Open source tools for MELCOR: 2014 update and application examples.

Petr Vokáč, vok@ujv.cz
ÚJV Řež, a.s.

6th EMUG, 15.-16. 4. 2014, ÚJV Řež, a.s.

Content

- Overview of available tools
- Examples of application:
  - `endf.py`: Analysis of simulated sample activity
  - `tubes.py`: Preprocessing of piping drawings for CVH (FL, HS) input using VTK:
    - VVER-440/213 degasification system
    - VVER-440/213 TL11 ventilation system
    - Test of STL import
    - VTK pitfalls
- Content of the 2014 distribution CD

  Note: Tools are available only CD or flash disk, possibly E-mail — — ÚJV web site can no longer be used for any file sharing

Overview of available tools

- `readptf.exe` — data extraction from MELPTF — no changes since 2012 EMUG
  problem: when the last variable is a vector, I do not know how to evaluate dimension of the vector — any hint?
- `tranptf.exe` — join or split MELPTF — no changes
  should similar tool exist for edf files? is the description of edf files available?
- `browseptf.py` — GTK based GUI for readptf.exe and gnuplot — no changes
- `cor-volf.py` — PyX based core snapshot plot — no changes
- `cor-tlh.py` — PyX based snapshot of temperature in the lower plenum and vessel wall — no changes
- `endf.py` — ORIGEN output to MELCOR DCH input preprocessing
  - minor modifications
  - added γ lines from ENDF database
- `tubes.py` — Preprocessing of piping drawings for CVH (FL, HS) input using VTK — new in the 2014 distribution CD

endf.py

Changes:

- line output for tabular function input shortened to 70 characters — 80 is possible according to MELCOR manual, however random errors were observed in total decay heat — with 70 characters it worked ok in all our inputs
  — maybe this issue concerns other tabular function as well —
- added class ConvMass for recalculation of ORIGEN outputs to different U irradiated mass (this mass should be the same for all data used in endf.py pre-processing)
- check of the user input: requested time should be monotonous
- empty lines allowed in activity or mass input files (occured when these files were prepared by hand)
- γ lines from ENDF database via conversion to Python pickle file
  (problem: reading of large pickle file is quite slow)
- activity output to csv file compatible with NUCLEONICA[a] (for simulation of measured γ spectra)
- spe2txt.py — utility to convert output spectrum from NUCLEONICA spe format to space separated column format

[a] see www.nucleonica.com

Simulated activity measurement of a sample taken from the containment

Purpose: what can be evaluated from a sample of atmosphere or water from the containment taken during a postulated accident depending on:

- phase of core degradation
- time of sampling (from SCRAM)
- delay of sample analysis from sampling

First estimate based on simple assumption: activity of the sample is given:

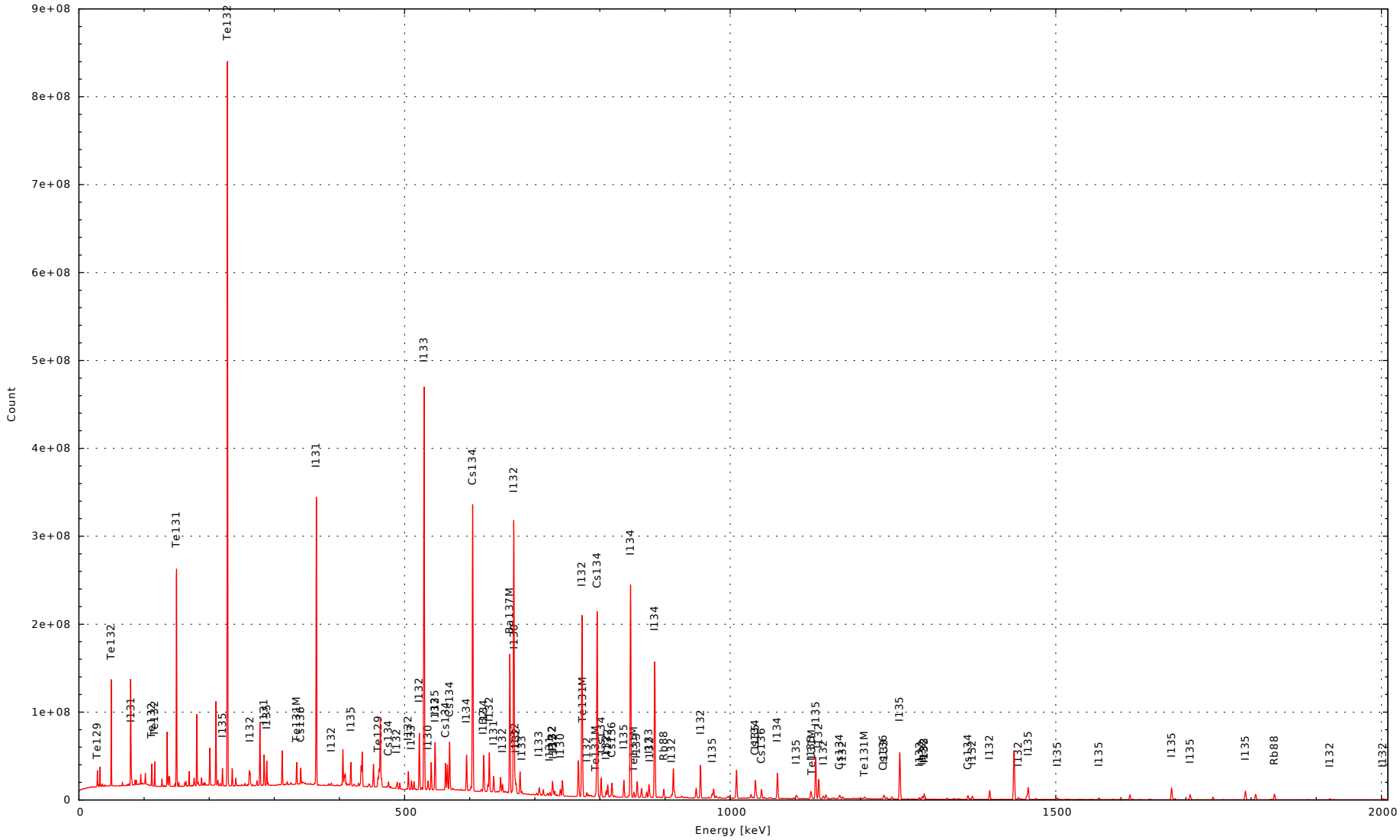
$$A_i(\text{sample}) = k_i \cdot A_i(\text{inventory})$$

where where A_i is calculated using `endf.py` and k_i is chosen different for volatiles, semi-volatiles, non-volatiles ...
Measured γ spectrum can be simulated using NUCLEONICA.

Next step would be to use MELCOR output to evaluate activity on the air filter or in sampled liquid. Neglecting decay chains, most of relevant processes are simulated:

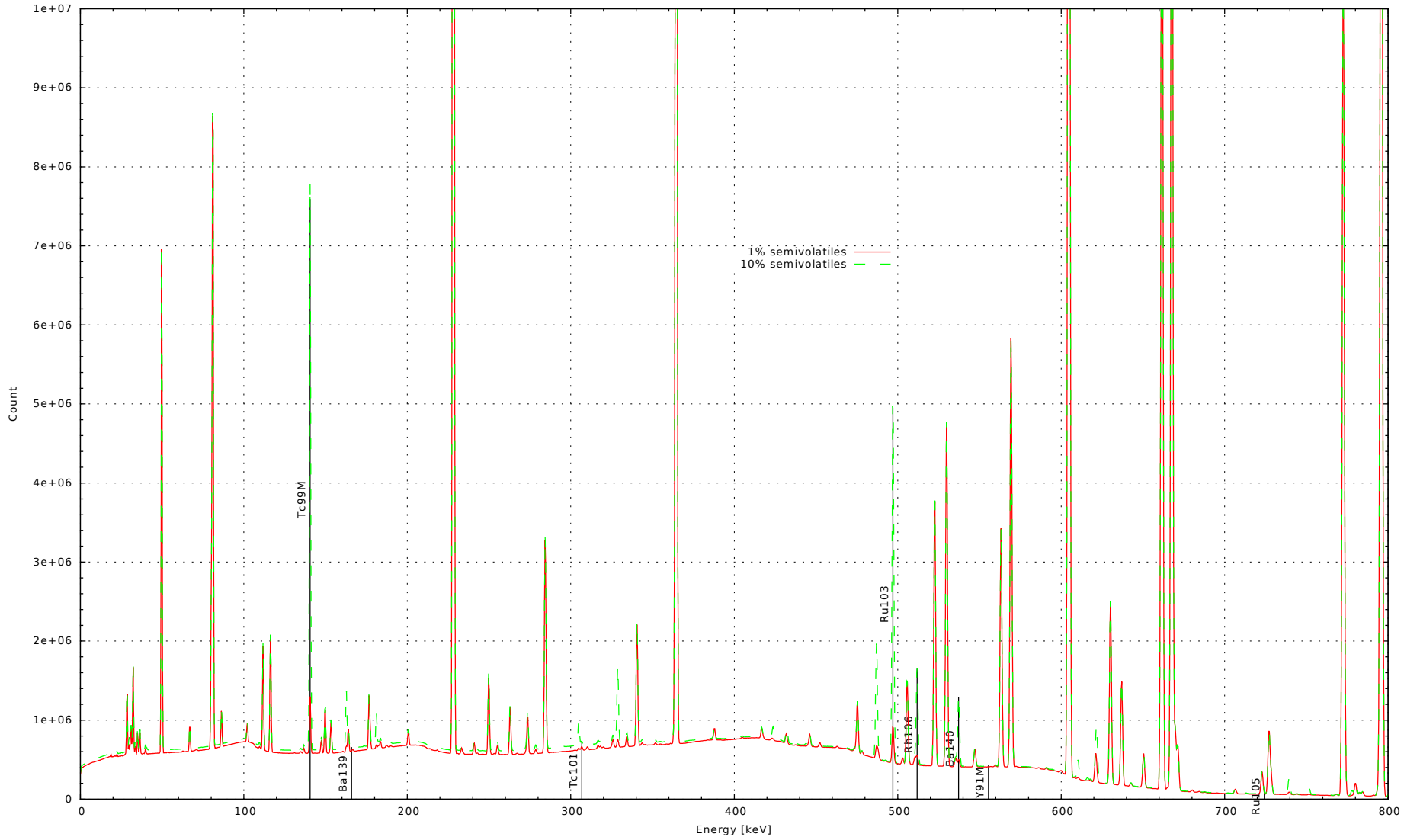
release from the core $\rightarrow$ retention in RCS $\rightarrow$ deposition in the containment

Missing phenomena: deposition from liquid



Example evaluation of γ spectrum of a sample with volatile aerosols using NUCLEONICA

Open source tools for MELCOR: 2014 update and examples



Example evaluation of γ spectrum of a sample with volatile aerosols plus 1% or 10% of semivolatiles added to the mixture using NUCLEONICA

Simulated sample spectra — preliminary conclusions

- after adding fraction of less volatile species, simple "first look" evaluation of spectrum using detector efficiency and FWHM on γ energy of less volatile isotope on background of Compton scattering of volatiles might be sufficient.
- key semi-volatile or non-volatile isotopes can be identified and minimum detectable fraction evaluated depending on time since SCRAM and delay from the containment media sampling to sample analysis

tubes.py

Purpose: to simplify evaluation of drawings of pipes and ducts of old plants or experiments via:

- 3D visulalization of the model
- calculation of elevation vs. volume tables for CVH input
- improvement of QA of input models

Implementation:[a] Python 2.7 + VTK 5 or 6 + SciPy

Tested on: Linux, Mac OS X

Works with:

- stright circular pipe
- circular pipe elbow (approximated as a section of annuloid)
- elliptic canopy
- rectangular duct (still under development)
- STL file exported from CAD (I am currently trying to find out how to use it to calculate internal volume)

[a] see: www.python.org, www.vtk.org, www.scipy.org

Example 1: VVER-440/213 degassification system

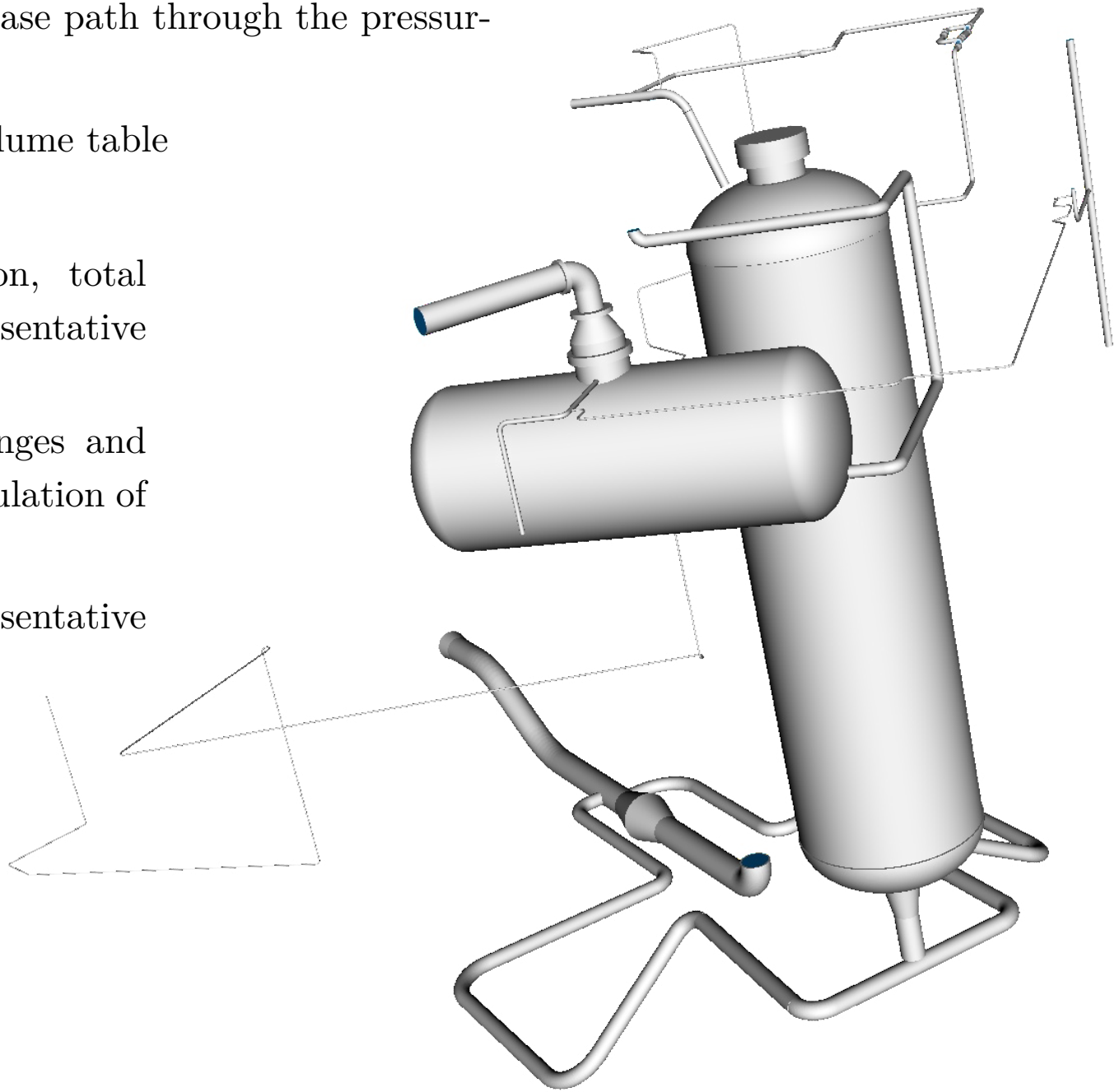
Purpose: to evaluate retention of fission products in the reactor degassification system during an accident at shutdown before removal of the reactor lid

Assumptions:

- reactor is at atmospheric pressure
- decay heat removal is via natural circulation in two primary loops to the secondary coolant ("liquid-liquid" mode)
- opened leak paths from the reactor primary circuit:
  - via central control assembly driver to reactor degassification line and to the pump deck
  - via pressurizer :
    - to the quench tank and through TL72 ventilation system to the SG box
    - to the pump deck
  - via the hose connected to the hot leg of primary loop in reserve for reactor water level measurement to the SG box
- initiating event: loss of heat transfer to secondary circuit

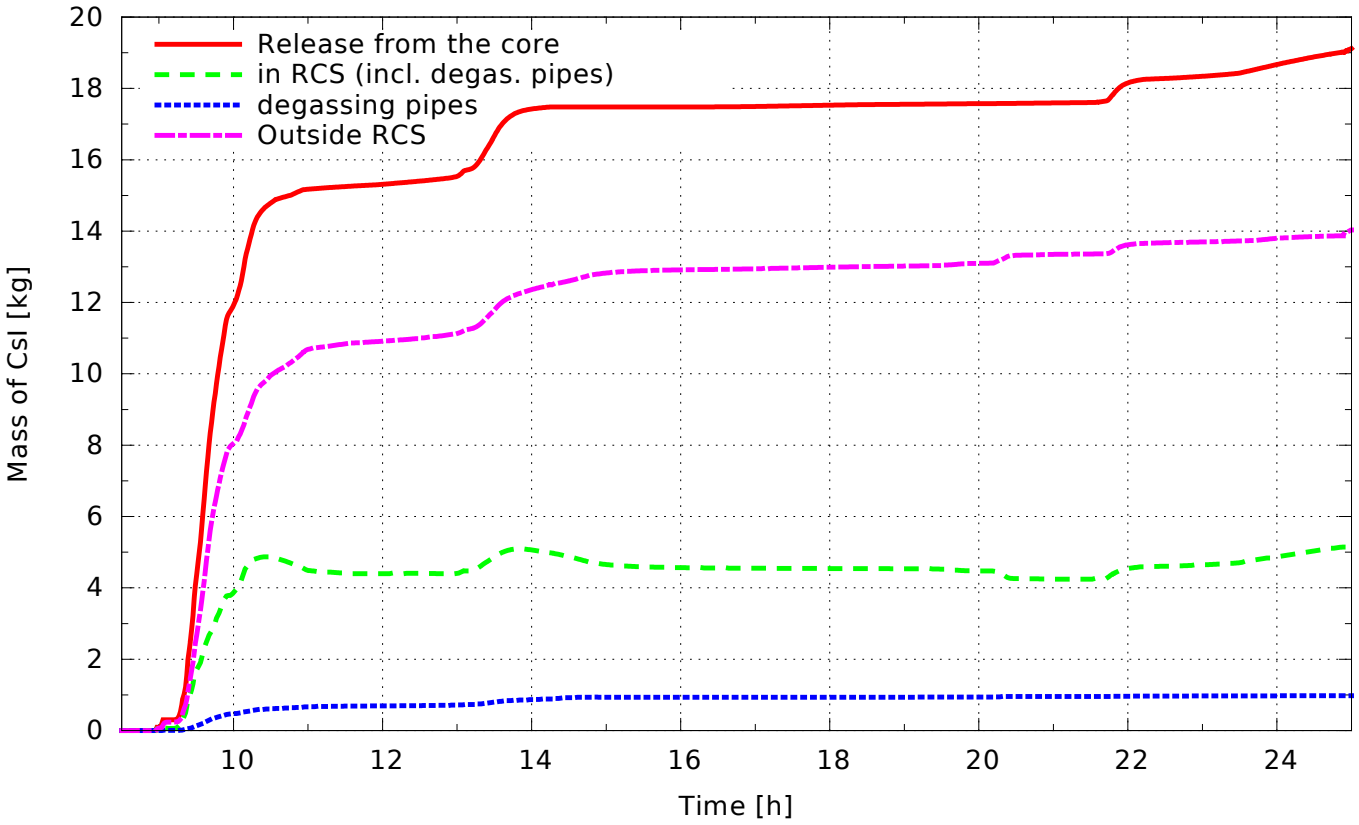
Example visualization of the release path through the pressurizer, it was used to prepare:

- CVH input: elevation vs. volume table
- FL input:
  - inlet and outlet elevation, total length of junction, representative diameter
  - number of direction changes and diameter changes for calculation of friction coefficients
- HS input: selection of representative HS for FP deposition



Example 1: VVER-440/213 degassification system

Result of simulations: surprisingly low deposition in leak paths, e.g. for CsI:



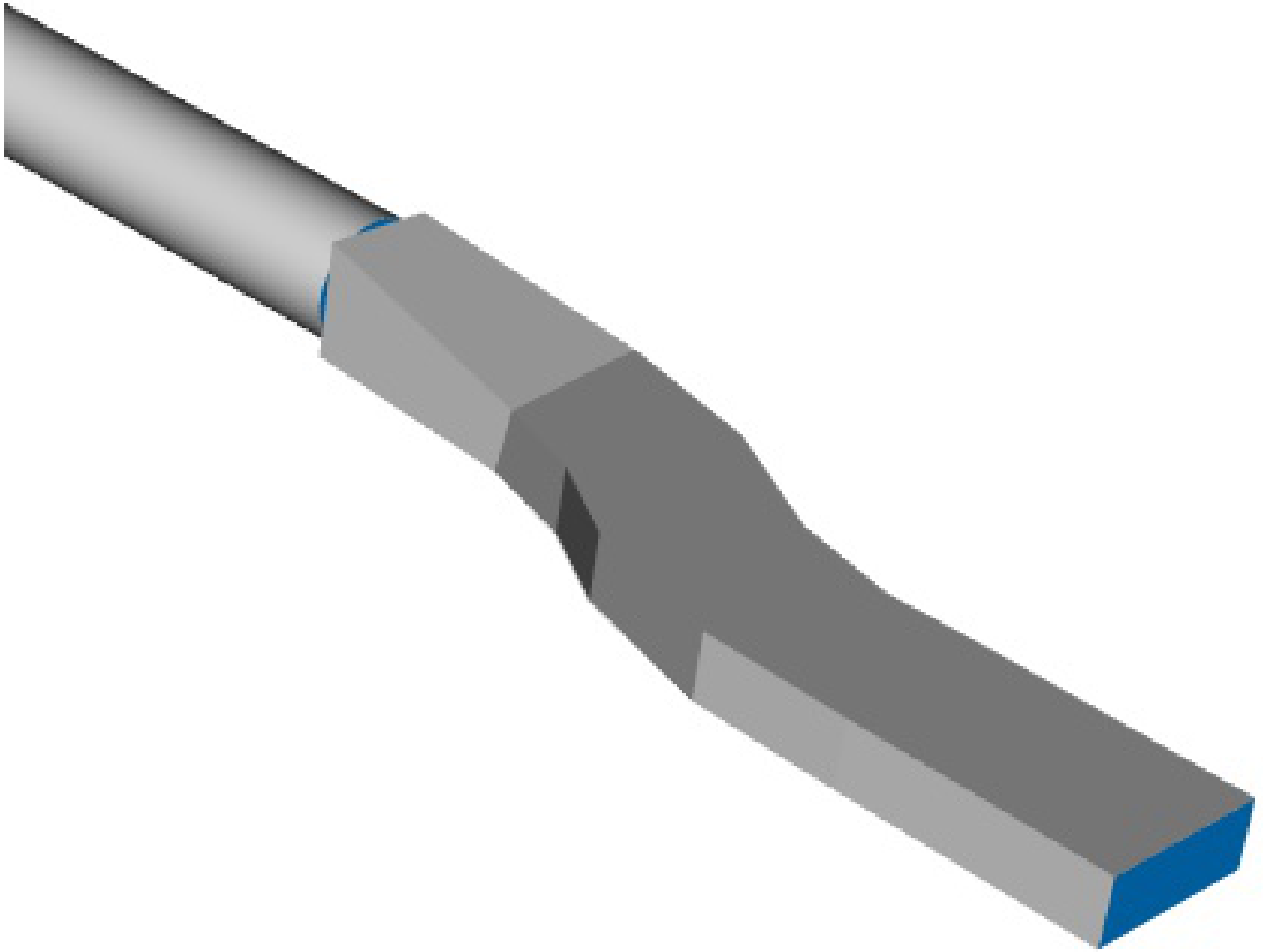
Possible explanation:

- Absence of turbulent deposition model in MELCOR 1.8.6?
- Washout by condensate? (or deposition to pool instead of HS)
- Because hygroscopic model not activated in the input?
- ...

Simulations were quite complicated and CPU demanding — sensitivity study of aerosol parameters was not possible

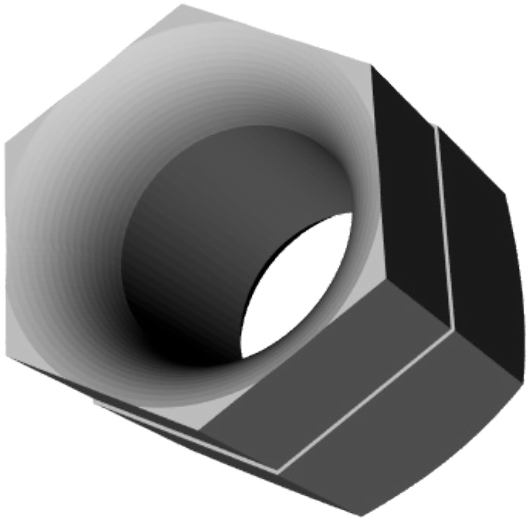
Example 2: VVER-440/213 TL11 ventilation system

Piece of ventilation duct showing rectangular section:

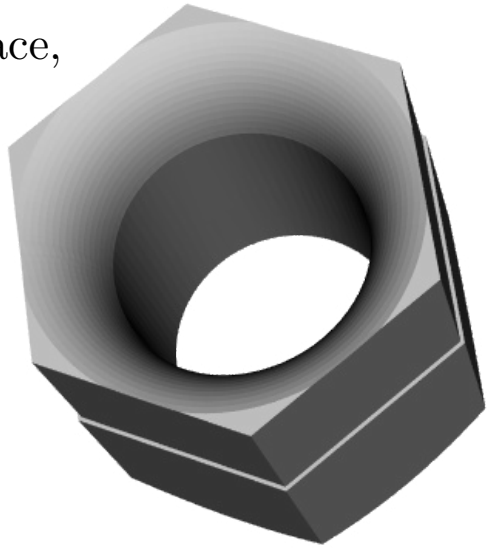


Tests of STL import

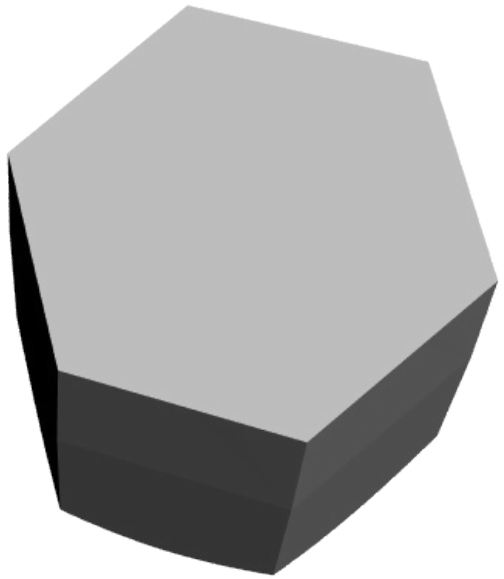
Example of "correctly" prepared STL file:
– it has internal surface,
– it has no holes in the surface,
– it is about convex



Original data from the STL file



Triangle filter applied: it allows to calculate volume of the wall, V_w



Delaunay3D filter applied: it allows to calculate total volume of the body (convex wrap), V_t

Then volume of fluid, V_f inside the object can be calculated as:

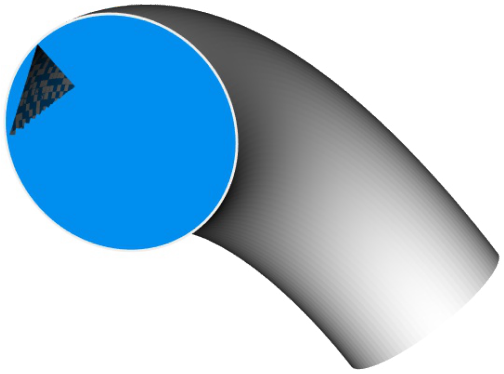
$$V_f = V_t - V_w$$

and the elevation-volume table can be calculated using ClipClosedSurface filter with water level plane.

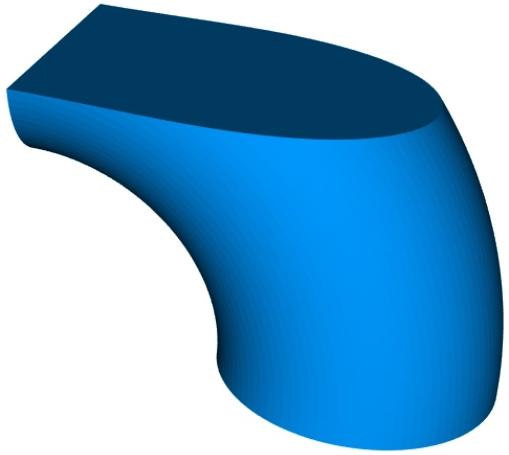
VTK pitfalls

Volume of an object is calculated by aproximating surface by triangles — circular objects should be aproximated with sufficient resolution, however for rendering resolution should be decreased to make visualization faster

Surface should be closed, othervise calculated volume is wrong. Sometimes an error occurs due to numerical precision, mainly for elbows. Two values are calculated by VTK mass property: `volume` and `projected volume`, these should be equal when everything is OK.



Hole in the surface: `volume` and `projected volume` differs. Volume is calculated incorrectly. Small change in the elbow parameters may cure the problem.



Error during “water level decrease”. `projected volume` is negative but equal to `volume` in absolute value. Value of `volume` seems to be correct.

Content of the 2014 distribution CD

- 140407 update, folder **archives-140407**
  tubes-140407.tar.gz: Pipes and ducts preprocessing using VTK
  tubes-tests-140407.tar.gz: Test inputs for pipes and ducts preprocessing using VTK
  endf2012open-140407.tar.gz: DCH pre-processing for MELCOR 1.8.6
  (Data file with gamma radiation properties `spectra.pickle` is not included)
  presentations-140407.tar.gz: Meltools presentations at EMUG
- 130422 update, folder **archives-130422**
  endf2012open-130422.tar.gz: DCH pre-processing for MELCOR 1.8.6
- 120523 update, folder **archives-120523**
  executables-120523.tar.gz: readptf&tranptf executables
  readptf-120523.tar.gz: readptf&tranptf sources
  doc-120523.tar.gz: documentation and global makefile
- 120504 update, folder **archives-120504**
  browseptf-120504.tar.gz: Python GTK GUI for readptf.exe and gnuplot
  pyc-120504.tar.gz: Python modules
  preproc-120504.tar.gz: preprocessing utilities
  melpyx-120504.tar.gz: graphic utilities using PyX
  melpycalc-120504.tar.gz: misc post-processing scripts
  examples.tar.gz: Examples of post-processing. About 200 MB. It includes MELCOR sample outputs.

Thank you for your attention

Any questions? (anwers?)