# OOFEM Extractor Input Data Format

Bořek Patzák

Czech Technical University
Faculty of Civil Engineering
Department of Structural Mechanics
Thákurova 7, 166 29 Prague, Czech Republic

January 2, 2018

"Extractor.py" is a package for automatic extraction (and checking) of user-defined values from OOFEM output file. The values to be extracted are defined in extractor input file. The tool can operate in two modes:

- Extraction mode, where for each solution step, the tool finds specific values and prints them on output, one row per each solution step. In particular columns are the extracted values, preserving their order as specified in input file.

- Checking mode, where the specific values are located in oofem output file and are compared to the values given. It the difference is larger than given tolerance, an error is issued.

The program parameters are following

```
extractor [-c] -f inputFileNamePath
```

The *inputFileNamePath* is the path to input file, containing description of user-defined values to be extracted or checked. The **-f** option is compulsory. If not supplied, the program prints usage information and exits. The **-c** flag turn on the checking mode. If not set, the extraction mode is used by default.

The format of extractor input file consists of two compulsory sections, the **source file section**, followed by the **value records section**, that determines the values to be extracted or checked.

**Source file section**   should be the first line of input file and should contains valid path to OOFEM output file, from which the data will be extracted.

Any lines between **source file section** and **value records section** are ignored. As the **source file section** is common for both oofem and extractor input files and all remaining extractor records start with '#' (considered as comment in oofem input file) the oofem and extractor input files can be merged together.

**Value records section**   describes the values to be extracted or checked. The general format of this section is following:

```
#%BEGIN_CHECK% [tolerance #]
#(DOFMAN|NODE)  {tStep #} number # dof # unknown [dtfpva] {value #}
#ELEMENT   {tStep #} number # [irule #] gp # keyword # component # {value #}
#BEAM_ELEMENT   {tStep #} number # keyword # component # {value #}
#REACTION  {tStep #} number # dof # {value #}
#LOADLEVEL {tStep #} {value #}
#%END_CHECK%
```

The #%BEGIN_CHECK% and #%END_CHECK% records are compulsory. The records in "[ ]" are optional. The notation (a|b) means "a" or "b". The records in "" are required only when in checker mode.

The order of records is optional, but extracted values are printed in columns according to their order in input file. The stand-alone # after *keyword* indicates the int or float value to be substituted, depending on the context. The value of *unknown* parameter is a single char, determining the physical mode (and type) of extracted value. For example, in structural analysis 'd' stands for displacement, 'v' for velocity, and 'a' for acceleration. Value parameter defines the value to which the extracted one is compared. The meaning of individual records is following:

- `#(DOFMAN|NODE) {tStep #} number # dof # unknown [dtfpva] {value #}`

  This records allow extraction of arbitrary unknown associated with a Degree of freedom (DOF) in a particular node or dof manager. The *number* value represent node number, the *dof* value is the DOF number. The DOF numbering is determined by domain type or by user defined DOF ordering in specific DofManager (see OOFEM Input format specification). The *unknown* value is represented by single character, which determine the mode of the extracted value. In checker mode, the *tStep* parameter determines the solution time (float value) and *value* parameter defines the expected value for given DOF at given time.

- `#ELEMENT   {tStep #} number # [irule #] gp # keyword # component # {value #}`

  Record representing the value associated to an integration point of an element. The *number* value determines the element number, the *gp* determines the integration point number of the element. The *irule* allows to determine the integration rule, if not specified, the "1" is substituted by default, which uses the first intergartion rule on element. Generally, there are many values associated with each integration point (for example, strain and stress vectors for structural analysis). Their order depends on problem being solved and constitutive model used. Each value is characterized by its keyword and index, that determines its position in an array associated to this keyword. The *keyword* parameter is a string, that determines the actual keyword and *component* parameter determines the position of extracted value in an array of values associated to given keyword.

- `#BEAM_ELEMENT {tStep #} number # keyword # component # {value #}`

  Records representing the unknown associated to a certain beam element (beam elements formulated without integration points, with exact ana-

lytical integration, the unknowns are provided as end forces and and-displacements. Currently Beam2d and Beam3d elements).

- `#REACTION {tStep #} number # dof # {value #}`

  Represent reaction force associated to certain DOF subjected to Dirichlet boundary condition. The DOF of interest is determined by number of its DofManger using *number* value and its number, using *dof* value.

- `#LOADLEVEL {tStep #} {value #}`

  Represent reached load level in particular solution step.

**Example (for checker mode):**

```
patch100.out
#%BEGIN_CHECK% tolerance 1.e-4
#ELEMENT tStep 1 number 1 gp 1 component 1 keyword "stresses" value -8.3333e+00
#ELEMENT tStep 1 number 1 gp 1 component 1 keyword "strains"  value -5.2083e-01
#DOFMAN  tStep 1 number 4 dof 1 type d value -1.5625
#%END_CHECK%
```