
 101,0.,.0508
 103,.00508,.0508
*NGEN,NSET=BOT
 1,3,1
*NGEN,NSET=TOP
 101,103,1
*NFILL,NSET=ALLN
 BOT,TOP,20,5
*NSET,NSET=LHS,GEN
 1,101,5
*NSET,NSET=RHS,GEN
 3,103,5
*ELEMENT,TYPE=CPE8RP,ELSET=BLOCK
 1,1,3,13,11,2,8,12,6
*ELGEN,ELSET=BLOCK
 1,10,10,1
*ELSET,ELSET=OUTE
 1,3,5,7,9
*SOLID SECTION,ELSET=BLOCK,MATERIAL=SEP2
 .02,
*MATERIAL,NAME=SEP2
*ELASTIC
 10000.,0.
*POROUS BULK MODULI
 ,2.E9
*PERMEABILITY,SPECIFIC=10000.
 3.7E-4,
*SORPTION,LAW=TABULAR,TYPE=ABSORPTION
 -100000.,.04
 -10000.,.05
 -4500.,.1
 -3500.,.18
 -2000.,.45
 -1000.,.91
 0.,1.
*SORPTION,LAW=TABULAR,TYPE=EXSORPTION
 -100000.,.09
 -10000.,.1
 -8000.,.11
 -6000.,.18
 -4500.,.33
 -3000.,.79
 -2000.,.91
 0.,1.
*SORPTION,TYPE=SCANNING
 9.45E6,
*GEL
 .0005,.0015,1.E8,500.
*INITIAL CONDITIONS,TYPE=SATURATION
 ALLN,.05
```

```
*NSET,NSET=PORN,GEN
 1,101,10
 3,103,10
*INITIAL CONDITIONS,TYPE=PORE
 PORN,-10000.
*INITIAL CONDITIONS,TYPE=RATIO
 ALLN,5.
*BOUNDARY
 BOT,2
 ALLN,1
*RESTART,WRITE,FREQ=10
*STEP
*SOILS,CONSOLIDATION
 1.E-7,1.E-7
*DLOAD
 10,P3,-500.
*CONTROLS,ANAL=DISC
*FILE FORMAT, ASCII
*EL FILE
 FLUVR
*END STEP
```

```
type('45.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS EQUIVALENT PLASTIC STRAIN COMPONENTS OUTPUT TO MATLAB (PEQC, RECORD KEY 45)
*RESTART,WRITE,FREQ=5
*NODE, NSET=BOT
1,0.,0.,0.
2,1.,0.,0.
3,1.,1.,0.
4,0.,1.,0.
*NODE, NSET=TOP
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*NSET, NSET=BACK
1,4,5,8
*NSET, NSET=LHS
1,2,5,6
*ELEMENT, TYPE=C3D8R, ELSET=EL1
1,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL1, MATERIAL=SCP3
*MATERIAL,NAME=SCP3
*ELASTIC
12857.1429, 0.28571429
*CAP PLASTICITY
173.20508, 30.0, 0.61858957, 0.027, 0.69258232, 1.0
*CAP HARDENING
 213.0, 0.00
 222.0, 0.01
 242.0, 0.02
 282.0, 0.03
 362.0, 0.04
 522.0, 0.05
 842.0, 0.06
1482.0, 0.07
2762.0, 0.08
*ELEMENT, TYPE=C3D8R, ELSET=EL2
101,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL2, MATERIAL=ELAS
*MATERIAL,NAME=ELAS
*ELASTIC
128.571429, 0.28571429
*STEP, INC=10, UNSYMM=YES
*STATIC, DIRECT
1., 10.
*BOUNDARY
BACK,1
LHS, 2
BOT, 3
*DLOAD
EL1, P4, -300.0
*FILE FORMAT, ASCII
*EL FILE,FREQ=10
PEQC
*END STEP
```

```
type('61.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT STATUS OUTPUT TO MATLAB (STATUS, RECORD KEY 61)
*PART,NAME=PART-1
*NODE
 1,  0.0,  0.0
 2, 0.25,  0.0
 3, 0.25, 0.25
 4,  0.0, 0.25
*NSET, NSET=BOT
 1,2
*NSET, NSET=TOP
 3,4
*ELEMENT,TYPE=CPS4R
 1,1,2,3,4
*ELSET, ELSET=EA
 1
*SOLID SECTION,ELSET=EA,MATERIAL=GLASS_EPOXY,ORIENT=RECT
 1.,
*NSET,NSET=LEFT
 1,4
*NSET, NSET=RIGHT
 2,3
*ORIENTATION,NAME=RECT
 1.0, 0.0, 0.0, 0.0, 1.0, 0.0
 3,0.0
*END PART
*ASSEMBLY,NAME=ASSEMBLY-1
*INSTANCE,NAME=PART-1-1,PART=PART-1
*END INSTANCE
*END ASSEMBLY
*MATERIAL,NAME=GLASS_EPOXY
*ELASTIC,TYPE=LAMINA
 53.8E9,17.9E9,0.25,8.96E9,8.96E9,6.88E9
*DAMAGE INITIATION,CRITERION=HASHIN,ALPHA=0.0
 1034E6,1034E6,27.6E6,138E6,41.4E6,50E6
*DAMAGE EVOLUTION, TYPE=ENERGY, SOFTENING=LINEAR
 10E6,10E6,5E6,5E6
*DAMAGE STABILIZATION
 0.0001, 0.0001, 0.0001, 0.0001
*STEP,INC=200,NLGEOM
*STATIC
 0.001,0.01,,0.001
*BOUNDARY
 ASSEMBLY-1.PART-1-1.RIGHT,1,1,0.001
 ASSEMBLY-1.PART-1-1.LEFT,1,1
 ASSEMBLY-1.PART-1-1.BOT, 2,2
*FILE FORMAT,ASCII
*EL FILE
 STATUS
*END STEP
```

```
type('78.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS WHOLE ELEMENT VOLUME OUTPUT TO MATLAB (EVOL, RECORD KEY 78)
*PREPRINT, ECHO=NO, MODEL=NO, HISTORY=NO, CONTACT=NO
*PART, NAME=PART-1
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-1-1, PART=PART-1
*NODE
      1,           0.,           0.
      2,           1.,           0.
      3,           1.,           1.
      4,           0.,           1.
      5,          0.5,           0.
      6,           1.,          0.5
      7,          0.5,           1.
      8,           0.,          0.5
*ELEMENT, TYPE=CPE8
 1,   1,   2,   3,   4,   5,   6,   7,   8
*ELSET, ELSET=_PICKEDSET2, INTERNAL
 1
*NSET, NSET=EDGE
 1,   2,   5
*NSET, NSET=LOAD
 3
*SOLID SECTION, ELSET=_PICKEDSET2, MATERIAL=MATERIAL-1
,
*END INSTANCE
*END ASSEMBLY
*MATERIAL, NAME=MATERIAL-1
*ELASTIC
 200.E9, 0.3
*STEP, NAME=STEP-1, NLGEOM=YES
*STATIC, DIRECT
 0.25, 1.
*CLOAD
 ASSEMBLY.PART-1-1.LOAD, 1, 100000
*BOUNDARY
 ASSEMBLY.PART-1-1.EDGE,1,2
*RESTART, WRITE, FREQUENCY=0
*FILE FORMAT, ASCII
*EL FILE
 EVOL
*END STEP
```

```
type('83.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS AVERAGE SHELL SECTION STRESS OUTPUT TO MATLAB (SSAVG, RECORD KEY 83)
*NODE
 1, 0.0, 0.0
 2, 2.0, 0.0
 3, 2.0, 1.0
 4, 0.0, 1.0
 11, 0.0, 0.0,1.0
 12, 2.0, 0.0,1.0
 13, 2.0, 1.0,1.0
 14, 0.0, 1.0,1.0
*ELEMENT,TYPE=SC8R, ELSET=EALL
 1, 1,2,3,4, 11,12,13,14
*SHELL SECTION,MATERIAL=A1, ELSET=EALL, POISSON=ELASTIC
 1.0,
*MATERIAL,NAME=A1
*ELASTIC,TYPE=ISOTROPIC
 30.0E6,0.3
*BOUNDARY
 1,1,3
 2,2,3
 3,3
 4,3
 11,1,2
 12,2,2
*STEP
*STATIC
*CLOAD
 2,1, 250.0
 3,1, -750.0
 3,2, -750.0
 4,1,-250.0
 4,2,-250.0
 12,1, 250.0
 13,1, -750.0
 13,2, -750.0
 14,1,-250.0
 14,2,-250.0
*FILE FORMAT, ASCII
*EL FILE
 SSAVG
*END STEP
```

```
type('88.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS THERMAL STRAINS OUTPUT TO MATLAB (THE, RECORD KEY 88)
*PREPRINT,ECHO=YES,HISTORY=NO,MODEL=NO
*NODE
  1, 0.  ,  0.
  5, 0.5 ,  0.
 21, 0.  ,  0.5
 25, 0.5 ,  0.5
 61, 0.  ,  1.
 69, 1.  ,  0.
 81, 0.  ,  2.
 89, 2.  ,  0.
151, 0.  ,  9.
159, 9.  ,  0.
161, 0.  ,  10.
163, 4.14,  10.
165, 10. ,  10.
167, 10. ,  4.14
169, 10. ,  0.
*NGEN,NSET=SMBOT
1,5,2
*NGEN,NSET=SMTOP
21,25,2
*NFILL,NSET=SMBOX
SMBOT,SMTOP,2,10
*NGEN,NSET=CIRC61,LINE=C
61,69,2
*NGEN,NSET=CIRC81,LINE=C
81,89,2
*NGEN,NSET=CIRC151,LINE=C
151,159,2
*NFILL,NSET=CIRCS
CIRC81,CIRC151,7,10
*NGEN
 145,165,10
*ELEMENT,TYPE=CPS3, ELSET=EALL
  1, 1,13,11
  2, 1, 3,13
 10,21,63,61
 11,21,23,63
 12,23,65,63
 13,23,25,65
 14,25,67,65
 15,25,15,67
 16,15,69,67
 17,15, 5,69
 30,81,103,101
 31,81,83,103
*ELGEN, ELSET=EALL
  1,2,2,2,2,10,4
  2,2,2,2,2,10,4
 30,4,2,2,4,20,10
 31,4,2,2,4,20,10
*NODE
 200, 0., 1.
```

```
 208, 1., 0.
*NGEN,NSET=OUTER,LINE=C
 200,208,2
*NSET,NSET=INNER
 CIRC61,
*ELEMENT,TYPE=CPS3, ELSET=EALL
 20,200,83,81
 21,200,202,83
*ELGEN, ELSET=EALL
 20,4,2,2
 21,4,2,2
*NSET,NSET=HOT,GENERATE
 1,25
 41,46
 51,59
 61,69
*NSET,NSET=SYMX1,GENERATE
 1,5
 59,169,10
*NSET,NSET=SYMX
 208,SYMX1
*NSET,NSET=SYMY1,GENERATE
 1,21,5
 51,161,10
*NSET,NSET=SYMY
 200,SYMY1
*ELSET,ELSET=SPOT,GENERATE
  1,8
 10,17
*ELSET,ELSET=PRINT
 16,17
 26,27
*EQUATION
2,
INNER,1,1.,OUTER,1,-1.
*EQUATION
2,
INNER,2,1.,OUTER,2,-1.
*MATERIAL,NAME=A1
*ELASTIC
 100.E9,0.3
*EXPANSION
 1.E-05,
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*BOUNDARY
 SYMX,2
 SYMY,1
*STEP
*STATIC
*TEMPERATURE
 HOT,100.
*FILE FORMAT, ASCII
*EL FILE,ELSET=PRINT,POSITION=AVERAGED AT NODES
 THE
*END STEP
```

```
type('89.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS LOGARITHMIC STRAIN OUTPUT TO MATLAB (LE, RECORD KEY 89)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      0,       0,       0
 2,      1,       0,       0
 3,      1,       1,       0
 4,      0,       1,       0
 5,      0,       0,       1
 6,      1,       0,       1
 7,      1,       1,       1
 8,      0,       1,       1
*ELEMENT, TYPE=C3D8
 1, 1, 2, 3, 4, 5, 6, 7, 8
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
 6,5
 7,6
 8,7
 9,8
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 1,2,3,4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5,6,7,8
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 100000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1, NLGEOM=YES
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 3, -100.
*FILE FORMAT, ASCII
*EL FILE
 LE
*END STEP
```

```
type('90.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NOMINAL STRAIN OUTPUT TO MATLAB (NE, RECORD KEY 89)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      0,       0,       0
 2,      1,       0,       0
 3,      1,       1,       0
 4,      0,       1,       0
 5,      0,       0,       1
 6,      1,       0,       1
 7,      1,       1,       1
 8,      0,       1,       1
*ELEMENT, TYPE=C3D8
 1, 1, 2, 3, 4, 5, 6, 7, 8
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*SOLID SECTION, ELSET=_PICKEDSET2_#1, MATERIAL=MAT1
 1,
*ELEMENT,TYPE=MASS,ELSET=MASSES
 2,1
 3,2
 4,3
 5,4
 6,5
 7,6
 8,7
 9,8
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 1,2,3,4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5,6,7,8
*END ASSEMBLY
*MATERIAL,NAME=MAT1
*ELASTIC
 100000, 0.3
*DENSITY
 1
*STEP, NAME=STEP-1, NLGEOM=YES
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 3, -100.
*FILE FORMAT, ASCII
*EL FILE
 NE
*END STEP
```

```
type('91.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS MECHANICAL STRAIN RATE OUTPUT TO MATLAB (ER, RECORD KEY 89)
*NODE,NSET=ALLN
 1,0.,0.,0.
 2,1.,0.,0.
 3,1.,1.,0.
 4,0.,1.,0.
 5,0.,0.,1.
 6,1.,0.,1.
 7,1.,1.,1.
 8,0.,1.,1.
*ELEMENT,TYPE=C3D8,ELSET=ALLE
 1,1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=ALLE
*MATERIAL,NAME=ALLE
*ELASTIC
 200.E3,.3
*PLASTIC
 200.,0.
 220.,.0009
 220.,.0029
*RATE DEPENDENT
 40.,5.
*BOUNDARY
 1,PINNED
 2,2
 5,2
 6,2
 4,1
 5,1
 8,1
 2,3
 3,3
 4,3
*STEP,INC=20
*STATIC,DIRECT
 1.E-3,20.E-3
*BOUNDARY
 7,3,,.004
 5,3,,.004
 6,3,,.004
 8,3,,.004
*FILE FORMAT, ASCII
*EL FILE
 ER
*END STEP
```

```
type('97.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PORE FLUID EFFECTIVE VELOCITY VECTOR OUTPUT TO MATLAB (FLVEL, RECORD KEY 89)
*PART,NAME=PART2
*NODE,NSET=SOIL
 1, 0.0, 0.0
 2, 1.0, 0.0
 3, 1.0, 1.0
 4, 0.0, 1.0
 5, 0.5, 0.0
 6, 1.0, 0.5
 7, 0.5, 1.0
 8, 0.0, 0.5
 9, 2.0, 0.0
 10, 3.0, 0.0
 11, 3.0, 1.0
 12, 2.0, 1.0
 13, 2.5, 0.0
 14, 3.0, 0.5
 15, 2.5, 1.0
 16, 2.0, 0.5
*NSET,NSET=TOP
 3,4,7, 11,12,15
*NSET,NSET=BOTTOM
 1,2,5, 9,10,13
*NSET,NSET=LEFT
 1,4,8, 9,12,16
*ELEMENT,TYPE=CPE8P,ELSET=SOIL
 1, 1,2,3,4, 5,6,7,8
*ELEMENT,TYPE=CPE8RP,ELSET=SOIL
 2, 9,10,11,12, 13,14,15,16
*END PART
*PART,NAME=PART4
*NODE,NSET=SOIL
 1, 0.0, 0.0
 2, 1.0, 0.0
 3, 1.0, 1.0
 4, 0.0, 1.0
 5, 0.5, 0.0
 6, 1.0, 0.5
 7, 0.5, 1.0
 8, 0.0, 0.5
 9, 2.0, 0.0
 10, 3.0, 0.0
 11, 3.0, 1.0
 12, 2.0, 1.0
 13, 2.5, 0.0
 14, 3.0, 0.5
 15, 2.5, 1.0
 16, 2.0, 0.5
 17, 4.0, 0.0
 18, 5.0, 0.0
 19, 5.0, 1.0
 20, 4.0, 1.0
 21, 4.5, 0.0
 22, 5.0, 0.5
```

```
 23, 4.5, 1.0
 24, 4.0, 0.5
 25, 6.0, 0.0
 26, 7.0, 0.0
 27, 7.0, 1.0
 28, 6.0, 1.0
 29, 6.5, 0.0
 30, 7.0, 0.5
 31, 6.5, 1.0
 32, 6.0, 0.5
*NSET,NSET=TOP
 3,4,7, 11,12,15, 19,20,23, 27,28,31
*NSET,NSET=BOTTOM
 1,2,5, 9,10,13, 17,18,21, 25,26,29
*NSET,NSET=LEFT
 1,4,8, 9,12,16, 17,20,24, 25,28,32
*ELEMENT,TYPE=CPE8P,ELSET=SOIL
 1, 1,2,3,4, 5,6,7,8
*ELEMENT,TYPE=CPE8RP,ELSET=SOIL
 2, 9,10,11,12, 13,14,15,16
*ELEMENT,TYPE=CPE8P,ELSET=SOIL
 3, 17,18,19,20, 21,22,23,24
*ELEMENT,TYPE=CPE8RP,ELSET=SOIL
 4, 25,26,27,28, 29,30,31,32
*END PART
*ASSEMBLY,NAME=GEOASSEMBLY
*INSTANCE,NAME=IELA,PART=PART4
 0,0,0
*SOLID SECTION,MATERIAL=MATELA, ELSET=SOIL
*END INSTANCE
*INSTANCE,NAME=IPOR,PART=PART2
 0,3,0
*SOLID SECTION,MATERIAL=MATPOR, ELSET=SOIL
*END INSTANCE
*END ASSEMBLY
*NSET,NSET=TOP
 IELA.TOP,IPOR.TOP
*NSET,NSET=BOTTOM
 IELA.BOTTOM,IPOR.BOTTOM
*NSET,NSET=LEFT
 IELA.LEFT,IPOR.LEFT
*NSET,NSET=SOIL
 IELA.SOIL,IPOR.SOIL
*ELSET,ELSET=SOIL
 IELA.SOIL,IPOR.SOIL
*MATERIAL,NAME=MATELA
*ELASTIC
 1000.0,0.3
*PERMEABILITY,SPECIFIC=10.0
 1.0,0.0
*MATERIAL,NAME=MATPOR
*POROUS ELASTIC
 .026,.3,100.0
*PERMEABILITY,SPECIFIC=10.0
 1.0,0.0
*INITIAL CONDITIONS,TYPE=RATIO
 SOIL,1.08,0.,1.08,1.
*INITIAL CONDITIONS,TYPE=STRESS
 SOIL,-10.,-10.,-10.
*BOUNDARY
 LEFT,   1,1
```

```
 BOTTOM, 2,2
*STEP,UNSYM=YES
 GEOSTATIC INITIAL STRESS STATE
*GEOSTATIC,UTOL
 0.5,1.0
*DLOAD
 SOIL,P2,100.
*BOUNDARY
 BOTTOM, 8,8
 TOP,    8,8, 100.0
*FILE FORMAT, ASCII
*EL FILE
 FLVEL
*END STEP
```

```
type('101.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL DISPLACEMENT OUTPUT TO MATLAB (U, RECORD KEY 101)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 U
*END STEP
```

```
type('102.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL VELOCITY OUTPUT TO MATLAB (V, RECORD KEY 102)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 V
*END STEP
```

Published with MATLAB® R2015a

```
type('103.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL ACCELERATION OUTPUT TO MATLAB (A, RECORD KEY 103)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 A
*END STEP
```

Published with MATLAB® R2015a

```
type('104.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL REACTION FORCE OUTPUT TO MATLAB (RF, RECORD KEY 104)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,      0,      360
 6,      0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 RF
*END STEP
```

```
type('105.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELECTRICAL POTENTIAL OUTPUT TO MATLAB (EPOT, RECORD KEY 105)
*NODE
 1, 0.0, 0.0, 0.0
 2, 2.0, 0.0, 0.0
 3, 2.0, 2.0, 0.0
 4, 0.0, 2.0, 0.0
 5, 0.0, 0.0, 1.0
 6, 2.0, 0.0, 1.0
 7, 2.0, 2.0, 1.0
 8, 0.0, 2.0, 1.0
*ELEMENT,TYPE=C3D8E, ELSET=EALL
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC,TYPE=ISOTROPIC
 30.0E6,0.3
*PIEZOELECTRIC,TYPE=S
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.
*DIELECTRIC,TYPE=ISO
 1.0E-3,
*BOUNDARY
 1,1,3
 2,2
 4,3
 5,1
 1,9
 2,9
 3,9
 4,9
*SURFACE, NAME=SIDE1485
 1,S6
*STEP,PERTURBATION
*STATIC
*DLOAD
 1, P1, 1000.0
 1, P2, 1000.0
 1, P3, 1000.0
 1, P4, 1000.0
 1, P5, 1000.0
 1, P6, 1000.0
*DECHARGE
 1, ES1, 1000.0
 1, ES2, 1000.0
 1, ES3, 1000.0
 1, ES4, 1000.0
 1, ES5, 1000.0
 1, ES6, 1000.0
*CLOAD
 2,1,  1500.00
 3,1,   500.00
 3,2,   500.00
 3,3, -1000.00
```

```
 4,1,   500.00
 4,2,  1500.00
 5,2,  -500.00
 5,3,  1000.00
 6,1,  -500.00
 6,2, -1500.00
 7,1, -1500.00
 7,2, -1500.00
 7,3, -1000.00
 8,1, -1500.00
 8,2,  -500.00
*CECHARGE
 1,,-2000.
 2,,-2000.
 3,,-2000.
 4,,-2000.
 5,,-1000.
 6,,-1000.
 7,,-1000.
 8,,-1000.
*FILE FORMAT, ASCII
*NODE FILE
 EPOT
*END STEP
```

```
type('106.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL POINT LOAD OUTPUT TO MATLAB (CF, RECORD KEY 106)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,        0,    360
 6,        0,    0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 CF
*END STEP
```

Published with MATLAB® R2015a

```
type('107.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL COORDINATE OUTPUT TO MATLAB (COORD, RECORD KEY 107)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 COORD
*END STEP
```

Published with MATLAB® R2015a

```
type('108.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PORE OR ACOUSTIC PRESSURE OUTPUT TO MATLAB (POR, RECORD KEY 108)
*NODE
 1, 1.
 3, 4.
 7, 1., 5.
 9, 4., 5.
*NGEN, NSET=SIDE1
 1, 3
*NGEN, NSET=SIDE3
 7, 9
*NGEN, NSET=SIDE4
 1, 7, 3
*NGEN, NSET=SIDE2
 3, 9, 3
*NSET, NSET=NALL, GENERATE
 1, 9
*NSET,NSET=CORNERS1
 1,3
*NSET,NSET=CORNERS2
 3,9
*NSET,NSET=CORNERS3
 7,9
*NSET,NSET=CORNERS4
 1,7
*NSET,NSET=CORNERS
 1,3,7,9
*ELEMENT,TYPE=CPE4P, ELSET=EALL
 1, 1,3,9,7
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC
 1.E8,
*PERMEABILITY,SPECIFIC=1.0
 1.E-5,
*DENSITY
 1.4142,
*INITIAL CONDITIONS,TYPE=RATIO
 NALL,1.
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS3, 8
*DLOAD, OP=NEW
*FLOW, OP=NEW
 1, Q1, 14.7, 2.E-5
*FILE FORMAT, ASCII
*NODE FILE
 POR
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
```

```
*BOUNDARY,OP=NEW
 NALL, 1,2
 CORNERS4, 8
*FLOW, OP=NEW
 1, Q2, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS1, 8
*FLOW, OP=NEW
 1, Q3, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS2, 8
*FLOW, OP=NEW
 1, Q4, 14.7, 2.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS3, 8
*FLOW, OP=NEW
*DFLOW, OP=NEW
 1, S1, 3.E-5
*END STEP
*STEP
*SOILS,CONSOLIDATION
 1., 1.
*BOUNDARY, OP=NEW
 NALL, 1,2
 CORNERS4, 8
*DFLOW, OP=NEW
 1, S2, 3.E-5
*END STEP
```

```
type('109.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS REACTIVE FLUID VOLUME FLUX OUTPUT TO MATLAB (RVF, RECORD KEY 109)
*NODE,NSET=ALL
 1    , 0.        , 0.        , 0.
 2    , 7.        , 0.        , 0.
 3    , 7.        , 0.        , -7.
 4    , 0.        , 0.        , -7.
 5    , 0.        , 7.        , 0.
 6    , 7.        , 7.        , 0.
 7    , 7.        , 7.        , -7.
 8    , 0.        , 7.        , -7.
*NSET,NSET=NS1
 1,4,5,8
*NSET,NSET=NS2
 1,2,3,4
*NSET,NSET=NS3
 1,2,5,6
*NSET,NSET=NS4
 2,3,7,6
*ELEMENT,TYPE=C3D8P, ELSET=EALL
 1,  1,2,3,4,5,6,7,8
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC
 3.E6,0.3
*DENSITY
 10.0,
*EXPANSION
 .0001,
*SPECIFIC HEAT
 1.0
*CONDUCTIVITY
 0.1
*DENSITY, PORE FLUID
 10.0,
*EXPANSION, PORE FLUID
 .0001,
*SPECIFIC HEAT, PORE FLUID
 1.0
*CONDUCTIVITY, PORE FLUID
 0.1
*PERMEABILITY, SPECIFIC=1.0
 0.01
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALL, 0.
*INITIAL CONDITIONS,TYPE=RATIO
 ALL,1.
*INITIAL CONDITIONS,TYPE=PORE
 ALL,0.
*BOUNDARY
 ALL,1,3
*STEP
*SOILS
 1.0,1.0
*DLOAD, OP=NEW
```

```
 1,GRAV,300.0,1.0,0.0,0.0
*BOUNDARY, OP=NEW
 ALL, 1, 3,
 ALL, 8, 8
 ALL, 11, 11
*FILE FORMAT, ASCII
*NODE FILE
 RVF
*END STEP
```

```
type('110.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS REACTIVE FLUID TOTAL VOLUME OUTPUT TO MATLAB (RVT, RECORD KEY 110)
*NODE
1,2.,0.
9,6.,0.
21,2.,6.
29,6.,6.
*NGEN,NSET=BOT1
1,9
*NGEN,NSET=TOP1
21,29
*NFILL,NSET=P001
BOT1,TOP1,2,10
*ELEMENT,TYPE=CAXA8P1,OFFSET=1200,ELSET=ALL
1,1,5,25,21,3,15,23,11
*ELGEN,ELSET=ALL
1,2,4,1
*SOLID SECTION,ELSET=ALL,MATERIAL=MATPROP
*MATERIAL,NAME=MATPROP
*ELASTIC
 1.E8,0.3
*PERMEABILITY,SPECIFIC=1.0
 1.E-5, 1.
*INITIAL CONDITIONS,TYPE=RATIO
NALL,1.
*NSET,NSET=NALL,GEN
1,29
1201,1229
*NSET,NSET=BOTTOM,GEN
1,9
1201,1209
*NSET,NSET=L1-000
1,21
*NSET,NSET=L1-180
1201,1221
*NSET,NSET=L3-000
9,29
*NSET,NSET=L3-180
1209,1229
*NSET,NSET=L1
L1-000,L1-180
*NSET,NSET=L3
L3-000,L3-180
*NSET,NSET=NFILE
NALL,
*ELSET,ELSET=EFILE
1,2
*BOUNDARY
1,1
BOTTOM,2,2
*RESTART,WRITE,FREQ=999
*STEP
*SOILS
 1., 1.
*BOUNDARY
```

```
L1-000,8,,1.E6
L1-180,8,,-1.E6
L3-000,8,,3.E6
L3-180,8,,-3.E6
*FILE FORMAT, ASCII
*NODE FILE,NSET=NFILE
RVT
*END STEP
```

Published with MATLAB® R2015a

```
type('119.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELECTRICAL REACTION CHARGE OUTPUT TO MATLAB (RCHG, RECORD KEY 119)
*NODE
 1, 0.0, 0.0, 0.0
 2, 2.0, 0.0, 0.0
 3, 2.0, 2.0, 0.0
 4, 0.0, 2.0, 0.0
 5, 0.0, 0.0, 1.0
 6, 2.0, 0.0, 1.0
 7, 2.0, 2.0, 1.0
 8, 0.0, 2.0, 1.0
*ELEMENT,TYPE=C3D8E, ELSET=EALL
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC,TYPE=ISOTROPIC
 30.0E6,0.3
*PIEZOELECTRIC,TYPE=S
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.
*DIELECTRIC,TYPE=ISO
 1.0E-3,
*BOUNDARY
 1,1,3
 2,2
 4,3
 5,1
 1,9
 2,9
 3,9
 4,9
*SURFACE, NAME=SIDE1485
 1,S6
*STEP,PERTURBATION
*STATIC
*DLOAD
 1, P1, 1000.0
 1, P2, 1000.0
 1, P3, 1000.0
 1, P4, 1000.0
 1, P5, 1000.0
 1, P6, 1000.0
*DECHARGE
 1, ES1, 1000.0
 1, ES2, 1000.0
 1, ES3, 1000.0
 1, ES4, 1000.0
 1, ES5, 1000.0
 1, ES6, 1000.0
*CLOAD
 2,1,  1500.00
 3,1,   500.00
 3,2,   500.00
 3,3, -1000.00
```

```
 4,1,   500.00
 4,2,  1500.00
 5,2,  -500.00
 5,3,  1000.00
 6,1,  -500.00
 6,2, -1500.00
 7,1, -1500.00
 7,2, -1500.00
 7,3, -1000.00
 8,1, -1500.00
 8,2,  -500.00
*CECHARGE
 1,,-2000.
 2,,-2000.
 3,,-2000.
 4,,-2000.
 5,,-1000.
 6,,-1000.
 7,,-1000.
 8,,-1000.
*FILE FORMAT, ASCII
*NODE FILE
 RCHG
*END STEP
```

```
type('120.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS CONCENTRATED ELECTRICAL NODAL CHARGE OUTPUT TO MATLAB (CECHG, RECORD KEY
119)
*NODE
 1, 0.0, 0.0, 0.0
 2, 2.0, 0.0, 0.0
 3, 2.0, 2.0, 0.0
 4, 0.0, 2.0, 0.0
 5, 0.0, 0.0, 1.0
 6, 2.0, 0.0, 1.0
 7, 2.0, 2.0, 1.0
 8, 0.0, 2.0, 1.0
*ELEMENT,TYPE=C3D8E, ELSET=EALL
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC,TYPE=ISOTROPIC
 30.0E6,0.3
*PIEZOELECTRIC,TYPE=S
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.,0.,0.,0.,0.,0.,0.
 0.,0.
*DIELECTRIC,TYPE=ISO
 1.0E-3,
*BOUNDARY
 1,1,3
 2,2
 4,3
 5,1
 1,9
 2,9
 3,9
 4,9
*SURFACE, NAME=SIDE1485
 1,S6
*STEP,PERTURBATION
*STATIC
*DLOAD
 1, P1, 1000.0
 1, P2, 1000.0
 1, P3, 1000.0
 1, P4, 1000.0
 1, P5, 1000.0
 1, P6, 1000.0
*DECHARGE
 1, ES1, 1000.0
 1, ES2, 1000.0
 1, ES3, 1000.0
 1, ES4, 1000.0
 1, ES5, 1000.0
 1, ES6, 1000.0
*CLOAD
 2,1,  1500.00
 3,1,   500.00
 3,2,   500.00
```

```
 3,3, -1000.00
 4,1,   500.00
 4,2,  1500.00
 5,2,  -500.00
 5,3,  1000.00
 6,1,  -500.00
 6,2, -1500.00
 7,1, -1500.00
 7,2, -1500.00
 7,3, -1000.00
 8,1, -1500.00
 8,2,  -500.00
*CECHARGE
 1,,-2000.
 2,,-2000.
 3,,-2000.
 4,,-2000.
 5,,-1000.
 6,,-1000.
 7,,-1000.
 8,,-1000.
*FILE FORMAT, ASCII
*NODE FILE
 CECHG
*END STEP
```

```
type('136.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS FLUID CAVITY PRESSURE OUTPUT TO MATLAB (PCAV, RECORD KEY 136)
*RESTART, WRITE, FREQ=1
*NODE, NSET=CAV1NOD
 1, 1., 1., 1.
 2, 2., 1., 1.
 3, 2., 2., 1.
 4, 1., 2., 1.
 5, 1., 1., 0.
 6, 2., 1., 0.
 7, 2., 2., 0.
 8, 1., 2., 0.
*ELEMENT,TYPE=SFM3D4,ELSET=STRUCTURE
 1, 2, 3, 7, 6
 2, 3, 4, 8, 7
 3, 6, 7, 8, 5
*SURFACE SECTION, ELSET=STRUCTURE
*SURFACE,TYPE=ELEMENT,NAME=CAV1
 STRUCTURE, SPOS
*PHYSICAL CONSTANTS, ABSOLUTE ZERO = 0
*FLUID CAVITY, NAME=CAVITY1, BEHAVIOR=FLUID, SURFACE=CAV1, REFNODE=1
 1.0
*FLUID BEHAVIOR, NAME=FLUID
*FLUID DENSITY
 10.,
*ELEMENT, TYPE=SPRING1, ELSET=SPRINGX1
 21, 2
*SPRING, ELSET=SPRINGX1
 1,
 400.,
*ELEMENT, TYPE=SPRING1, ELSET=SPRINGY1
 42, 4
*SPRING, ELSET=SPRINGY1
 2,
 1.E-6,
*BOUNDARY
 2, 2, 3, 0.0
 3, 3, 3, 0.0
 4, 1, 1, 0.0
 4, 3, 3, 0.0
 5, 1, 3, 0.0
 6, 2, 3, 0.0
 7, 3, 3, 0.0
 8, 1, 1, 0.0
 8, 3, 3, 0.0
*EQUATION
 2,
 3, 1, 1.0, 2, 1, -1.0
 2,
 6, 1, 1.0, 2, 1, -1.0
 2,
 7, 1, 1.0, 2, 1, -1.0
 2,
 3, 2, 1.0, 4, 2, -1.0
 2,
```

```
 7, 2, 1.0, 4, 2, -1.0
 2,
 8, 2, 1.0, 4, 2, -1.0
*NSET,NSET=REFNODE
 1,
*STEP, NLGEOM
*STATIC
 .2, 1.
*CLOAD
 4, 2, -600.
*FILE FORMAT, ASCII
*NODE FILE
 PCAV
*END STEP
```

```
type('137.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS FLUID CAVITY VOLUME OUTPUT TO MATLAB (CVOL, RECORD KEY 137)
*RESTART, WRITE, FREQ=1
*NODE, NSET=CAV1NOD
 1, 1., 1., 1.
 2, 2., 1., 1.
 3, 2., 2., 1.
 4, 1., 2., 1.
 5, 1., 1., 0.
 6, 2., 1., 0.
 7, 2., 2., 0.
 8, 1., 2., 0.
*ELEMENT,TYPE=SFM3D4,ELSET=STRUCTURE
 1, 2, 3, 7, 6
 2, 3, 4, 8, 7
 3, 6, 7, 8, 5
*SURFACE SECTION, ELSET=STRUCTURE
*SURFACE,TYPE=ELEMENT,NAME=CAV1
 STRUCTURE, SPOS
*PHYSICAL CONSTANTS, ABSOLUTE ZERO = 0
*FLUID CAVITY, NAME=CAVITY1, BEHAVIOR=FLUID, SURFACE=CAV1, REFNODE=1
 1.0
*FLUID BEHAVIOR, NAME=FLUID
*FLUID DENSITY
 10.,
*ELEMENT, TYPE=SPRING1, ELSET=SPRINGX1
 21, 2
*SPRING, ELSET=SPRINGX1
 1,
 400.,
*ELEMENT, TYPE=SPRING1, ELSET=SPRINGY1
 42, 4
*SPRING, ELSET=SPRINGY1
 2,
 1.E-6,
*BOUNDARY
 2, 2, 3, 0.0
 3, 3, 3, 0.0
 4, 1, 1, 0.0
 4, 3, 3, 0.0
 5, 1, 3, 0.0
 6, 2, 3, 0.0
 7, 3, 3, 0.0
 8, 1, 1, 0.0
 8, 3, 3, 0.0
*EQUATION
 2,
 3, 1, 1.0, 2, 1, -1.0
 2,
 6, 1, 1.0, 2, 1, -1.0
 2,
 7, 1, 1.0, 2, 1, -1.0
 2,
 3, 2, 1.0, 4, 2, -1.0
 2,
```

```
 7, 2, 1.0, 4, 2, -1.0
 2,
 8, 2, 1.0, 4, 2, -1.0
*NSET,NSET=REFNODE
 1,
*STEP, NLGEOM
*STATIC
 .2, 1.
*CLOAD
 4, 2, -600.
*FILE FORMAT, ASCII
*NODE FILE
 CVOL
*END STEP
```

Published with MATLAB® R2015a

```
type('138.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELECTRICAL REACTION CURRENT OUTPUT TO MATLAB (RECUR, RECORD KEY 138)
*RESTART,WRITE
*NODE
 900000001,   0.0, 0.0
 900000010,   1.0, 0.0
*NGEN,NSET=NALL
 900000001,900000010
*NSET,NSET=EDGE1
 900000001
*NSET,NSET=EDGE2
 900000010
*ELEMENT,TYPE=DC1D2E, ELSET=EALL
 900000001,900000001,900000002
*ELGEN,ELSET=EALL
 900000001,9
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
 0.1,
*MATERIAL,NAME=A1
*CONDUCTIVITY
 45.0,
*ELECTRICAL CONDUCTIVITY
 6.58E6,
*JOULE HEAT FRACTION
 1.0,
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY
 EDGE1,  9, ,   0.0
 EDGE2,  9, ,   0.1
 EDGE2, 11, , 100.0
*FILE FORMAT, ASCII
*NODE FILE
 RECUR
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*BOUNDARY,OP=NEW
 NALL, 11, , 0.0
*END STEP
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY,OP=NEW
 EDGE1,  9, ,   0.0
 EDGE2, 11, , 100.0
*CECURRENT,OP=NEW
 EDGE2,  9, 6.58E4
*END STEP
```

```
type('139.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS CONCENTRATED ELECTRICAL NODAL CURRENT OUTPUT TO MATLAB (CECUR, RECORD KEY
139)
*RESTART,WRITE
*NODE
 900000001,   0.0, 0.0
 900000010,   1.0, 0.0
*NGEN,NSET=NALL
 900000001,900000010
*NSET,NSET=EDGE1
 900000001
*NSET,NSET=EDGE2
 900000010
*ELEMENT,TYPE=DC1D2E, ELSET=EALL
 900000001,900000001,900000002
*ELGEN,ELSET=EALL
 900000001,9
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
 0.1,
*MATERIAL,NAME=A1
*CONDUCTIVITY
 45.0,
*ELECTRICAL CONDUCTIVITY
 6.58E6,
*JOULE HEAT FRACTION
 1.0,
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY
 EDGE1,  9, ,   0.0
 EDGE2,  9, ,   0.1
 EDGE2, 11, , 100.0
*FILE FORMAT, ASCII
*NODE FILE
 CECUR
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*BOUNDARY,OP=NEW
 NALL, 11, , 0.0
*END STEP
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY,OP=NEW
 EDGE1,  9, ,   0.0
 EDGE2, 11, , 100.0
*CECURRENT,OP=NEW
 EDGE2,  9, 6.58E4
*END STEP
```

```
type('145.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS VISCOUS FORCES DUE TO STATIC STABILIZATION OUTPUT TO MATLAB (VF, RECORD KEY
145)
*NODE
 1,
 2,10.
 3,10.,5.,
 4,0.,5.,
*ELEMENT,TYPE=CPE4,ELSET=ONE
 1,1,2,3,4
*SOLID SECTION,ELSET=ONE,MATERIAL=SIMPLE
 1.,
*MATERIAL,NAME=SIMPLE
*HYPERFOAM,N=3,TEST DATA INPUT,POISSON=0.,MODULI=INSTANTANEOUS
*UNIAXIAL TEST DATA
 -39. ,   -.05
 -57. ,   -.10
 -66. ,   -.15
 -72. ,   -.20
 -78. ,   -.25
 -84. ,   -.30
 -90. ,   -.35
 -96. ,   -.40
 -102. ,   -.45
 -108. ,   -.50
 -115. ,   -.55
 -130. ,   -.60
 -150. ,   -.65
 -185. ,   -.70
 -260. ,   -.75
 -400. ,   -.80
*VISCOELASTIC,TIME=PRONY
 0.5,0.5,3.
*BOUNDARY
 1,1,2
 2,2,2
 4,1,1
*NSET,NSET=OUT
 2
*STEP,NLGEOM,INC=200
*VISCO,CETOL=.01,STABILIZE
 2.,10.,,10.,
*BOUNDARY
 3,1,1,2.
 2,1,1,2.
*FILE FORMAT, ASCII
*NODE FILE,NSET=OUT
 VF
*END STEP
```

```
type('146.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TOTAL FORCE OUTPUT TO MATLAB (TF, RECORD KEY 146)
*RESTART,WRITE
*NODE
 1,          0.,          0.
 2,          1.,          0.
 3,          0.,          1.
 4,          1.,          1.
 7,          0.,          2.
 8,          1.,          2.
*NODE,NSET=NREF
 1001,          0.,          0.
*ELEMENT, TYPE=CPE4R, ELSET=ECPE
 1,       1,       2,       4,       3
 2,       3,       4,       8,       7
*SOLID SECTION,ELSET=ECPE,MATERIAL=MAT
 1.0,
*MATERIAL,NAME=MAT
*ELASTIC
 2.1E11,0.3
*DENSITY
 7800.0,
*NSET, NSET=XSYMM
 1,       3,       7
*NSET, NSET=YSYMM
 1,       2
*NSET, NSET=NPULL
 7,       8
*SURFACE,NAME=SCPE
 1,S3
*PRE-TENSION SECTION, SURFACE=SCPE, NODE=1001
*STEP,NLGEOM
 PRE-TENSION SECTION BY TIGHTENING
*STATIC
 0.1,1.
*BOUNDARY
 XSYMM,1,,
 YSYMM,2,,
 NPULL,2,,
 NREF,1,,0.1
*END STEP
*STEP,NLGEOM,INC=500
 FURTHER LOADING FROM INITIAL TIGHTENED STATE
*STATIC
 0.1,1.
*BOUNDARY
 NPULL,2,,0.2
*FILE FORMAT, ASCII
*NODE FILE
 TF
*END STEP
```

Published with MATLAB® R2015a

```
type('201.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS TEMPERATURE OUTPUT TO MATLAB (NT, RECORD KEY 201)
*RESTART,WRITE
*NODE
 900000001,   0.0, 0.0
 900000010,   1.0, 0.0
*NGEN,NSET=NALL
 900000001,900000010
*NSET,NSET=EDGE1
 900000001
*NSET,NSET=EDGE2
 900000010
*ELEMENT,TYPE=DC1D2E, ELSET=EALL
 900000001,900000001,900000002
*ELGEN,ELSET=EALL
 900000001,9
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
 0.1,
*MATERIAL,NAME=A1
*CONDUCTIVITY
 45.0,
*ELECTRICAL CONDUCTIVITY
 6.58E6,
*JOULE HEAT FRACTION
 1.0,
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY
 EDGE1,  9, ,   0.0
 EDGE2,  9, ,   0.1
 EDGE2, 11, , 100.0
*FILE FORMAT, ASCII
*NODE FILE
 NT
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*BOUNDARY,OP=NEW
 NALL, 11, , 0.0
*END STEP
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY,OP=NEW
 EDGE1,  9, ,   0.0
 EDGE2, 11, , 100.0
*CECURRENT,OP=NEW
 EDGE2,  9, 6.58E4
*END STEP
```

```
type('204.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS RESIDUAL FLUX OUTPUT TO MATLAB (RFL, RECORD KEY 204)
*RESTART,WRITE
*NODE
 900000001,   0.0, 0.0
 900000010,   1.0, 0.0
*NGEN,NSET=NALL
 900000001,900000010
*NSET,NSET=EDGE1
 900000001
*NSET,NSET=EDGE2
 900000010
*ELEMENT,TYPE=DC1D2E, ELSET=EALL
 900000001,900000001,900000002
*ELGEN,ELSET=EALL
 900000001,9
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
 0.1,
*MATERIAL,NAME=A1
*CONDUCTIVITY
 45.0,
*ELECTRICAL CONDUCTIVITY
 6.58E6,
*JOULE HEAT FRACTION
 1.0,
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY
 EDGE1,  9, ,   0.0
 EDGE2,  9, ,   0.1
 EDGE2, 11, , 100.0
*FILE FORMAT, ASCII
*NODE FILE
 RFL
*END STEP
*STEP
*HEAT TRANSFER,STEADY STATE
*BOUNDARY,OP=NEW
 NALL, 11, , 0.0
*END STEP
*STEP
*COUPLED THERMAL-ELECTRICAL,STEADY STATE
*BOUNDARY,OP=NEW
 EDGE1,  9, ,   0.0
 EDGE2, 11, , 100.0
*CECURRENT,OP=NEW
 EDGE2,  9, 6.58E4
*END STEP
```

```
type('206.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS CONCENTRATED FLUX OUTPUT TO MATLAB (CFL, RECORD KEY 206)
*NODE,NSET=ALL
 900000001,0,0,0
 900000002,1,0,0
 900000003,1,1,0
 900000004,0,1,0
 900000005,0,0,2
 900000006,1,0,2
 900000007,1,1,2
 900000008,0,1,2
*ELEMENT,TYPE=C3D8T, ELSET=EALL
 900000001,900000001,900000002,900000003,900000004,900000005,900000006,900000007,900000008
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC
 1.0,0.0
*EXPANSION
 0.0
*CONDUCTIVITY
 1.0
*DENSITY
 1.0
*SPECIFIC HEAT
 1.0
*ELEMENT,TYPE=HEATCAP,ELSET=CAP
 900000101,900000001
 900000102,900000002
 900000103,900000003
 900000104,900000004
 900000105,900000005
 900000106,900000006
 900000107,900000007
 900000108,900000008
*HEATCAP,ELSET=CAP
 0.125
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALL,100
*BOUNDARY
 ALL,1,3,0.0
*AMPLITUDE,NAME=CONSTANT_FILM,VALUE=ABSOLUTE
 0.,1.0,10.0,1.0
*STEP,INC=100
 COOL DOWN BY CONVECTION
*COUPLED TEMPERATURE-DISPLACEMENT,DELTMX=2.5
 0.025,10
*CFILM,FILM AMPLITUDE=CONSTANT_FILM
 900000001,0.25,20,.0
 900000002,0.25,20,.0
 900000003,0.25,20,.0
 900000004,0.25,20,.0
 900000005,0.25,20,.0
 900000006,0.25,20,.0
 900000007,0.25,20,.0
 900000008,0.25,20,.0
```

```
*FILE FORMAT, ASCII
*NODE FILE
 CFL
*ENDSTEP
*STEP,INC=100
 HEATED UP BY PRESCRIBED FLUX
*COUPLED TEMPERATURE-DISPLACEMENT,DELTMX=2.5
 0.25,10
*CFILM,OP=NEW
*CFLUX,OP=NEW
 900000001,11,3.0
 900000002,11,3.0
 900000003,11,3.0
 900000004,11,3.0
 900000005,11,3.0
 900000006,11,3.0
 900000007,11,3.0
 900000008,11,3.0
*END STEP
```

```
type('214.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS INTERNAL FLUX OUTPUT TO MATLAB (RFLE, RECORD KEY 214)
*NODE,NSET=ALL
 900000001,0,0,0
 900000002,1,0,0
 900000003,1,1,0
 900000004,0,1,0
 900000005,0,0,2
 900000006,1,0,2
 900000007,1,1,2
 900000008,0,1,2
*ELEMENT,TYPE=C3D8T, ELSET=EALL
 900000001,900000001,900000002,900000003,900000004,900000005,900000006,900000007,900000008
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*MATERIAL,NAME=A1
*ELASTIC
 1.0,0.0
*EXPANSION
 0.0
*CONDUCTIVITY
 1.0
*DENSITY
 1.0
*SPECIFIC HEAT
 1.0
*ELEMENT,TYPE=HEATCAP,ELSET=CAP
 900000101,900000001
 900000102,900000002
 900000103,900000003
 900000104,900000004
 900000105,900000005
 900000106,900000006
 900000107,900000007
 900000108,900000008
*HEATCAP,ELSET=CAP
 0.125
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALL,100
*BOUNDARY
 ALL,1,3,0.0
*AMPLITUDE,NAME=CONSTANT_FILM,VALUE=ABSOLUTE
 0.,1.0,10.0,1.0
*STEP,INC=100
 COOL DOWN BY CONVECTION
*COUPLED TEMPERATURE-DISPLACEMENT,DELTMX=2.5
 0.025,10
*CFILM,FILM AMPLITUDE=CONSTANT_FILM
 900000001,0.25,20,.0
 900000002,0.25,20,.0
 900000003,0.25,20,.0
 900000004,0.25,20,.0
 900000005,0.25,20,.0
 900000006,0.25,20,.0
 900000007,0.25,20,.0
 900000008,0.25,20,.0
```

```
*FILE FORMAT, ASCII
*NODE FILE
 RFLE
*ENDSTEP
*STEP,INC=100
 HEATED UP BY PRESCRIBED FLUX
*COUPLED TEMPERATURE-DISPLACEMENT,DELTMX=2.5
 0.25,10
*CFILM,OP=NEW
*CFLUX,OP=NEW
 900000001,11,3.0
 900000002,11,3.0
 900000003,11,3.0
 900000004,11,3.0
 900000005,11,3.0
 900000006,11,3.0
 900000007,11,3.0
 900000008,11,3.0
*END STEP
```

```
type('221.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NORMALIZED CONCENTRATION OUTPUT TO MATLAB (NNC, RECORD KEY 221)
*NODE,NSET=ALL
 1, 0.
 2, 7.
 3, 7., 7.
 4, 0., 7.
*NSET,NSET=FIX1
 1,2
*NSET,NSET=FIX2
 2,3
*NSET,NSET=FIX3
 3,4
*NSET,NSET=FIX4
 1,4
*ELEMENT,TYPE=DC2D3, ELSET=EALL
 1,1,2,3
 2,1,3,4
*ORIENTATION,NAME=RECT
 1.0, 0.0, 0.0, 0.0, 1.0, 0.0
 1, 0.0
*SOLID SECTION,MATERIAL=A1, ELSET=EALL,ORIENT=RECT
*MATERIAL,NAME=A1
*DIFFUSIVITY,TYPE=ORTHO,LAW=GENERAL
 3.77E-5,7.54E-5,11.31E-5
*SOLUBILITY
 1.,
*BOUNDARY
 FIX1, 11
*STEP
*MASS DIFFUSION,STEADY STATE
*FILE FORMAT, ASCII
*NODE FILE
 NNC
*DFLUX,OP=NEW
 1, BF, .3
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX3,11
*DFLUX,OP=NEW
 1, S1, .3
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX4,11
*DFLUX,OP=NEW
 1, S2, .3
*NODE FILE
 NNC
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
```

```
*BOUNDARY,OP=NEW
 FIX1,11
*DFLUX,OP=NEW
 2, S2, .3
*NODE FILE
 NNC
*END STEP
*STEP
*MASS DIFFUSION, STEADY STATE
*BOUNDARY,OP=NEW
 FIX2,11
*DFLUX,OP=NEW
 2, S3, .3
*END STEP
```

```
type('237.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS MOTIONS (IN CAVITY RADIATION ANALYSIS) OUTPUT TO MATLAB (MOT, RECORD KEY
237)
*PHYSICAL CONSTANTS, STEFAN=5.669E-8,ABSOLUTE ZERO=0.0
*NODE
 1,0.,0.
 18,17.,0.
 421,0.,21.
 438,17.,21.
*NGEN,NSET=LHS
 1,421,20
*NGEN,NSET=RHS
 18,438,20
*NFILL,NSET=ALLN
 LHS,RHS,17
*NSET,NSET=LLHS,GEN
 1,401,40
 3,403,40
 5,405,40
 7,407,40
 9,409,40
 11,411,40
 13,413,40
 15,415,40
 17,417,40
*NSET,NSET=URHS,GEN
 22,422,40
 24,424,40
 26,426,40
 28,428,40
 30,430,40
 32,432,40
 34,434,40
 36,436,40
 38,438,40
*NSET,NSET=OUTN,GEN
 209,210
 229,230
*ELEMENT,TYPE=DC2D4,ELSET=CONDEL
 1,1,2,22,21
*ELGEN,ELSET=CONDEL
 1,9,2,1,11,40,10
*ELSET,ELSET=OUTEL,GEN
 55,55
*SOLID SECTION,ELSET=CONDEL,MATERIAL=ALUM
*MATERIAL,NAME=ALUM
*CONDUCTIVITY
 204.,
*DENSITY
 2707.,
*SPECIFIC HEAT
 896.,
*SURFACE,NAME=SALL,PROPERTY=PALL
 CONDEL,S1
 CONDEL,S2
```

```
 CONDEL,S3
 CONDEL,S4
*CAVITY DEFINITION,NAME=ARR2D,AMB=200.
 SALL,
*SURFACE PROPERTY,NAME=PALL
*EMISSIVITY
 0.7,
*INITIAL CONDITIONS,TYPE=TEMPERATURE
 ALLN,300.
*RESTART,WRITE,FREQ=10
*STEP,INC=20
*HEAT TRANSFER,STEADY STATE
 1.,1.
*BOUNDARY
 LLHS,11,,1000.
 URHS,11,,400.
*RADIATION VIEW,REFLECTION=NO
*FILE FORMAT, ASCII
*NODE FILE
 MOT
*VIEWFACTOR OUTPUT,CAVITY=ARR2D
*END STEP
```

```
type('401.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PRINCIPAL STRESSES OUTPUT TO MATLAB (SP, RECORD KEY 401)
*PREPRINT, ECHO=NO, MODEL=NO, HISTORY=NO, CONTACT=NO
*PART, NAME=PART-1
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-1-1, PART=PART-1
*NODE
      1,           0.,           0.
      2,           1.,           0.
      3,           1.,           1.
      4,           0.,           1.
      5,          0.5,           0.
      6,           1.,          0.5
      7,          0.5,           1.
      8,           0.,          0.5
*ELEMENT, TYPE=CPE8
 1,   1,   2,   3,   4,   5,   6,   7,   8
*ELSET, ELSET=_PICKEDSET2, INTERNAL
 1
*NSET, NSET=EDGE
 1,   2,   5
*NSET, NSET=LOAD
 3
*SOLID SECTION, ELSET=_PICKEDSET2, MATERIAL=MATERIAL-1
,
*END INSTANCE
*END ASSEMBLY
*MATERIAL, NAME=MATERIAL-1
*ELASTIC
 200.E9, 0.3
*STEP, NAME=STEP-1, NLGEOM=YES
*STATIC, DIRECT
 0.25, 1.
*CLOAD
 ASSEMBLY.PART-1-1.LOAD, 1, 100000
*BOUNDARY
 ASSEMBLY.PART-1-1.EDGE,1,2
*RESTART, WRITE, FREQUENCY=0
*FILE FORMAT, ASCII
*EL FILE
 SP
*END STEP
```

```
type('402.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL VALUES OF BACKSTRESS TENSOR FOR KINEMATIC HARDENING
PLASTICITY OUTPUT TO MATLAB (ALPHAP, RECORD KEY 402)
*RESTART,WRITE
*NODE, NSET=NALL
1,
2,1.,
*ELEMENT,TYPE=PIPE31,ELSET=EALL
1,1,2,
*BEAM SECTION,SECTION=PIPE,ELSET=EALL,MATERIAL=COPPER
1.01,0.02,
*MATERIAL,NAME=COPPER
*ELASTIC
104.E3,0.3
*PLASTIC,HARD=COMBINED,DATA TYPE=STABILIZED
 0.46940E+02, 0.0000
 0.52044E+02, 0.50927E-04
 0.56826E+02, 0.10494E-03
 0.61273E+02, 0.16218E-03
 0.65813E+02, 0.21853E-03
 0.69556E+02, 0.27253E-03
 0.72703E+02, 0.34228E-03
 0.76541E+02, 0.39538E-03
 0.79630E+02, 0.49568E-03
 0.82886E+02, 0.55437E-03
 0.85974E+02, 0.61468E-03
 0.88910E+02, 0.69644E-03
 0.91647E+02, 0.76012E-03
 0.94263E+02, 0.83498E-03
 0.96266E+02, 0.92571E-03
 0.99020E+02, 0.99923E-03
 0.10099E+03, 0.10803E-02
 0.10310E+03, 0.11600E-02
 0.10534E+03, 0.12385E-02
 0.10691E+03, 0.13233E-02
 0.10846E+03, 0.14085E-02
 0.11054E+03, 0.14785E-02
 0.11252E+03, 0.15795E-02
 0.11328E+03, 0.16621E-02
 0.11521E+03, 0.17336E-02
 0.11612E+03, 0.18248E-02
 0.11769E+03, 0.19197E-02
 0.11906E+03, 0.20065E-02
 0.12018E+03, 0.20858E-02
 0.12137E+03, 0.21743E-02
 0.12239E+03, 0.22746E-02
 0.12338E+03, 0.23550E-02
 0.12467E+03, 0.24426E-02
 0.12500E+03, 0.25494E-02
 0.12649E+03, 0.26351E-02
 0.12715E+03, 0.27288E-02
 0.12805E+03, 0.28101E-02
 0.12879E+03, 0.29130E-02
 0.12980E+03, 0.30033E-02
 0.13040E+03, 0.30875E-02
```

0.13049E+03, 0.31867E-02
0.13165E+03, 0.32855E-02
0.13191E+03, 0.33830E-02
0.13263E+03, 0.34660E-02
0.13324E+03, 0.35602E-02
0.13386E+03, 0.36642E-02
0.13398E+03, 0.37631E-02
0.13501E+03, 0.38632E-02
0.13513E+03, 0.39620E-02
0.13614E+03, 0.40423E-02
0.13623E+03, 0.41314E-02
0.13682E+03, 0.42358E-02
0.13691E+03, 0.43349E-02
0.13749E+03, 0.44193E-02
0.13738E+03, 0.45304E-02
0.13810E+03, 0.46234E-02
0.13859E+03, 0.47087E-02
0.13885E+03, 0.48062E-02
0.13930E+03, 0.49019E-02
0.13906E+03, 0.50042E-02
0.13991E+03, 0.50960E-02
0.13986E+03, 0.51965E-02
0.13979E+03, 0.53072E-02
0.14023E+03, 0.54030E-02
0.14061E+03, 0.54893E-02
0.14020E+03, 0.55933E-02
0.14118E+03, 0.56838E-02
0.14098E+03, 0.57958E-02
0.14218E+03, 0.58542E-02
0.14132E+03, 0.59825E-02
0.14153E+03, 0.60705E-02
0.14202E+03, 0.61658E-02
0.14215E+03, 0.62746E-02
0.14222E+03, 0.63738E-02
0.14177E+03, 0.64682E-02
0.14313E+03, 0.65651E-02
0.14205E+03, 0.66755E-02
0.14320E+03, 0.67744E-02
0.14245E+03, 0.68817E-02
0.14317E+03, 0.69447E-02
0.14294E+03, 0.70670E-02
0.14374E+03, 0.71492E-02
0.14306E+03, 0.72558E-02
0.14371E+03, 0.73596E-02
0.14361E+03, 0.74605E-02
0.14352E+03, 0.75514E-02
0.14353E+03, 0.76412E-02
0.14399E+03, 0.77468E-02
0.14427E+03, 0.78441E-02
0.14364E+03, 0.79502E-02
0.14501E+03, 0.80370E-02
0.14371E+03, 0.81495E-02
0.14478E+03, 0.82393E-02
0.14419E+03, 0.83449E-02
0.14451E+03, 0.84318E-02
0.14439E+03, 0.85430E-02
0.14511E+03, 0.86360E-02
0.14394E+03, 0.87473E-02
0.14519E+03, 0.88253E-02
0.14441E+03, 0.89427E-02
0.14546E+03, 0.90327E-02

```
 0.14455E+03, 0.91414E-02
 0.14560E+03, 0.92313E-02
 0.14484E+03, 0.93386E-02
 0.14582E+03, 0.94193E-02
 0.14515E+03, 0.95357E-02
 0.14491E+03, 0.96179E-02
 0.14558E+03, 0.97215E-02
 0.14533E+03, 0.98239E-02
 0.14613E+03, 0.99062E-02
 0.14477E+03, 0.10029E-01
 0.14638E+03, 0.10104E-01
 0.14557E+03, 0.10212E-01
 0.14609E+03, 0.10307E-01
 0.14571E+03, 0.10420E-01
 0.14599E+03, 0.10518E-01
 0.14546E+03, 0.10623E-01
 0.14617E+03, 0.10706E-01
 0.14574E+03, 0.10790E-01
 0.14644E+03, 0.10903E-01
 0.14546E+03, 0.11013E-01
 0.14632E+03, 0.11104E-01
 0.14608E+03, 0.11207E-01
 0.14615E+03, 0.11306E-01
 0.14643E+03, 0.11403E-01
 0.14565E+03, 0.11501E-01
 0.14700E+03, 0.11588E-01
 0.14585E+03, 0.11699E-01
 0.14714E+03, 0.11787E-01
 0.14537E+03, 0.11914E-01
*CYCLIC HARDENING
13.,0.000
13.,0.0068
29.37,.034
50.13,.0612
63.91,.0884
73.01,.1156
80.76,.1428
86.25,.1700
90.498,.1972
93.95,.2244
96.155,.2516
*BOUNDARY
1,1,6
*STEP,INC=10
*STATIC,DIRECT
0.1,1.,
*BOUNDARY
2,1,1,0.01
*FILE FORMAT, ASCII
*EL FILE
ALPHAP
*END STEP
```

```
type('403.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PRINCIPAL STRAINS OUTPUT TO MATLAB (EP, RECORD KEY 403)
*RESTART,WRITE
*NODE, NSET=NALL
1,
2,1.,
*ELEMENT,TYPE=PIPE31,ELSET=EALL
1,1,2,
*BEAM SECTION,SECTION=PIPE,ELSET=EALL,MATERIAL=COPPER
1.01,0.02,
*MATERIAL,NAME=COPPER
*ELASTIC
104.E3,0.3
*PLASTIC,HARD=COMBINED,DATA TYPE=STABILIZED
 0.46940E+02, 0.0000
 0.52044E+02, 0.50927E-04
 0.56826E+02, 0.10494E-03
 0.61273E+02, 0.16218E-03
 0.65813E+02, 0.21853E-03
 0.69556E+02, 0.27253E-03
 0.72703E+02, 0.34228E-03
 0.76541E+02, 0.39538E-03
 0.79630E+02, 0.49568E-03
 0.82886E+02, 0.55437E-03
 0.85974E+02, 0.61468E-03
 0.88910E+02, 0.69644E-03
 0.91647E+02, 0.76012E-03
 0.94263E+02, 0.83498E-03
 0.96266E+02, 0.92571E-03
 0.99020E+02, 0.99923E-03
 0.10099E+03, 0.10803E-02
 0.10310E+03, 0.11600E-02
 0.10534E+03, 0.12385E-02
 0.10691E+03, 0.13233E-02
 0.10846E+03, 0.14085E-02
 0.11054E+03, 0.14785E-02
 0.11252E+03, 0.15795E-02
 0.11328E+03, 0.16621E-02
 0.11521E+03, 0.17336E-02
 0.11612E+03, 0.18248E-02
 0.11769E+03, 0.19197E-02
 0.11906E+03, 0.20065E-02
 0.12018E+03, 0.20858E-02
 0.12137E+03, 0.21743E-02
 0.12239E+03, 0.22746E-02
 0.12338E+03, 0.23550E-02
 0.12467E+03, 0.24426E-02
 0.12500E+03, 0.25494E-02
 0.12649E+03, 0.26351E-02
 0.12715E+03, 0.27288E-02
 0.12805E+03, 0.28101E-02
 0.12879E+03, 0.29130E-02
 0.12980E+03, 0.30033E-02
 0.13040E+03, 0.30875E-02
 0.13049E+03, 0.31867E-02
```

```
0.13165E+03, 0.32855E-02
0.13191E+03, 0.33830E-02
0.13263E+03, 0.34660E-02
0.13324E+03, 0.35602E-02
0.13386E+03, 0.36642E-02
0.13398E+03, 0.37631E-02
0.13501E+03, 0.38632E-02
0.13513E+03, 0.39620E-02
0.13614E+03, 0.40423E-02
0.13623E+03, 0.41314E-02
0.13682E+03, 0.42358E-02
0.13691E+03, 0.43349E-02
0.13749E+03, 0.44193E-02
0.13738E+03, 0.45304E-02
0.13810E+03, 0.46234E-02
0.13859E+03, 0.47087E-02
0.13885E+03, 0.48062E-02
0.13930E+03, 0.49019E-02
0.13906E+03, 0.50042E-02
0.13991E+03, 0.50960E-02
0.13986E+03, 0.51965E-02
0.13979E+03, 0.53072E-02
0.14023E+03, 0.54030E-02
0.14061E+03, 0.54893E-02
0.14020E+03, 0.55933E-02
0.14118E+03, 0.56838E-02
0.14098E+03, 0.57958E-02
0.14218E+03, 0.58542E-02
0.14132E+03, 0.59825E-02
0.14153E+03, 0.60705E-02
0.14202E+03, 0.61658E-02
0.14215E+03, 0.62746E-02
0.14222E+03, 0.63738E-02
0.14177E+03, 0.64682E-02
0.14313E+03, 0.65651E-02
0.14205E+03, 0.66755E-02
0.14320E+03, 0.67744E-02
0.14245E+03, 0.68817E-02
0.14317E+03, 0.69447E-02
0.14294E+03, 0.70670E-02
0.14374E+03, 0.71492E-02
0.14306E+03, 0.72558E-02
0.14371E+03, 0.73596E-02
0.14361E+03, 0.74605E-02
0.14352E+03, 0.75514E-02
0.14353E+03, 0.76412E-02
0.14399E+03, 0.77468E-02
0.14427E+03, 0.78441E-02
0.14364E+03, 0.79502E-02
0.14501E+03, 0.80370E-02
0.14371E+03, 0.81495E-02
0.14478E+03, 0.82393E-02
0.14419E+03, 0.83449E-02
0.14451E+03, 0.84318E-02
0.14439E+03, 0.85430E-02
0.14511E+03, 0.86360E-02
0.14394E+03, 0.87473E-02
0.14519E+03, 0.88253E-02
0.14441E+03, 0.89427E-02
0.14546E+03, 0.90327E-02
0.14455E+03, 0.91414E-02
```

```
 0.14560E+03, 0.92313E-02
 0.14484E+03, 0.93386E-02
 0.14582E+03, 0.94193E-02
 0.14515E+03, 0.95357E-02
 0.14491E+03, 0.96179E-02
 0.14558E+03, 0.97215E-02
 0.14533E+03, 0.98239E-02
 0.14613E+03, 0.99062E-02
 0.14477E+03, 0.10029E-01
 0.14638E+03, 0.10104E-01
 0.14557E+03, 0.10212E-01
 0.14609E+03, 0.10307E-01
 0.14571E+03, 0.10420E-01
 0.14599E+03, 0.10518E-01
 0.14546E+03, 0.10623E-01
 0.14617E+03, 0.10706E-01
 0.14574E+03, 0.10790E-01
 0.14644E+03, 0.10903E-01
 0.14546E+03, 0.11013E-01
 0.14632E+03, 0.11104E-01
 0.14608E+03, 0.11207E-01
 0.14615E+03, 0.11306E-01
 0.14643E+03, 0.11403E-01
 0.14565E+03, 0.11501E-01
 0.14700E+03, 0.11588E-01
 0.14585E+03, 0.11699E-01
 0.14714E+03, 0.11787E-01
 0.14537E+03, 0.11914E-01
*CYCLIC HARDENING
13.,0.000
13.,0.0068
29.37,.034
50.13,.0612
63.91,.0884
73.01,.1156
80.76,.1428
86.25,.1700
90.498,.1972
93.95,.2244
96.155,.2516
*BOUNDARY
1,1,6
*STEP,INC=10
*STATIC,DIRECT
0.1,1.,
*BOUNDARY
2,1,1,0.01
*FILE FORMAT, ASCII
*EL FILE
EP
*END STEP
```

```
type('404.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PRINCIPAL NOMINAL STRAINS OUTPUT TO MATLAB (NEP, RECORD KEY 404)
*RESTART,WRITE
*NODE, NSET=NALL
1,
2,1.,
*ELEMENT,TYPE=PIPE31,ELSET=EALL
1,1,2,
*BEAM SECTION,SECTION=PIPE,ELSET=EALL,MATERIAL=COPPER
1.01,0.02,
*MATERIAL,NAME=COPPER
*ELASTIC
104.E3,0.3
*PLASTIC,HARD=COMBINED,DATA TYPE=STABILIZED
 0.46940E+02, 0.0000
 0.52044E+02, 0.50927E-04
 0.56826E+02, 0.10494E-03
 0.61273E+02, 0.16218E-03
 0.65813E+02, 0.21853E-03
 0.69556E+02, 0.27253E-03
 0.72703E+02, 0.34228E-03
 0.76541E+02, 0.39538E-03
 0.79630E+02, 0.49568E-03
 0.82886E+02, 0.55437E-03
 0.85974E+02, 0.61468E-03
 0.88910E+02, 0.69644E-03
 0.91647E+02, 0.76012E-03
 0.94263E+02, 0.83498E-03
 0.96266E+02, 0.92571E-03
 0.99020E+02, 0.99923E-03
 0.10099E+03, 0.10803E-02
 0.10310E+03, 0.11600E-02
 0.10534E+03, 0.12385E-02
 0.10691E+03, 0.13233E-02
 0.10846E+03, 0.14085E-02
 0.11054E+03, 0.14785E-02
 0.11252E+03, 0.15795E-02
 0.11328E+03, 0.16621E-02
 0.11521E+03, 0.17336E-02
 0.11612E+03, 0.18248E-02
 0.11769E+03, 0.19197E-02
 0.11906E+03, 0.20065E-02
 0.12018E+03, 0.20858E-02
 0.12137E+03, 0.21743E-02
 0.12239E+03, 0.22746E-02
 0.12338E+03, 0.23550E-02
 0.12467E+03, 0.24426E-02
 0.12500E+03, 0.25494E-02
 0.12649E+03, 0.26351E-02
 0.12715E+03, 0.27288E-02
 0.12805E+03, 0.28101E-02
 0.12879E+03, 0.29130E-02
 0.12980E+03, 0.30033E-02
 0.13040E+03, 0.30875E-02
 0.13049E+03, 0.31867E-02
```

```
0.13165E+03, 0.32855E-02
0.13191E+03, 0.33830E-02
0.13263E+03, 0.34660E-02
0.13324E+03, 0.35602E-02
0.13386E+03, 0.36642E-02
0.13398E+03, 0.37631E-02
0.13501E+03, 0.38632E-02
0.13513E+03, 0.39620E-02
0.13614E+03, 0.40423E-02
0.13623E+03, 0.41314E-02
0.13682E+03, 0.42358E-02
0.13691E+03, 0.43349E-02
0.13749E+03, 0.44193E-02
0.13738E+03, 0.45304E-02
0.13810E+03, 0.46234E-02
0.13859E+03, 0.47087E-02
0.13885E+03, 0.48062E-02
0.13930E+03, 0.49019E-02
0.13906E+03, 0.50042E-02
0.13991E+03, 0.50960E-02
0.13986E+03, 0.51965E-02
0.13979E+03, 0.53072E-02
0.14023E+03, 0.54030E-02
0.14061E+03, 0.54893E-02
0.14020E+03, 0.55933E-02
0.14118E+03, 0.56838E-02
0.14098E+03, 0.57958E-02
0.14218E+03, 0.58542E-02
0.14132E+03, 0.59825E-02
0.14153E+03, 0.60705E-02
0.14202E+03, 0.61658E-02
0.14215E+03, 0.62746E-02
0.14222E+03, 0.63738E-02
0.14177E+03, 0.64682E-02
0.14313E+03, 0.65651E-02
0.14205E+03, 0.66755E-02
0.14320E+03, 0.67744E-02
0.14245E+03, 0.68817E-02
0.14317E+03, 0.69447E-02
0.14294E+03, 0.70670E-02
0.14374E+03, 0.71492E-02
0.14306E+03, 0.72558E-02
0.14371E+03, 0.73596E-02
0.14361E+03, 0.74605E-02
0.14352E+03, 0.75514E-02
0.14353E+03, 0.76412E-02
0.14399E+03, 0.77468E-02
0.14427E+03, 0.78441E-02
0.14364E+03, 0.79502E-02
0.14501E+03, 0.80370E-02
0.14371E+03, 0.81495E-02
0.14478E+03, 0.82393E-02
0.14419E+03, 0.83449E-02
0.14451E+03, 0.84318E-02
0.14439E+03, 0.85430E-02
0.14511E+03, 0.86360E-02
0.14394E+03, 0.87473E-02
0.14519E+03, 0.88253E-02
0.14441E+03, 0.89427E-02
0.14546E+03, 0.90327E-02
0.14455E+03, 0.91414E-02
```

```
 0.14560E+03, 0.92313E-02
 0.14484E+03, 0.93386E-02
 0.14582E+03, 0.94193E-02
 0.14515E+03, 0.95357E-02
 0.14491E+03, 0.96179E-02
 0.14558E+03, 0.97215E-02
 0.14533E+03, 0.98239E-02
 0.14613E+03, 0.99062E-02
 0.14477E+03, 0.10029E-01
 0.14638E+03, 0.10104E-01
 0.14557E+03, 0.10212E-01
 0.14609E+03, 0.10307E-01
 0.14571E+03, 0.10420E-01
 0.14599E+03, 0.10518E-01
 0.14546E+03, 0.10623E-01
 0.14617E+03, 0.10706E-01
 0.14574E+03, 0.10790E-01
 0.14644E+03, 0.10903E-01
 0.14546E+03, 0.11013E-01
 0.14632E+03, 0.11104E-01
 0.14608E+03, 0.11207E-01
 0.14615E+03, 0.11306E-01
 0.14643E+03, 0.11403E-01
 0.14565E+03, 0.11501E-01
 0.14700E+03, 0.11588E-01
 0.14585E+03, 0.11699E-01
 0.14714E+03, 0.11787E-01
 0.14537E+03, 0.11914E-01
*CYCLIC HARDENING
13.,0.000
13.,0.0068
29.37,.034
50.13,.0612
63.91,.0884
73.01,.1156
80.76,.1428
86.25,.1700
90.498,.1972
93.95,.2244
96.155,.2516
*BOUNDARY
1,1,6
*STEP,INC=10
*STATIC,DIRECT
0.1,1.,
*BOUNDARY
2,1,1,0.01
*FILE FORMAT, ASCII
*EL FILE
NEP
*END STEP
```

```
type('405.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL LOGARITHMIC STRAINS OUTPUT TO MATLAB (LEP, RECORD KEY 405)
*RESTART,WRITE
*NODE, NSET=NALL
1,
2,1.,
*ELEMENT,TYPE=PIPE31,ELSET=EALL
1,1,2,
*BEAM SECTION,SECTION=PIPE,ELSET=EALL,MATERIAL=COPPER
1.01,0.02,
*MATERIAL,NAME=COPPER
*ELASTIC
104.E3,0.3
*PLASTIC,HARD=COMBINED,DATA TYPE=STABILIZED
 0.46940E+02, 0.0000
 0.52044E+02, 0.50927E-04
 0.56826E+02, 0.10494E-03
 0.61273E+02, 0.16218E-03
 0.65813E+02, 0.21853E-03
 0.69556E+02, 0.27253E-03
 0.72703E+02, 0.34228E-03
 0.76541E+02, 0.39538E-03
 0.79630E+02, 0.49568E-03
 0.82886E+02, 0.55437E-03
 0.85974E+02, 0.61468E-03
 0.88910E+02, 0.69644E-03
 0.91647E+02, 0.76012E-03
 0.94263E+02, 0.83498E-03
 0.96266E+02, 0.92571E-03
 0.99020E+02, 0.99923E-03
 0.10099E+03, 0.10803E-02
 0.10310E+03, 0.11600E-02
 0.10534E+03, 0.12385E-02
 0.10691E+03, 0.13233E-02
 0.10846E+03, 0.14085E-02
 0.11054E+03, 0.14785E-02
 0.11252E+03, 0.15795E-02
 0.11328E+03, 0.16621E-02
 0.11521E+03, 0.17336E-02
 0.11612E+03, 0.18248E-02
 0.11769E+03, 0.19197E-02
 0.11906E+03, 0.20065E-02
 0.12018E+03, 0.20858E-02
 0.12137E+03, 0.21743E-02
 0.12239E+03, 0.22746E-02
 0.12338E+03, 0.23550E-02
 0.12467E+03, 0.24426E-02
 0.12500E+03, 0.25494E-02
 0.12649E+03, 0.26351E-02
 0.12715E+03, 0.27288E-02
 0.12805E+03, 0.28101E-02
 0.12879E+03, 0.29130E-02
 0.12980E+03, 0.30033E-02
 0.13040E+03, 0.30875E-02
 0.13049E+03, 0.31867E-02
```

```
0.13165E+03, 0.32855E-02
0.13191E+03, 0.33830E-02
0.13263E+03, 0.34660E-02
0.13324E+03, 0.35602E-02
0.13386E+03, 0.36642E-02
0.13398E+03, 0.37631E-02
0.13501E+03, 0.38632E-02
0.13513E+03, 0.39620E-02
0.13614E+03, 0.40423E-02
0.13623E+03, 0.41314E-02
0.13682E+03, 0.42358E-02
0.13691E+03, 0.43349E-02
0.13749E+03, 0.44193E-02
0.13738E+03, 0.45304E-02
0.13810E+03, 0.46234E-02
0.13859E+03, 0.47087E-02
0.13885E+03, 0.48062E-02
0.13930E+03, 0.49019E-02
0.13906E+03, 0.50042E-02
0.13991E+03, 0.50960E-02
0.13986E+03, 0.51965E-02
0.13979E+03, 0.53072E-02
0.14023E+03, 0.54030E-02
0.14061E+03, 0.54893E-02
0.14020E+03, 0.55933E-02
0.14118E+03, 0.56838E-02
0.14098E+03, 0.57958E-02
0.14218E+03, 0.58542E-02
0.14132E+03, 0.59825E-02
0.14153E+03, 0.60705E-02
0.14202E+03, 0.61658E-02
0.14215E+03, 0.62746E-02
0.14222E+03, 0.63738E-02
0.14177E+03, 0.64682E-02
0.14313E+03, 0.65651E-02
0.14205E+03, 0.66755E-02
0.14320E+03, 0.67744E-02
0.14245E+03, 0.68817E-02
0.14317E+03, 0.69447E-02
0.14294E+03, 0.70670E-02
0.14374E+03, 0.71492E-02
0.14306E+03, 0.72558E-02
0.14371E+03, 0.73596E-02
0.14361E+03, 0.74605E-02
0.14352E+03, 0.75514E-02
0.14353E+03, 0.76412E-02
0.14399E+03, 0.77468E-02
0.14427E+03, 0.78441E-02
0.14364E+03, 0.79502E-02
0.14501E+03, 0.80370E-02
0.14371E+03, 0.81495E-02
0.14478E+03, 0.82393E-02
0.14419E+03, 0.83449E-02
0.14451E+03, 0.84318E-02
0.14439E+03, 0.85430E-02
0.14511E+03, 0.86360E-02
0.14394E+03, 0.87473E-02
0.14519E+03, 0.88253E-02
0.14441E+03, 0.89427E-02
0.14546E+03, 0.90327E-02
0.14455E+03, 0.91414E-02
```

```
 0.14560E+03, 0.92313E-02
 0.14484E+03, 0.93386E-02
 0.14582E+03, 0.94193E-02
 0.14515E+03, 0.95357E-02
 0.14491E+03, 0.96179E-02
 0.14558E+03, 0.97215E-02
 0.14533E+03, 0.98239E-02
 0.14613E+03, 0.99062E-02
 0.14477E+03, 0.10029E-01
 0.14638E+03, 0.10104E-01
 0.14557E+03, 0.10212E-01
 0.14609E+03, 0.10307E-01
 0.14571E+03, 0.10420E-01
 0.14599E+03, 0.10518E-01
 0.14546E+03, 0.10623E-01
 0.14617E+03, 0.10706E-01
 0.14574E+03, 0.10790E-01
 0.14644E+03, 0.10903E-01
 0.14546E+03, 0.11013E-01
 0.14632E+03, 0.11104E-01
 0.14608E+03, 0.11207E-01
 0.14615E+03, 0.11306E-01
 0.14643E+03, 0.11403E-01
 0.14565E+03, 0.11501E-01
 0.14700E+03, 0.11588E-01
 0.14585E+03, 0.11699E-01
 0.14714E+03, 0.11787E-01
 0.14537E+03, 0.11914E-01
*CYCLIC HARDENING
13.,0.000
13.,0.0068
29.37,.034
50.13,.0612
63.91,.0884
73.01,.1156
80.76,.1428
86.25,.1700
90.498,.1972
93.95,.2244
96.155,.2516
*BOUNDARY
1,1,6
*STEP,INC=10
*STATIC,DIRECT
0.1,1.,
*BOUNDARY
2,1,1,0.01
*FILE FORMAT, ASCII
*EL FILE
LEP
*END STEP
```

```
type('406.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL MECHANICAL STRAIN RATES OUTPUT TO MATLAB (ERP, RECORD KEY
406)
*RESTART,WRITE,FREQ=5
*NODE,NSET=ALLN
1,0.,0.,0.
2,1.,0.,0.
3,1.,1.,0.
4,0.,1.,0.
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*ELEMENT,TYPE=C3D8R,ELSET=ALLE
1, 1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=DEL3
*MATERIAL,NAME=DEL3
*ELASTIC
200.E3,.3
*DENSITY
82.9,
*BOUNDARY
1,PINNED
2,2
5,2
6,2
4,1
5,1
8,1
2,3
3,3
4,3
*STEP,NLGEOM,INC=10
*DYNAMIC,DIRECT
.1,1.
*BOUNDARY
5,3,,.04
6,3,,.04
7,3,,.04
8,3,,.04
*FILE FORMAT, ASCII
*EL FILE
ERP
*END STEP
```

```
type('407.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL VALUES OF DEFORMATION GRADIENT OUTPUT TO MATLAB (DGP,
RECORD KEY 407)
*NODE,NSET=ALL
1,
2,1.
3,1.,1.,
4,0.,1.,
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*NSET,NSET=FACE1
1,2,3,4
*NSET,NSET=FACE2
5,6,7,8
*NSET,NSET=FACE3
1,2,5,6
*NSET,NSET=FACE4
2,
*NSET,NSET=FACE42
3,6,7
*NSET,NSET=FACE5
3,4,7,8
*NSET,NSET=FACE6
4,1,8,5
*EQUATION
2,
FACE42,1,1, 2,1,-1
*ELEMENT,TYPE=C3D8R,ELSET=ONE
 1, 1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ONE,MATERIAL=FOAM
*MATERIAL,NAME=FOAM
*HYPERFOAM,N=2,TEST DATA INPUT,POISSON=0.
*UNIAXIAL TEST DATA
 -.0217,   -.05
 -.0317,   -.10
 -.0367,   -.15
 -.0402,   -.20
 -.0433,   -.25
 -.0467,   -.30
 -.0504,   -.35
 -.0542,   -.40
 -.0604,   -.45
 -.0668,   -.50
 -.0759,   -.55
 -.0909,   -.60
 -.1083,   -.65
 -.1410,   -.70
 -.1933,   -.75
 -.2896,   -.80
*SIMPLE SHEAR TEST DATA
  .0140, .08, .0046
  .0334, .16, .0166
  .0533, .24, .0366
```

```
  .0853, .32, .0573
  .1280, .40, .0817
  .1653, .48, .1098
  .2080, .56, .1394
  .2560, .64, .1666
  .2987, .72, .1904
*RESTART,WRITE,FREQUENCY=5
*STEP,NLGEOM,INC=20
*STATIC
.05,.80,.05,.05
*BOUNDARY
FACE1,3
FACE3,2
FACE6,1
FACE4,1,1,-.80
*FILE FORMAT, ASCII
*EL FILE,FREQUENCY=5
DGP
*END STEP
```

```
type('408.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL ELASTIC STRAINS OUTPUT TO MATLAB (EEP, RECORD KEY 408)
*RESTART,WRITE,FREQ=5
*NODE, NSET=BOT
1,0.,0.,0.
2,1.,0.,0.
3,1.,1.,0.
4,0.,1.,0.
*NODE, NSET=TOP
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*NSET, NSET=BACK
1,4,5,8
*NSET, NSET=LHS
1,2,5,6
*ELEMENT, TYPE=C3D8R, ELSET=EL1
1,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL1, MATERIAL=SCP3
*MATERIAL,NAME=SCP3
*ELASTIC
12857.1429, 0.28571429
*CAP PLASTICITY
173.20508, 30.0, 0.61858957, 0.027, 0.69258232, 1.0
*CAP HARDENING
 213.0, 0.00
 222.0, 0.01
 242.0, 0.02
 282.0, 0.03
 362.0, 0.04
 522.0, 0.05
 842.0, 0.06
1482.0, 0.07
2762.0, 0.08
*ELEMENT, TYPE=C3D8R, ELSET=EL2
101,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL2, MATERIAL=ELAS
*MATERIAL,NAME=ELAS
*ELASTIC
128.571429, 0.28571429
*STEP, INC=10, UNSYMM=YES
*STATIC, DIRECT
1., 10.
*BOUNDARY
BACK,1
LHS, 2
BOT, 3
*DLOAD
EL1, P4, -300.0
*FILE FORMAT, ASCII
*EL FILE,FREQ=10
EEP
*END STEP
```

```
type('409.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL INELASTIC STRAINS OUTPUT TO MATLAB (IEP, RECORD KEY 409)
*RESTART,WRITE,FREQ=5
*NODE, NSET=BOT
1,0.,0.,0.
2,1.,0.,0.
3,1.,1.,0.
4,0.,1.,0.
*NODE, NSET=TOP
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*NSET, NSET=BACK
1,4,5,8
*NSET, NSET=LHS
1,2,5,6
*ELEMENT, TYPE=C3D8R, ELSET=EL1
1,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL1, MATERIAL=SCP3
*MATERIAL,NAME=SCP3
*ELASTIC
12857.1429, 0.28571429
*CAP PLASTICITY
173.20508, 30.0, 0.61858957, 0.027, 0.69258232, 1.0
*CAP HARDENING
 213.0, 0.00
 222.0, 0.01
 242.0, 0.02
 282.0, 0.03
 362.0, 0.04
 522.0, 0.05
 842.0, 0.06
1482.0, 0.07
2762.0, 0.08
*ELEMENT, TYPE=C3D8R, ELSET=EL2
101,1,2,3,4,5,6,7,8
*SOLID SECTION, ELSET=EL2, MATERIAL=ELAS
*MATERIAL,NAME=ELAS
*ELASTIC
128.571429, 0.28571429
*STEP, INC=10, UNSYMM=YES
*STATIC, DIRECT
1., 10.
*BOUNDARY
BACK,1
LHS, 2
BOT, 3
*DLOAD
EL1, P4, -300.0
*FILE FORMAT, ASCII
*EL FILE,FREQ=10
IEP
*END STEP
```

```
type('410.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS PRINCIPAL THERMAL STRAINS OUTPUT TO MATLAB (THEP, RECORD KEY 410)
*PREPRINT,ECHO=YES,HISTORY=NO,MODEL=NO
*NODE
  1, 0.  ,  0.
  5, 0.5 ,  0.
 21, 0.  ,  0.5
 25, 0.5 ,  0.5
 61, 0.  ,  1.
 69, 1.  ,  0.
 81, 0.  ,  2.
 89, 2.  ,  0.
151, 0.  ,  9.
159, 9.  ,  0.
161, 0.  ,  10.
163, 4.14,  10.
165, 10. ,  10.
167, 10. ,  4.14
169, 10. ,  0.
*NGEN,NSET=SMBOT
1,5,2
*NGEN,NSET=SMTOP
21,25,2
*NFILL,NSET=SMBOX
SMBOT,SMTOP,2,10
*NGEN,NSET=CIRC61,LINE=C
61,69,2
*NGEN,NSET=CIRC81,LINE=C
81,89,2
*NGEN,NSET=CIRC151,LINE=C
151,159,2
*NFILL,NSET=CIRCS
CIRC81,CIRC151,7,10
*NGEN
 145,165,10
*ELEMENT,TYPE=CPS3, ELSET=EALL
  1, 1,13,11
  2, 1, 3,13
 10,21,63,61
 11,21,23,63
 12,23,65,63
 13,23,25,65
 14,25,67,65
 15,25,15,67
 16,15,69,67
 17,15, 5,69
 30,81,103,101
 31,81,83,103
*ELGEN, ELSET=EALL
  1,2,2,2,2,10,4
  2,2,2,2,2,10,4
 30,4,2,2,4,20,10
 31,4,2,2,4,20,10
*NODE
 200, 0., 1.
```

```
 208, 1., 0.
*NGEN,NSET=OUTER,LINE=C
 200,208,2
*NSET,NSET=INNER
 CIRC61,
*ELEMENT,TYPE=CPS3, ELSET=EALL
 20,200,83,81
 21,200,202,83
*ELGEN, ELSET=EALL
 20,4,2,2
 21,4,2,2
*NSET,NSET=HOT,GENERATE
 1,25
 41,46
 51,59
 61,69
*NSET,NSET=SYMX1,GENERATE
 1,5
 59,169,10
*NSET,NSET=SYMX
 208,SYMX1
*NSET,NSET=SYMY1,GENERATE
 1,21,5
 51,161,10
*NSET,NSET=SYMY
 200,SYMY1
*ELSET,ELSET=SPOT,GENERATE
  1,8
 10,17
*ELSET,ELSET=PRINT
 16,17
 26,27
*EQUATION
2,
INNER,1,1.,OUTER,1,-1.
*EQUATION
2,
INNER,2,1.,OUTER,2,-1.
*MATERIAL,NAME=A1
*ELASTIC
 100.E9,0.3
*EXPANSION
 1.E-05,
*SOLID SECTION,MATERIAL=A1, ELSET=EALL
*BOUNDARY
 SYMX,2
 SYMY,1
*STEP
*STATIC
*TEMPERATURE
 HOT,100.
*FILE FORMAT, ASCII
*EL FILE,ELSET=PRINT,POSITION=AVERAGED AT NODES
 THEP
*END STEP
```

```
type('411.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT PRINCIPAL PLASTIC STRAINS OUTPUT TO MATLAB (PEP, RECORD KEY 411)
*RESTART,WRITE,FREQ=5
*NODE,NSET=ALLN
1,0.,0.,0.
2,1.,0.,0.
3,1.,1.,0.
4,0.,1.,0.
5,0.,0.,1.
6,1.,0.,1.
7,1.,1.,1.
8,0.,1.,1.
*ELEMENT,TYPE=C3D8R,ELSET=ALLE
1,1,2,3,4,5,6,7,8
*SOLID SECTION,ELSET=ALLE,MATERIAL=SCN3
*MATERIAL,NAME=SCN3
*ELASTIC
4.65E6,.18
*CONCRETE
1300.,0.
2200.,.000027
3000.,.0001
3600.,.000225
4450.,.00055
4650.,.001
4200.,.002
2000.,.0035
*FAILURE RATIOS
1.18,.1,1.25,.2
*TENSION STIFFENING
1.,0.
0.,5.E-4
*BOUNDARY
1,PINNED
2,2
5,2
6,2
4,1
5,1
8,1
2,3
3,3
4,3
*STEP,INC=20
*STATIC,DIRECT
1.,20.
*BOUNDARY
7,3,,.0008
5,3,,.0008
6,3,,.0008
8,3,,.0008
*FILE FORMAT, ASCII
*EL FILE
PEP
*END STEP
```

```
type('1900.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ELEMENT DEFINITIONS OUTPUT TO MATLAB (RECORD KEY 1900)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,      0,      360
 6,      0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*END STEP
```

```
type('1901.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS NODAL DEFINITIONS OUTPUT TO MATLAB (RECORD KEY 1901)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,    360
 2,      720,    0
 3,      360,    360
 4,      360,    0
 5,      0,      360
 6,      0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*NODE FILE
 COORD
*END STEP
```

Published with MATLAB® R2015a

```
type('1902.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ACTIVE DEGREES OF FREEDOM OUTPUT TO MATLAB (RECORD KEY 1902)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*END STEP
```

Published with MATLAB® R2015a

```
type('1911.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS OUTPUT REQUEST DEFINITION OUTPUT TO MATLAB (RECORD KEY 1911)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*CLOAD
 _PICKEDSET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*NODE FILE
 U
*EL FILE
 ELEN
*END STEP
```

Published with MATLAB® R2015a

```
type('1921.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS ANALYSIS INFORMATION OUTPUT TO MATLAB (RECORD KEY 1921)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,      720,     360
 2,      720,     0
 3,      360,     360
 4,      360,     0
 5,      0,       360
 6,      0,       0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=SET1
 1,
*ELSET, ELSET=SET2
 2,
*ELSET, ELSET=SET3
 3,
*ELSET, ELSET=SET4
 4,
*ELSET, ELSET=SET5
 5,
*ELSET, ELSET=SET6
 6,
*ELSET, ELSET=SET7
 7,
*ELSET, ELSET=SET8
 8,
*ELSET, ELSET=SET9
 9,
*ELSET, ELSET=SET10
 10,
*FRAME SECTION, ELSET=SET1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=SET4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=SET21, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=SET22, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 SET22, 1, 1
 SET22, 2, 2
*CLOAD
 SET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*END STEP
```

```
type('1940.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS LABEL CROSS-REFERENCE OUTPUT TO MATLAB (RECORD KEY 1940)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,     720,    360
 2,     720,    0
 3,     360,    360
 4,     360,    0
 5,     0,      360
 6,     0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=SET1
 1,
*ELSET, ELSET=SET2
 2,
*ELSET, ELSET=SET3
 3,
*ELSET, ELSET=SET4
 4,
*ELSET, ELSET=SET5
 5,
*ELSET, ELSET=SET6
 6,
*ELSET, ELSET=SET7
 7,
*ELSET, ELSET=SET8
 8,
*ELSET, ELSET=SET9
 9,
*ELSET, ELSET=SET10
 10,
*FRAME SECTION, ELSET=SET1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=SET4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=SET10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=SET21, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=SET22, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-1
*DYNAMIC
 1., 1., 1E-05, 1.
*BOUNDARY
 SET22, 1, 1
 SET22, 2, 2
*CLOAD
 SET21, 2, -100.
*FILE FORMAT, ASCII
*EL FILE
 COORD
*END STEP
```

```
type('1980.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS FREQUENCY ANALYSIS OUTPUT TO MATLAB (RECORD KEY 1980)
*PART, NAME=PART-5
*END PART
*ASSEMBLY, NAME=ASSEMBLY
*INSTANCE, NAME=PART-5-1, PART=PART-5
*NODE
 1,     720,    360
 2,     720,    0
 3,     360,    360
 4,     360,    0
 5,     0,      360
 6,     0,      0
*ELEMENT, TYPE=FRAME2D
 1, 5, 3
 2, 3, 1
 3, 6, 4
 4, 4, 2
 5, 3, 4
 6, 1, 2
 7, 5, 4
 8, 6, 3
 9, 3, 2
 10, 4, 1
*ELSET, ELSET=_PICKEDSET2_#1, INTERNAL
 1,
*ELSET, ELSET=_PICKEDSET2_#2, INTERNAL
 2,
*ELSET, ELSET=_PICKEDSET2_#3, INTERNAL
 3,
*ELSET, ELSET=_PICKEDSET2_#4, INTERNAL
 4,
*ELSET, ELSET=_PICKEDSET2_#5, INTERNAL
 5,
*ELSET, ELSET=_PICKEDSET2_#6, INTERNAL
 6,
*ELSET, ELSET=_PICKEDSET2_#7, INTERNAL
 7,
*ELSET, ELSET=_PICKEDSET2_#8, INTERNAL
 8,
*ELSET, ELSET=_PICKEDSET2_#9, INTERNAL
 9,
*ELSET, ELSET=_PICKEDSET2_#10, INTERNAL
 10,
*FRAME SECTION, ELSET=_PICKEDSET2_#1, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#2, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#3, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
```

```
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#4, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#5, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#6, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#7, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#8, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#9, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*FRAME SECTION, ELSET=_PICKEDSET2_#10, PINNED
 1, 1.6E-3, 0, 9E-4
 0.,0.,-1.
 1E4, 1E3
*ELEMENT,TYPE=MASS,ELSET=MASSES
 11,1
 12,2
 13,3
 14,4
 15,5
 16,6
*MASS, ELSET=MASSES
 1
*END INSTANCE
*NSET, NSET=_PICKEDSET21, INTERNAL, INSTANCE=PART-5-1
 2, 4
*NSET, NSET=_PICKEDSET22, INTERNAL, INSTANCE=PART-5-1
 5, 6
*END ASSEMBLY
*STEP, NAME=STEP-2, NLGEOM=YES, PERTURBATION
*FREQUENCY, EIGENSOLVER=LANCZOS, ACOUSTIC COUPLING=ON, NORMALIZATION=DISPLACEMENT
10, , , , ,
*BOUNDARY
 _PICKEDSET22, 1, 1
 _PICKEDSET22, 2, 2
*FILE FORMAT, ASCII
*EL FILE
 SF
*END STEP
```

```
type('1991.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS J-INTEGRAL OUTPUT TO MATLAB (JK, RECORD KEY 1991)
*SYSTEM
10.,0.,0.
*NODE,SYSTEM=C
1833,0.,0.,0.
9833,0.,0.,0.
57833,0.,0.,0.
57865,0.,0.,0.
65833,0.,0.,0.
1001,15.,-45.,0.
65001,15.,-45.,0.
1033,8.,-45.,0.
65033,8.,-45.,0.
33033,25.,-45.,0.
41033,25.,-18.,0.
57033,10.,-18.,0.
9033,10.,-72.,0.
25033,25.,-72.,0.
41065,25.,0.,0.
57065,10.,0.,0.
*SYSTEM
0.,0.,0.
*NODE,SYSTEM=C
9865,0.,0.,0.
9065,15.,-90.,0.
25065,30.,-90.,0.
25865,170.,-90.,0.
25965,340.,-90.,0.
25833,170.,-60.,0.
25933,340.,-60.,0.
33833,170.,-45.,0.
33933,340.,-45.,0.
41833,170.,-30.,0.
41933,340.,-30.,0.
41865,170.,0.,0.
41965,340.,0.,0.
*NGEN,NSET=TIP
1001,65001,1000
*NGEN,NSET=OUTER1
1033,9033,1000
*NGEN,NSET=OUTER2
9033,25033,1000
*NGEN,NSET=OUTER3
25033,33033,1000
*NGEN,NSET=OUTER4
33033,41033,1000
*NGEN,NSET=OUTER5
41033,57033,1000
*NGEN,NSET=OUTER6
57033,65033,1000
*NSET,NSET=OUTER
OUTER1,OUTER2,OUTER3,OUTER4,OUTER5,OUTER6
*NFILL,NSET=JREGION,SINGULAR=1
TIP,OUTER,16,2
```

```
*NGEN,NSET=BOT9
9033,9065,1
*NGEN,NSET=TOP9
9833,9865,1
*NFILL,NSET=ALL9
BOT9,TOP9,16,50
*NGEN,NSET=BOT25
25033,25065,1
*NGEN,NSET=TOP25,LINE=C
25833,25865,1,9865
*NFILL,NSET=ALL25,BIAS=0.8
BOT25,TOP25,16,50
*NGEN,NSET=BOT41
41033,41065,1
*NGEN,NSET=TOP41,LINE=C
41833,41865,1,9865
*NFILL,NSET=ALL41,BIAS=0.8
BOT41,TOP41,16,50
*NFILL,NSET=ALL925
BOT9,BOT25,16,1000
*NSET,NSET=BOT2533,GENERATE
25033,33033,1000
*NGEN,NSET=TOP2533,LINE=C
25833,33833,1000,9865
*NSET,NSET=BOT3341,GENERATE
33033,41033,1000
*NGEN,NSET=TOP3341,LINE=C
33833,41833,1000,9865
*NSET,NSET=BOT2541
BOT2533,BOT3341
*NSET,NSET=TOP2541
TOP2533,TOP3341
*NFILL,NSET=ALL2541,BIAS=0.8
BOT2541,TOP2541,16,50
*NGEN,NSET=BOT57
57033,57065,1
*NFILL,NSET=ALL4157
BOT41,BOT57,16,1000
*NGEN,NSET=TOP57
57833,57865,1
*NFILL,NSET=ALL57,BIAS=1.0
BOT57,TOP57,16,50
*NGEN,NSET=TOP5765
57833,65833,1000
*NFILL,NSET=ALL5765,BIAS=1.0
OUTER6,TOP5765,16,50
*NGEN,NSET=TOP19
1833,9833,1000
*NFILL,NSET=ALL19
OUTER1,TOP19,16,50
*NGEN,NSET=INF25,LINE=C
25933,25965,8,9865
*NGEN,NSET=INF2533,LINE=C
25933,33933,4000,9865
*NGEN,NSET=INF3341,LINE=C
33933,41933,4000,9865
*NGEN,NSET=INF41,LINE=C
41933,41965,8,9865
*NSET,NSET=INF
INF25,INF2533,INF3341,INF41
*NSET,NSET=N0
```

```
JREGION,ALL9,ALL25,ALL41,ALL57,ALL19,ALL925,ALL2541,
ALL4157,ALL5765,INF
*NCOPY,CHANGE NUMBER=100000,OLD SET=N0,MULTIPLE=19,SHIFT,NEW SET=N119
0.,0.,0.
0.,0.,0.,0.,1.,0.,-4.5
*NCOPY,CHANGE NUMBER=2000000,OLD SET=N0,MULTIPLE=1,SHIFT,NEW SET=N20
0.,0.,0.
0.,0.,0.,0.,1.,0.,-90.
*NSET,NSET=R9,GENERATE
9065,9865,100
*NSET,NSET=R925,GENERATE
9065,25065,2000
*NSET,NSET=R25,GENERATE
25065,25865,100
*NSET,NSET=Y0
R9,R925,R25
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y0,MULTIPLE=1,SHIFT,NEW SET=Y1
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y1,MULTIPLE=1,SHIFT,NEW SET=Y2
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y2,MULTIPLE=1,SHIFT,NEW SET=Y3
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y3,MULTIPLE=1,SHIFT,NEW SET=Y4
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y4,MULTIPLE=1,SHIFT,NEW SET=Y5
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y5,MULTIPLE=1,SHIFT,NEW SET=Y6
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y6,MULTIPLE=1,SHIFT,NEW SET=Y7
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y7,MULTIPLE=1,SHIFT,NEW SET=Y8
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y8,MULTIPLE=1,SHIFT,NEW SET=Y9
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y9,MULTIPLE=1,SHIFT,NEW SET=Y10
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y10,MULTIPLE=1,SHIFT,NEW SET=Y11
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y11,MULTIPLE=1,SHIFT,NEW SET=Y12
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y12,MULTIPLE=1,SHIFT,NEW SET=Y13
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y13,MULTIPLE=1,SHIFT,NEW SET=Y14
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y14,MULTIPLE=1,SHIFT,NEW SET=Y15
0.,0.,0.
```

```
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y15,MULTIPLE=1,SHIFT,NEW SET=Y16
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y16,MULTIPLE=1,SHIFT,NEW SET=Y17
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y17,MULTIPLE=1,SHIFT,NEW SET=Y18
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y18,MULTIPLE=1,SHIFT,NEW SET=Y19
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y19,MULTIPLE=1,SHIFT,NEW SET=Y20
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NSET,NSET=R91,GENERATE
9065,9865,200
*NSET,NSET=R9251,GENERATE
13065,25065,4000
*NSET,NSET=R251,GENERATE
25265,25865,200
*NSET,NSET=Y01
R91,R9251,R251
*NCOPY,CHANGE NUMBER=100000,OLD SET=Y01,MULTIPLE=1,SHIFT,NEW SET=Y1_1
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y1_1,MULTIPLE=1,SHIFT,NEW SET=Y31
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y31,MULTIPLE=1,SHIFT,NEW SET=Y51
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y51,MULTIPLE=1,SHIFT,NEW SET=Y71
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y71,MULTIPLE=1,SHIFT,NEW SET=Y91
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y91,MULTIPLE=1,SHIFT,NEW SET=Y111
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y111,MULTIPLE=1,SHIFT,NEW SET=Y131
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y131,MULTIPLE=1,SHIFT,NEW SET=Y151
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y151,MULTIPLE=1,SHIFT,NEW SET=Y171
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NCOPY,CHANGE NUMBER=200000,OLD SET=Y171,MULTIPLE=1,SHIFT,NEW SET=Y191
0.,0.,0.
0.,0.,0.,0.,1.,0.,0.
*NSET,NSET=T0,GENERATE
1001,65001,1000
*NSET,NSET=T1,GENERATE
101001,165001,1000
*NSET,NSET=T2,GENERATE
201001,265001,1000
*NSET,NSET=T3,GENERATE
```

```
301001,365001,1000
*NSET,NSET=T4,GENERATE
401001,465001,1000
*NSET,NSET=T5,GENERATE
501001,565001,1000
*NSET,NSET=T6,GENERATE
601001,665001,1000
*NSET,NSET=T7,GENERATE
701001,765001,1000
*NSET,NSET=T8,GENERATE
801001,865001,1000
*NSET,NSET=T9,GENERATE
901001,965001,1000
*NSET,NSET=T10,GENERATE
1001001,1065001,1000
*NSET,NSET=T11,GENERATE
1101001,1165001,1000
*NSET,NSET=T12,GENERATE
1201001,1265001,1000
*NSET,NSET=T13,GENERATE
1301001,1365001,1000
*NSET,NSET=T14,GENERATE
1401001,1465001,1000
*NSET,NSET=T15,GENERATE
1501001,1565001,1000
*NSET,NSET=T16,GENERATE
1601001,1665001,1000
*NSET,NSET=T17,GENERATE
1701001,1765001,1000
*NSET,NSET=T18,GENERATE
1801001,1865001,1000
*NSET,NSET=T19,GENERATE
1901001,1965001,1000
*NSET,NSET=T20,GENERATE
2001001,2065001,1000
*NSET,NSET=TIP01,GENERATE
1001,64001,1000
*NSET,NSET=TIP11,GENERATE
101001,164001,1000
*NSET,NSET=TIP21,GENERATE
201001,264001,1000
*NSET,NSET=TIP31,GENERATE
301001,364001,1000
*NSET,NSET=TIP41,GENERATE
401001,464001,1000
*NSET,NSET=TIP51,GENERATE
501001,564001,1000
*NSET,NSET=TIP61,GENERATE
601001,664001,1000
*NSET,NSET=TIP71,GENERATE
701001,764001,1000
*NSET,NSET=TIP81,GENERATE
801001,864001,1000
*NSET,NSET=TIP91,GENERATE
901001,964001,1000
*NSET,NSET=TIP101,GENERATE
1001001,1064001,1000
*NSET,NSET=TIP111,GENERATE
1101001,1164001,1000
*NSET,NSET=TIP121,GENERATE
```

```
1201001,1264001,1000
*NSET,NSET=TIP131,GENERATE
1301001,1364001,1000
*NSET,NSET=TIP141,GENERATE
1401001,1464001,1000
*NSET,NSET=TIP151,GENERATE
1501001,1564001,1000
*NSET,NSET=TIP161,GENERATE
1601001,1664001,1000
*NSET,NSET=TIP171,GENERATE
1701001,1764001,1000
*NSET,NSET=TIP181,GENERATE
1801001,1864001,1000
*NSET,NSET=TIP191,GENERATE
1901001,1964001,1000
*NSET,NSET=TIP201,GENERATE
2001001,2064001,1000
*NSET,NSET=TIP02,GENERATE
2001,65001,1000
*NSET,NSET=TIP12,GENERATE
102001,165001,1000
*NSET,NSET=TIP22,GENERATE
202001,265001,1000
*NSET,NSET=TIP32,GENERATE
302001,365001,1000
*NSET,NSET=TIP42,GENERATE
402001,465001,1000
*NSET,NSET=TIP52,GENERATE
502001,565001,1000
*NSET,NSET=TIP62,GENERATE
602001,665001,1000
*NSET,NSET=TIP72,GENERATE
702001,765001,1000
*NSET,NSET=TIP82,GENERATE
802001,865001,1000
*NSET,NSET=TIP92,GENERATE
902001,965001,1000
*NSET,NSET=TIP102,GENERATE
1002001,1065001,1000
*NSET,NSET=TIP112,GENERATE
1102001,1165001,1000
*NSET,NSET=TIP122,GENERATE
1202001,1265001,1000
*NSET,NSET=TIP132,GENERATE
1302001,1365001,1000
*NSET,NSET=TIP142,GENERATE
1402001,1465001,1000
*NSET,NSET=TIP152,GENERATE
1502001,1565001,1000
*NSET,NSET=TIP162,GENERATE
1602001,1665001,1000
*NSET,NSET=TIP172,GENERATE
1702001,1765001,1000
*NSET,NSET=TIP182,GENERATE
1802001,1865001,1000
*NSET,NSET=TIP192,GENERATE
1902001,1965001,1000
*NSET,NSET=TIP202,GENERATE
2002001,2065001,1000
*ELEMENT,TYPE=C3D20R
1001,1001,1005,5005,5001,201001,201005,205005,205001,1003,3005,5003,3001,
```

```
201003,203005,205003,203001,101001,101005,105005,105001
*ELGEN,ELSET=RINGS
1001,16,4000,4000,8,4,4
*ELEMENT,TYPE=C3D20R
9033,9033,9233,9241,9041,209033,209233,209241,209041,9133,9237,9141,9037,
209133,209237,209141,209037,109033,109233,109241,109041
*ELGEN,ELSET=SECT9
9033,4,8,8,4,200,200
*ELSET,ELSET=TOP90,GENERATE
9633,9657,8
*ELSET,ELSET=TOP92,GENERATE
209633,209657,8
*ELSET,ELSET=TOP94,GENERATE
409633,409657,8
*ELSET,ELSET=TOP96,GENERATE
609633,609657,8
*ELSET,ELSET=TOP98,GENERATE
809633,809657,8
*ELSET,ELSET=TOP910,GENERATE
1009633,1009657,8
*ELSET,ELSET=TOP912,GENERATE
1209633,1209657,8
*ELSET,ELSET=TOP914,GENERATE
1409633,1409657,8
*ELSET,ELSET=TOP916,GENERATE
1609633,1609657,8
*ELSET,ELSET=TOP918,GENERATE
1809633,1809657,8
*ELSET,ELSET=TOP9
TOP90,TOP92,TOP94,TOP96,TOP98,TOP910,TOP912,TOP914,TOP916,TOP918
*ELEMENT,TYPE=C3D20R
25233,25233,25033,25041,25241,225233,225033,225041,225241,25133,25037,
25141,25237,225133,225037,225141,225237,125233,125033,125041,125241
*ELGEN,ELSET=SECT25
25233,4,8,8,4,200,200
*ELEMENT,TYPE=C3D20R
1033,1033,1233,5233,5033,201033,201233,205233,205033,1133,3233,5133,3033,
201133,203233,205133,203033,101033,101233,105233,105033
*ELGEN,ELSET=SECT19
1033,2,4000,4000,4,200,200
*ELEMENT,TYPE=C3D20R
13033,13033,9033,9041,13041,213033,209033,209041,213041,11033,9037,11041,
13037,211033,209037,211041,213037,113033,109033,109041,113041
*ELGEN,ELSET=SECT925
13033,4,8,8,4,4000,4000
*ELEMENT,TYPE=C3D20R
29233,29233,29033,25033,25233,229233,229033,225033,225233,29133,27033,
25133,27233,229133,227033,225133,227233,129233,129033,125033,125233
*ELGEN,ELSET=SECT2541
29233,4,4000,4000,4,200,200
*ELEMENT, TYPE=C3D20R
41241,41241,41041,41033,41233,241241,241041,241033,241233,41141,41037,
41133,41237,241141,241037,241133,241237,141241,141041,141033,141233
*ELGEN,ELSET=SECT41
41241,4,8,8,4,200,200
*ELEMENT,TYPE=C3D20R
41041,41041,45041,45033,41033,241041,245041,245033,241033,43041,45037,
43033,41037,243041,245037,243033,241037,141041,145041,145033,141033
*ELGEN,ELSET=SECT4157
41041,4,8,8,4,4000,4000
```

```
*ELEMENT, TYPE=C3D20R
57041,57041,57241,57233,57033,257041,257241,257233,257033,57141,57237,
57133,57037,257141,257237,257133,257037,157041,157241,157233,157033
*ELGEN,ELSET=SECT57
57041,4,8,8,4,200,200
*ELEMENT,TYPE=C3D20R
57033,57033,57233,61233,61033,257033,257233,261233,261033,57133,59233,
61133,59033,257133,259233,261133,259033,157033,157233,161233,161033
*ELGEN,ELSET=SECT5765
57033,2,4000,4000,4,200,200
*ELEMENT,TYPE=CIN3D12R
41941,41841,241841,241833,41833,141841,241837,141833,41837,41941,241941,
241933,41933
*ELGEN,ELSET=INF41
41941,4,8,8
*ELEMENT,TYPE=CIN3D12R
29933,29833,229833,225833,25833,129833,227833,125833,27833,29933,229933,
225933,25933
*ELGEN,ELSET=INF2541
29933,4,4000,4000
*ELEMENT,TYPE=CIN3D12R
25933,25833,225833,225841,25841,125833,225837,125841,25837,25933,225933,
225941,25941
*ELGEN,ELSET=INF25
25933,4,8,8
*ELSET,ELSET=INF
INF41,INF2541,INF25
*ELSET,ELSET=E1
RINGS,SECT9,SECT25,SECT41,SECT57,SECT19,SECT925,
SECT2541,SECT4157,SECT5765,INF
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E1,SHIFT NODES=200000,NEW SET=E2
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E2,SHIFT NODES=200000,NEW SET=E3
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E3,SHIFT NODES=200000,NEW SET=E4
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E4,SHIFT NODES=200000,NEW SET=E5
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E5,SHIFT NODES=200000,NEW SET=E6
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E6,SHIFT NODES=200000,NEW SET=E7
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E7,SHIFT NODES=200000,NEW SET=E8
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E8,SHIFT NODES=200000,NEW SET=E9
*ELCOPY,ELEMENT SHIFT=200000,OLD SET=E9,SHIFT NODES=200000,NEW SET=E10
*ELSET,ELSET=EALL
E1,E2,E3,E4,E5,E6,E7,E8,E9,E10
*MATERIAL,NAME=STEEL
*ELASTIC
30.E6,0.3
*SOLID SECTION, MATERIAL=STEEL, ELSET=EALL
*MPC
TIE,TIP01,TIP02
TIE,TIP11,TIP12
TIE,TIP21,TIP22
TIE,TIP31,TIP32
TIE,TIP41,TIP42
TIE,TIP51,TIP52
TIE,TIP61,TIP62
TIE,TIP71,TIP72
TIE,TIP81,TIP82
TIE,TIP91,TIP92
TIE,TIP101,TIP102
TIE,TIP111,TIP112
TIE,TIP121,TIP122
TIE,TIP131,TIP132
TIE,TIP141,TIP142
```

```
TIE,TIP151,TIP152
TIE,TIP161,TIP162
TIE,TIP171,TIP172
TIE,TIP181,TIP182
TIE,TIP191,TIP192
TIE,TIP201,TIP202
*NSET,NSET=N571,GENERATE
57833,2057833,100000
*NSET,NSET=N572,GENERATE
57865,2057865,100000
*NSET,NSET=N65,GENERATE
65833,2065833,100000
*MPC
TIE,N572,N571
TIE,N65,N571
*NSET,NSET=N18,GENERATE
1833,2001833,100000
*NSET,NSET=N98,GENERATE
9833,2009833,100000
*MPC
TIE,N18,N98
*MPC
TIE,Y1_1,Y01
TIE,Y2,Y0
TIE,Y31,Y01
TIE,Y4,Y0
TIE,Y51,Y01
TIE,Y6,Y0
TIE,Y71,Y01
TIE,Y8,Y0
TIE,Y91,Y01
TIE,Y10,Y0
TIE,Y111,Y01
TIE,Y12,Y0
TIE,Y131,Y01
TIE,Y14,Y0
TIE,Y151,Y01
TIE,Y16,Y0
TIE,Y171,Y01
TIE,Y18,Y0
TIE,Y191,Y01
TIE,Y20,Y0
*BOUNDARY
N0,3,3
N20,1,1
Y0,1,1
Y0,3,3
*STEP
APPLY PRESSURE LOAD
*STATIC
1.0,1.0
*DLOAD
TOP9,P4,10.
*CONTOUR INTEGRAL,CONTOURS=8,OUTPUT=BOTH
T0,0.70710678,-0.70710678,0.
T1,0.70492701,-0.70710678,0.055478959
T2,0.69840112,-0.70710678,0.11061587
T3,0.68756936,-0.70710678,0.1650708
T4,0.67249851,-0.70710678,0.21850801
T5,0.65328148,-0.70710678,0.27059805
```

```
T6,0.63003676,-0.70710678,0.32101976
T7,0.60290764,-0.70710678,0.36946228
T8,0.5720614,-0.70710678,0.41562694
T9,0.53768821,-0.70710678,0.45922912
T10,0.5,-0.70710678,0.5
T11,0.45922912,-0.70710678,0.53768821
T12,0.41562694,-0.70710678,0.5720614
T13,0.36946228,-0.70710678,0.60290764
T14,0.32101976,-0.70710678,0.63003676
T15,0.27059805,-0.70710678,0.65328148
T16,0.21850801,-0.70710678,0.67249851
T17,0.1650708,-0.70710678,0.68756936
T18,0.11061587,-0.70710678,0.69840112
T19,0.055478959,-0.70710678,0.70492701
T20,0.,-0.70710678,0.70710678
*CONTOUR INTEGRAL,CONTOURS=8,OUTPUT=BOTH,
TYPE=K FACTORS
T0,0.70710678,-0.70710678,0.
T1,0.70492701,-0.70710678,0.055478959
T2,0.69840112,-0.70710678,0.11061587
T3,0.68756936,-0.70710678,0.1650708
T4,0.67249851,-0.70710678,0.21850801
T5,0.65328148,-0.70710678,0.27059805
T6,0.63003676,-0.70710678,0.32101976
T7,0.60290764,-0.70710678,0.36946228
T8,0.5720614,-0.70710678,0.41562694
T9,0.53768821,-0.70710678,0.45922912
T10,0.5,-0.70710678,0.5
T11,0.45922912,-0.70710678,0.53768821
T12,0.41562694,-0.70710678,0.5720614
T13,0.36946228,-0.70710678,0.60290764
T14,0.32101976,-0.70710678,0.63003676
T15,0.27059805,-0.70710678,0.65328148
T16,0.21850801,-0.70710678,0.67249851
T17,0.1650708,-0.70710678,0.68756936
T18,0.11061587,-0.70710678,0.69840112
T19,0.055478959,-0.70710678,0.70492701
T20,0.,-0.70710678,0.70710678
*CONTOUR INTEGRAL,CONTOURS=8,OUTPUT=BOTH,
TYPE=T-STRESS
T0,0.70710678,-0.70710678,0.
T1,0.70492701,-0.70710678,0.055478959
T2,0.69840112,-0.70710678,0.11061587
T3,0.68756936,-0.70710678,0.1650708
T4,0.67249851,-0.70710678,0.21850801
T5,0.65328148,-0.70710678,0.27059805
T6,0.63003676,-0.70710678,0.32101976
T7,0.60290764,-0.70710678,0.36946228
T8,0.5720614,-0.70710678,0.41562694
T9,0.53768821,-0.70710678,0.45922912
T10,0.5,-0.70710678,0.5
T11,0.45922912,-0.70710678,0.53768821
T12,0.41562694,-0.70710678,0.5720614
T13,0.36946228,-0.70710678,0.60290764
T14,0.32101976,-0.70710678,0.63003676
T15,0.27059805,-0.70710678,0.65328148
T16,0.21850801,-0.70710678,0.67249851
T17,0.1650708,-0.70710678,0.68756936
T18,0.11061587,-0.70710678,0.69840112
T19,0.055478959,-0.70710678,0.70492701
T20,0.,-0.70710678,0.70710678
```

```
*FILE FORMAT, ASCII
*EL FILE
JK
*ENDSTEP
```

Published with MATLAB® R2015a

```
type('2000.inp')
```

```
*HEADING
 VERIFICATION OF ABAQUS INCREMENT START RECORD OUTPUT TO MATLAB (RECORD KEY 2000)
*NODE
 1,
 2,10.
 3,10.,5.,
 4,0.,5.,
*ELEMENT,TYPE=CPE4,ELSET=ONE
 1,1,2,3,4
*SOLID SECTION,ELSET=ONE,MATERIAL=SIMPLE
 1.,
*MATERIAL,NAME=SIMPLE
*HYPERFOAM,N=3,TEST DATA INPUT,POISSON=0.,MODULI=INSTANTANEOUS
*UNIAXIAL TEST DATA
 -39. ,   -.05
 -57. ,   -.10
 -66. ,   -.15
 -72. ,   -.20
 -78. ,   -.25
 -84. ,   -.30
 -90. ,   -.35
 -96. ,   -.40
 -102. ,   -.45
 -108. ,   -.50
 -115. ,   -.55
 -130. ,   -.60
 -150. ,   -.65
 -185. ,   -.70
 -260. ,   -.75
 -400. ,   -.80
*VISCOELASTIC,TIME=PRONY
 0.5,0.5,3.
*BOUNDARY
 1,1,2
 2,2,2
 4,1,1
*NSET,NSET=OUT
 2
*STEP,NLGEOM,INC=200
 STEP-1
*VISCO,CETOL=.01,STABILIZE
 2.,10.,,10.,
*BOUNDARY
 3,1,1,2.
 2,1,1,2.
*FILE FORMAT, ASCII
*NODE FILE,NSET=OUT
 VF
*END STEP
```