UNED

ETS de Ingenieros Industriales

GEOUNED 1.0 User's guide

Authors	Affiliation & email
Juan Pablo Catalán	ETSI Industriales UNED, jpcatalan@ind.uned.es
Patrick Sauvan	ETSI Industriales UNED, psauvan@ind.uned.es
Juan García	ETSI Industriales UNED, jgarcia@ind.uned.es

September 2023

Table of Contents

List of Figures

List of Tables

Major improvements in Version 0.9.8.2

- Interface with user changed.
- Compatibility with all FreeCAD releases (up to 20.x).
- Run in Windows and Linux
- Reverse conversion MCNP to CAD
- Add of new CARD options.
- Advanced cell definition simplification.
- Sort void enclosure cell in the MCNP file.
- Minor bugs fixed

Major improvements in Version 1.0

- Output geometry format for OpenMC.
- Minor bugs fixed

Changes in Version 1.0

- “mcnpfile" keyword in FILE section substituted by “geometryName” keyword
- New parameter to change the tolerance value of the FreeCAD geometry splitting function.

1. Introduction

Most of the Monte Carlo (MC) radiation transport codes use a special kind of CSG geometries to define physical system to be simulated. In modern engineering, models are described using 3D CAD tools that are not directly compatible with the CSG approach followed by MC codes. GEOUNED is a conversion tool that generates the CSG MC geometry from the information stored in the CAD models. The code is based on FreeCAD software that is used mostly as python API for OPENCASCADE CAD engine. GEOUNED is structured as a python module package but is designed to its use as standalone software. This means, GEOUNED is not designed to provide easy access to the variables, classes and functions used inside the module. This manual describes the basic use of GEOUNED for standard user understanding that as those interested in convert models from STEP format to MCNP or openMC geometry format.

The first part of the manual deals with the installation of the module in Windows OS. This step is simple for people with some experience in the use of python modules but some advice is given to save time for new users. The second part is dedicated to the call GEOUNED as part of a python script and how to introduce some parameters that can be modified by the user. Special attention is given to the unique novelties implemented in the automatic void generation of GEOUNED like user defined enclosures and envelopes. Finally, some very simple examples are provided to illustrate the use of the code.

2. GEOUNED installation procedure in Windows

From version 0.18 FreeCAD uses python 3 and so GEOUNED source code is based also on it. FreeCAD python libraries depends on the specific version of python 3 and its updating is less frequent than python one making that direct python installation could not work. FreeCAD Windows installation includes its own version of python that can be used to run GEOUNED module and avoid this problem. This python executable is located in the subfolder /bin/ within the FreeCAD folder which is selected during the installation process (by default is C:\Program Files\FreeCAD 0.xx\ in W10).

The following steps explain in detail how to install and run GEOUNED code:

<u>1st step: Install FreeCAD</u>

FreeCAD can be downloaded from https://www.freecad.org/ . This version of GEOUNED support any FreeCAD version (compatibility version checked up to FreeCAD 0.20.2 Release).

<u>2nd step: Set GEOUNED execution PATH</u>

GEOUNED code is compound by the main executable "geouned" and its dependencies (the GEOUNED modules). The geouned executable, is a executable python script. For this script to run properly the path to the FreeCAD libraries and GEOUNED modules should be set.

There is two ways to set the paths:

- setting the path in the Windows or Linux python environment variable "PYTHONPATH",
- editing the executable script "geouned" and setting the correct path in the "FreeCADPath" and "GEOUNEDPath" variables.

3rd step: Run GEOUNED

Once the paths are set, geouned run like a python script. It can be called with as a python script, or directly as an executable in Linux if the location of the python executable has been defined in the first line of the geouned executable script.

In the zipped file with the GEOUNED python module package, there are folders with examples (e.g. ~Examples/case1/). This folder contains 3 files, the original *stp* file to be converted, a *txt* file with the materials definition and a config file called *config.ini*. The "config.ini" file is the file in which all input parameters (differing from default values) for the GEOUNED execution are set. python file is a simple example that contains a call to GEOUNED module. "config.ini" is a default name for the file, the user can use any name for this file. The configuration filename must be indicated at the execution of geouned.

In Windows, you have to open a cmd window in the folder that contains this file and use this command to run the code (again assuming default installation folder for FreeCAD):

```
C:\..\folder_Main.py>"C:\Program Files\FreeCAD 0.18\bin\python.exe" geouned
config.ini
```

In Linux :

> ./geouned config.ini

To be able to run geouned as an executable the geouned file has to be set to execution mode (chmode +x).

In the config file input CAD and MCNP output filenames should be defined. The reverse conversion mode (MCNP to CAD) is run using the option "-r" in the execution command line. In this mode is used the config file should be different from the one used for direct conversion.

3. Input parameters

3.1. Configuration file

The parameters that rule the conversion process are set in the configuration file and are described in this section. For the direct or reverse mode different config file should be used.

For direct conversion the parameters are divided in sections. There are 5 sections: "Files", "Parameters", "Tolerances", "MCNP_Numeric_Format", "Options". The sections in the configuration file are written as [*section*]. Sections can be omitted in the configuration file, in this case default values are taken. The only section that must be present is the section [Files].

For the reverse conversion four sections are defined: "Setting", "Levels", "Cells" and "Materials".

An example of the configuration file is:

```
[Files]
Title = my MCNP model
stepFile = myModel.stp
matFile = materials.txt
geometryName = myModel

[Parameters]
startCell = 10000
startSurf = 10000
```

Table 1. List of section of the direct conversion configuration file.

Section	Description
Files	Section where input and output file names are defined. The title of the MCNP model is also set in this section.
Parameters	Set the parameters to produce the MCNP output.
Tolerances	Change the default tolerances parameters
MCNP_Numeric_Format	Change default numeric format of surface definition in MCNP output
Options	Change module/parameters used in the geouned conversion process.

Table 2. List of section of the reverse conversion configuration file.

Section	Description
Setting	Set name of input and output files as well as the size of the region where the solid is defined
Levels	Select starting universe level and maximum level depth for solid generation
Cells	Filter cells to convert to CAD by cell label
Materials	Filter cells to convert to CAD by material label

3.2. Direct conversion process

3.2.1. Files.

The file section is used to define the name of input and output files, and to select which output format will be generated (MCNP and/or OpenMC). This section must be defined in the configuration file. The only mandatory requested entry in this section is the name of the step file.

Table 3. List of entries of the Files section.

Keyword	Default Value	Description
stepFile	None	Name of the CAD file (in STEP format) to be converted.
matFile	None	Name of the file with the materials information.
geometryName	Step filename without ".stp" extension	Base name of the output file(s)
title	Step filename without ".stp" extension	Title of the model
outFormat	mcnp	Format for the output geometry. Available format are: mcnp, openMC_XML, openMC_PY. Several output format can be written in the same geouned run.

If the material file is not provided, the density of the material defined in the STP model is set as -100 (for MCNP output file).

3.2.2. Parameters

The parameters section is the section where specific behavior of the code is specified by the user to obtain the desired output file.

Table 4. List of parameters to be passed to GEOUNED through set method.

Keyword	**Default Value**	**Description**
startCell	1	Starting cell numbering label
startSurf	1	Starting surface numbering label
UCARD	0	Write universe card in the cell definition with the specified universe number (if value = 0 Universe card is not written)
volCARD	True	Write the CAD calculated volume in the cell definition using the VOL card
volSDEF	False	Write SDEF definition and tally of solid cell for stochastic volume checking.
dummyMat	False	Write dummy material definition card in the MCNP output file for all material labels present in the model. Dummy material definition is "MX 1001 1".
voidMat	(0, None, '')	Assign a material defined by the user instead of void for cells without material definition and the cells generated in the automatic void generation. The format is a 3 valued tuple (mat_label, mat_density, mat_description). Example (100,1e-3,'Air assigned to Void')
sortEnclosure	False	If enclosures are defined in the CAD models, the voids cells of the enclosure will be located in the output file in the same location where the enclosure solid is located in the CAD solid tree.
compSolids	True	Join subsolids of STEP file as a single compound solid. Step files generated with SpaceClaim have not exactly the same level of solids as FreeCAD. It may a happened that solids defined has separated solids are read by FreeCAD as a single compound solid (and will produce only one MCNP cell). In this case compSolids should be set to False.
cellSummaryFile	False	Write an additional file with information on the CAD cell translated
cellCommentFile	False	Write an additional file with comment associated to each CAD cell in the MCNP output file.
exportSolids	''	Export CAD solid after reading. The execution is stopped after export, the translation is not carried out.
voidGen	True	Generate voids of the geometry.
simplify	no	Simplify the cell definition considering relative surfaces position and using Boolean logics. Available options are: - ***no*** : no optimization - ***void*** : only void cells are simplified. Algorithm is faster but the simplification is not optimal. - ***voidfull*** : only void cells are simplified with the most optimal algorithm. The time of the conversion can be multiplied by 5 or more. - ***full :*** all the cells (solids and voids) are simplified.

minVoidSize	100	Minimum size of the edges of the void cell. Units are in mm.
maxSurf	50	Maximum number of surfaces allowed in void cell definition. This number is the number of different surface label, if this label is used several times in the void definition, it will be counted only once.
maxBracket	30	Maximum number of brackets (solid complementary) allowed in void cell definition
cellRange	[]	Range of cell to be converted (only one range is allowed, e.g [100,220]). Default all solids are converted.
debug	False	Write step files of original and decomposed solids, for each solid in the STEP file.

3.2.3. Tolerances

The tolerances used during the GEOUNED translation process can be modified by the user. There are two type of tolerance distance and angle. Distance tolerance is used to compare distances between two points or the radius of a cylinder or sphere. Angle tolerance is used when checking whether axes are parallel or comparing semi-angle of two cones.

The keyword used for the different tolerances with their default values are defined in Table 5. If this section is not defined or some keywords are not present the default values are used. If *relativeTolerance* is False (default value) distance units are mm and angle units radians.

Table 5. Keyword for tolerances setting.

Keyword	**Default Value**	**Description**
relativeTolerance	False	define the values as relative (True) or absolute in mm or rad.
relativePrecision	1e-6	relative precision in comparison between two numbers
generalDistance	1e-4	distance between objects
generalAngle	1e-4	angle between axis
planeDistance	1e-4	distance between parallel planes. Planes are assumed equal if distance between planes < 1e-4 mm
planeAngle	1e-4	angle between the normal of planes. Planes are parallel if angle < 1e-4
cylinderDistance	1e-4	distance between axis. Difference in radii size
cylinderAngle	1e-4	angle between axis
sphereDistance	1e-4	distance between centers. Difference in radii size
coneDistance	1e-4	distance between apex
coneAngle	1e-4	angle between semiangles/axis
torusDistance	1e-4	distance between Major/Minor radii/center
torusAngle	1e-4	angle between axis

3.2.4. Format

The numeric format of the surface parameters written in the MCNP file can be adjusted for each kind of surface and parameter. The keywords and default values used for each parameter are listed in Table 6. If one variable is not defined the default value is used.

Table 6. keywords of numeric formats for surface parameters.

Keyword	Default Value	Description
P_abc	14.7e	A, B, C coefficients of MCNP general plane definition
P_d	14.7e	D coefficient of MCNP general plane definition
P_xyz	14.7e	Coefficients for PX/PY/PZ
S_r	14.7e	Radii SO/SX/SY/SZ/S
S_xyz	14.7e	X,Y,Z parameters in MCNP sphere definition
C_r	12f	Cylinder Radius
C_xyz	12f	X,Y,Z parameters in MCNP cylinder definition
K_xyz	13.6e	X,Y,Z parameters in MCNP cone definition
K_tan2	12f	Tangent in MCNP cone definition
T_r	14.7e	Torus Radius
T_xyz	14.7e	Torus X,Y,Z in MCNP definition
GQ_1to6	18.15f	1^{st} to 6^{th} parameters in GQ MCNP definition
GQ_7to9	18.15f	7^{th} to 9^{th} parameters in GQ MCNP definition
GQ_10	18.15f	10^{th} parameter in GQ MCNP definition

3.2.5. Options

Select option used during conversion mode. The keywords and default values used for each parameter are listed in Table 7. If one variable is not defined the default value is used.

Table 7. keywords of conversion options.

Keyword	Default Value	Description
forceCylinder	False	Use cylinder (instead of cones) as ancillary surface where unclosed torus surfaces are involved in the solid definition.
newSplitPlane	True	New method to consider plane as cutting surface during the decomposition process. Former method split first planes perpendicular to X,Y,Z axis and then the other planes involved in the solid definition. New method group all parallel planes independently whether their normal are along X,Y,Z axes, and start the decomposition process cutting first with the group having the highest number of parallel planes.
enlargeBox	2	Enlarge box boundary when evaluating the constraint table during the simplification of the void cell definition. (unit is millimeter)
verbose	False	Print output warning during geouned run
nPlaneReverse	0	Threshold value to determine whether cut with parallel planes should be carried out first.
splitTolerance	0	Fuzzy tolerance value used in the FreeCAD function "BOPTools.SplitAPI.slice". This function is used during the solid decomposition process.
quadricPY	False	In openMC python script format, the cones or cylinders no aligned with the X,Y, or Z axis can be defined using the openmc.Cone or open.Cylinder methods but can also be defined with their quadric parameter. If "quadricPY" is

		True then all cones and cylinders will be defined in the openMC python script format under their quadric form.

Note1: During the decomposition phase geouned groups the parallels planes to perform the geometry splitting in order to get the basic solids decomposition. The plane splitting process is carried out by cutting first the solid with groups of parallel planes with lowest elements. This order can be reversed using the nPlaneReverse parameter. If the group of parallel planes with the highest element number is lower than the nReversePlane value then the solid splitting will be carried out by cutting first with group of parallel planes with the highest number of planes. This means that if nReverseplane = 0 the decomposition of the solid by planes will be carried out by cutting first with planes which are not parallel to any other, and then by group of parallel planes starting first by groups with the lowest number of elements. If nReversePlane ≠ 0, if the number of planes in the group of parallel planes with the highest number of elements nh, is lower than nReversePlane, then the decomposition of the solid by planes will be performed first with the group of parallel planes with the highest number of elements, if nh > nReversePlane the process is identical as the nReversePlane = 0 process.

Note2 : The BOPTools.SplitAPI.slice function, is a fundamental function in geouned since it is used for the decomposition process. This function uses a tolerance parameter to determine how the shapes intersection should be considered. It has been observed that in for practically all solids the value of the tolerance parameter doesn't affect the result of the decomposition. But sometimes depending of the value of the tolerance the decomposition can failed (giving a code crashing, or a bad decomposition), and the value of the tolerance to obtain a good decomposition is solid dependent and sometimes this value has to be very low or rather high. So in case of the geouned decomposition give a problem, this value can be adjusted to overcome the issue.

3.3. Reverse conversion process

The reverse conversion (MCNP to CAD) uses a different input file that the one used for the direct conversion, but the structure of the file is the same. The reverse input file has 4 sections described in the following.

The reverse process provides two CAD files: one as a FreeCAD file (FCStd) and another in STEP file. In the FreeCAD file the converted cells are group by universe and material, and labeled with the MCNP label, whereas in the STEP file only solids are written in the same order as in the MCNP file with no further information. A STEP file with the full information can be obtain by exporting the model in FCStd file in STEP format with the FreeCAD software.

3.3.1. Setting.

The setting section is used to define the name of input and output files as well as the region where the model to convert is located. These parameters have no default values and must be set by the user.

During the reverse conversion process, the code slices the space with the surfaces defined in the cell definition in order to reconstruct the solid structure. Since MCNP use infinite surfaces, the user should enter the dimension of a region large enough to contain the full model described in the MCNP input. This region is defined as a parallelepiped, in principle this region can be as large as the user wants to ensure the full model in embedded in the parallelepiped, but it is recommended to adjust is dimension to the size of the model.

Table 8. List of entries of the Setting section.

Keyword	Default Value	Description
mcnpFile	None	Name of the MCNP input file
CADFile	None	Name of the file with the materials information.
outbox	None	Name of the output mcnp file

3.3.2. Levels.

The Levels section is used if the user wants to translate only a specific universe of the model, or only the model up to a given level avoiding nested sub universe. In this last case the solids inside the container cell are not translated, the CAD solid displayed is the container cell itself.

Table 9. List of keywords of the Levels section.

Keyword	Default Value	Description
UStart	0	Universe to be converted to CAD. If UStart = 0 (default) the full model is translated. If UStart ≠ 0, only the universe UStart (and its nested universes) will be translated.
levelMaX	all	Level maximum of nested universe to be translated. If levelMax < highest nested universe level, cells inside the container cell whose level is levelMax will not be translated. This container cell will be the CAD solid written in the CAD file.

3.3.3. Cells and Materials.

Cells and Materials sections are used to filter the cell to be translated. The filter is made by cell label (Cells section) and/or materials value (Materials section). Both sections have the same keywords used in the same way. The filtering is made by include or excluding singles or a range of cells and materials. Default values are not identical for Cells and Materials sections, by default all cells are considered, but cells with no materials (mat = 0) will be excluded of the conversion.

Table 10. List of keywords of the Cells and Materials sections.

Keyword	Default Value	Description
rangeType	all (for cells) exclude (for materials)	Define how to consider the range values. **all** : all the cells with any materials will be translated (range a no effect). **include** : include only cells/materials defined in range. **exclude** : exclude all cells/materials defined in range.
range	None (for cells) 0 (for materials)	List of cells or materials to be included/excluded during the conversion. The cells or material values can be entered as a single value or a range values. Single and range values entries are separated by colon. A range value is defined as is:ie where "is" is the starting value and "ie" the ending value of the range. Both "is" and "ie" values are considering in the range values. Range example: range = 1, 4, 6:10, 20, 45:100

3.3.4. Options

Select option used during conversion mode. The keywords and default values used for each parameter are listed in Table 11. If one variable is not defined the default value is used.

Table 11. keywords of conversion options.

Keyword	**Default Value**	**Description**
splitTolerance	0	Fuzzy tolerance value used in the FreeCAD function “BOPTools.SplitAPI.slice”. This function is used during the solid decomposition process.

4. Void generation capabilities

Two innovative automatic void generation capabilities are included in GEOUNED: enclosures and envelopes. These capabilities are detailed in the next points.

4.1. User defined enclosures

Enclosures are CAD defined solids that are used as base to generate the void for the region covered by them instead of the usual bound box that cover the region of interest. The use of user defined enclosures allows to generate separated void regions inside the same model. This is useful to assign different materials to the void (e.g. different air composition between rooms) and for later user modifications of the geometry. In addition, enclosures can be nested making easy to give a hierarchical structure to the void. In summary, the use of user defined enclosures is a very useful capability to generate clean and structured voids.

Enclosures are defined directly in the CAD model with the component name enclosureX_Y_ where X is the number ID for the enclosure and Y is the enclosure on which the enclosure X depends (in the lowest level Y is 0). For example, enclosure1_0_ is enclosure 1 that has not any enclosure above him and enclosure2_1_ is enclosure number 2 that is under enclosure 1. There are no restrictions on the level of dependence on enclosures. Also, each enclosure should be completely inscribed inside the upper enclosure (part of one enclosure cannot be outside the upper enclosure).

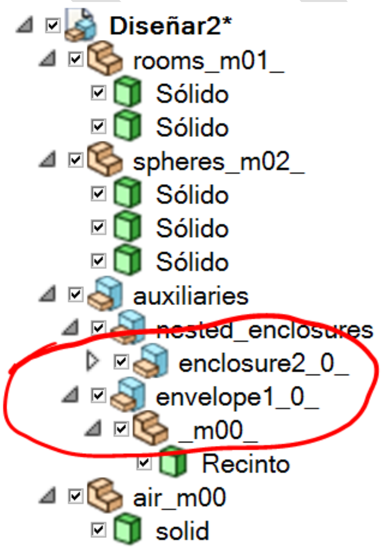

Fig. 1. Example of CAD tree with enclosures and envelopes (highlighted in red).

In section 6 an example of enclosures is provided.

Enclosure solids can be defined anywhere in the CAD model tree. By default, the void cells corresponding to each enclosure are written in the MCNP output file after all the solid definition independently of their position in the CAD model tree. The order of enclosures in the output file is the same order (top to bottom) they appear in the CAD model tree. The last void cells to be written in the output file are the level 0 voids, that means outside of any enclosure.

If the parameters option *sortEnclosure* is True, then the voids of the enclosures will be written in the output file in between the solid cell definition. They will be written in the same order they appear in the CAD model tree independently if there are solid cells defined before or after the enclosure solid.

4.2. Envelopes

In some cases, the extraction of a region from void generation can be useful to generate more optimized MCNP models. For example, if a region of the space is full of other components, an envelope solid covering this region (frequently simpler than the components that it covers) can be defined in the CAD model. This envelope will be used during the void generation to ignore all the components inside itself (only components completely inside the envelope will be avoided in the void generation process). This should produce a simpler void definition. In counterpart, the user should be sure that the region covered by the envelope is clean (i.e. no lost particles) and full.

Envelopes follow the same nomenclature that enclosures with 'envelope' text instead of 'enclosure'. Dependency has no sense in this case as the higher level is the one to be used in the void generation process.

In section 6 an example of envelopes is provided.

5. Material assignment

The material of the components can be assigned directly in the tree CAD by including a text with _mXX_ within the component name. The material assigned to the corresponding cell should be defined in the material definition file whose name is provided by the *matFile* keyword of the *Files* section (see Table 3). This file has the following format:

```
# Materials and density
# FORMAT: ID DENS NAME
1 1.0 Water
```

Where # is used for comments and the rest of the lines specifies the id of the material, the nominal density and the text to be included as a comment. The nominal density value is multiplied by -1 so the criterium is inverted with respect to MCNP one (i.e. g/cm^3 positive and atm/b/cm negative). The nominal density can be changed by the inclusion of a multiplication factor using the following text in the CAD tree _mXX_dXX.XX_. For example, in our case _m01_d2.0_ will produce a cell with material 1 and density 2.0.

Note: The material assigned for components without material definition in the CAD model tree is the one defined in through the voidMat variable of set function (see **Error! Reference source not found.**)

6. Examples

In this section, a simple example is converted to illustrate how GEOUNED works. The same case is converted using different void generation options. The geometry of the case consists of two rooms, one of them with some blocks inside (see Fig. 2). This geometry has been converted using different void generation options.

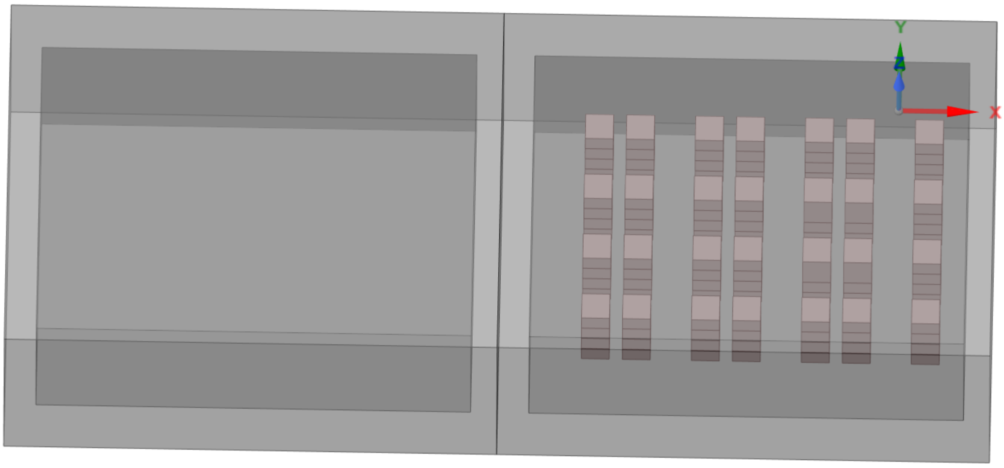

Fig. 2. Geometry of the example

Case 1: Simple automatic void generation

In this case the CAD file is translated using 'standard' automatic void generation. This is without any enclosure defined by the user. In this case the code automatically generates a parallelepiped that covers the geometry as base for the automatic void generation. The result is shown in fig. 4 for the complete geometry. In fig. 5 we can see that several void cells cover both outside and inside room space (e.g. cell 114).

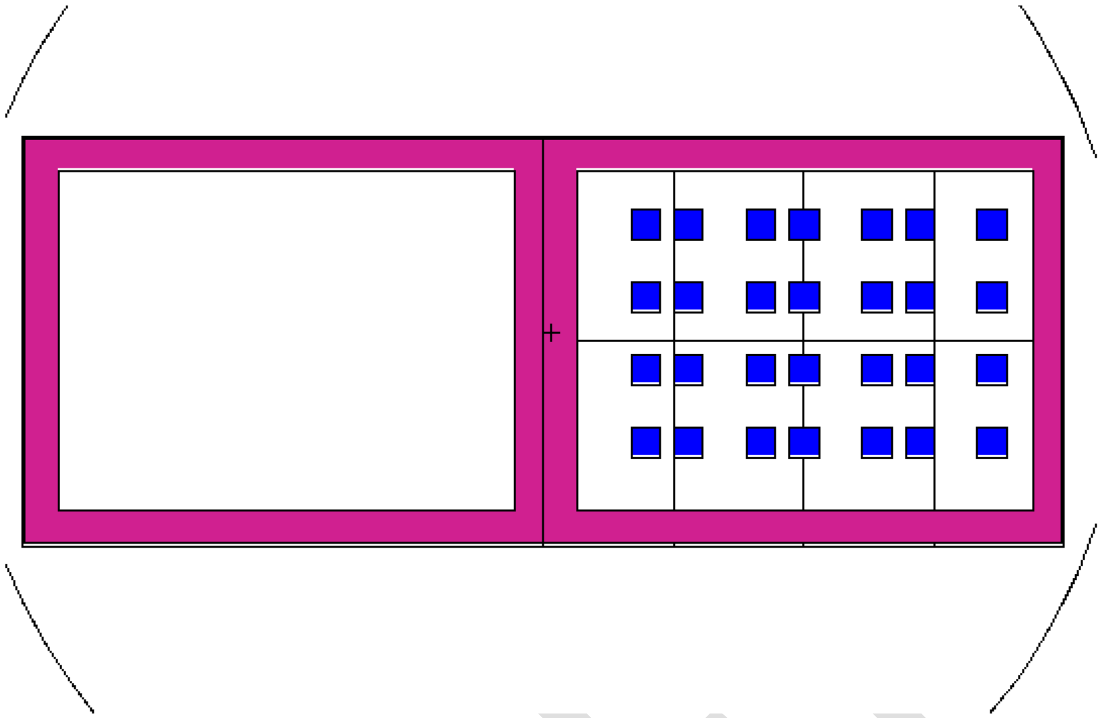

Fig. 3. MCNP model generated with 'standard' automatic void

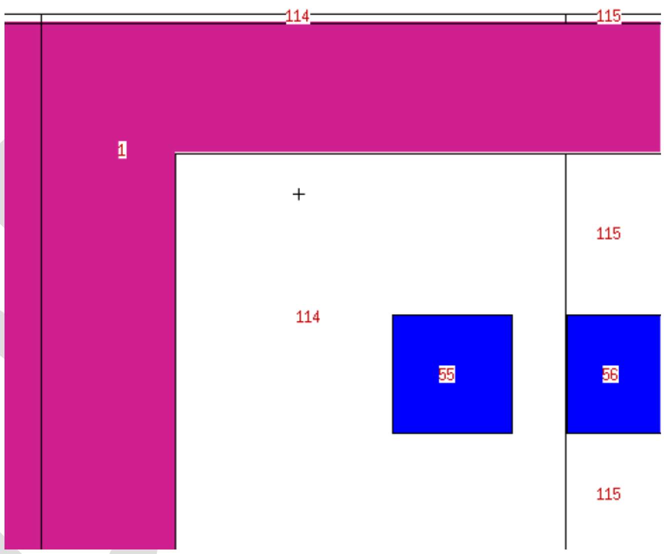

Fig. 4. Detail of the MCNP model close to the wall. Red numbers correspond with cell ids.

Case 2: User defined enclosure

The previous case has been solved using user defined enclosures for the region inside the rooms. The enclosures are simple boxes as it is shown in Fig. 5. The automatic void generation is performed first in the enclosures level and after in the global level avoiding the solids completely inside the enclosures. In Fig. 6 we can see the same detail that in the previous case and we can see that the cell numbers are not shared between the outer and the inner part of the room. In this sense, this capability generates more structured voids and cleaner inputs for the user.

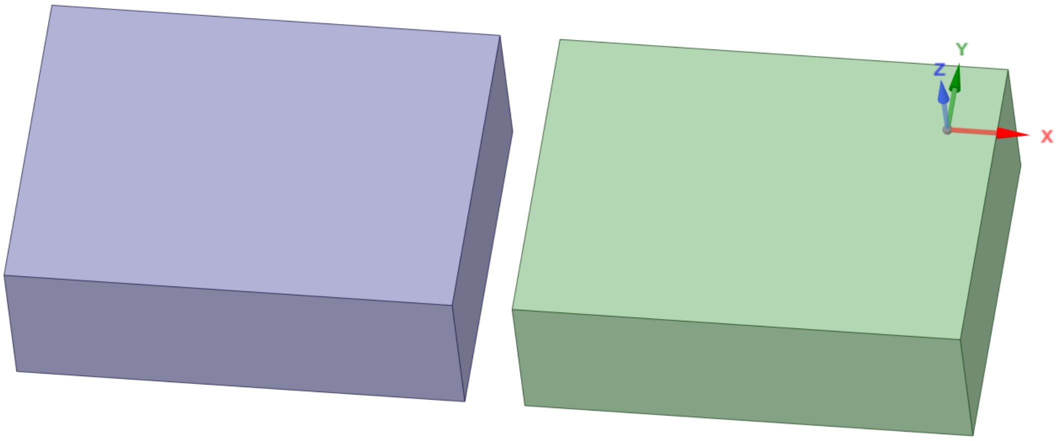

Fig. 5. User defined enclosures.

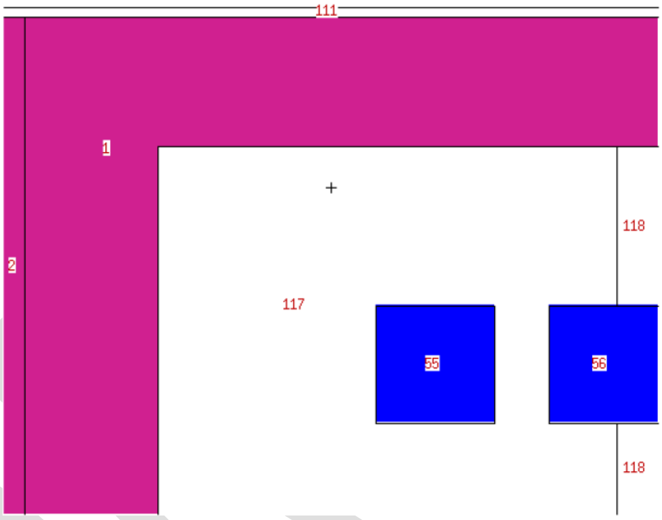

Fig. 6. Detail of the MCNP model close to the wall. Red numbers correspond with cell ids.

Case 3: User defined envelope

Finally, the example is solved using an envelope instead of an enclosure in the room of the blocks. The use of an envelope requires to define a solid for the air of the room because inside the envelope automatic void generation in not active. In this particular case the application of this capability does not provide any advantage but in cases with regions completely full of solids its use can produce lighter and faster inputs. In counterpart, the user should be sure that the geometry inside the envelope do not lost particles during the transport.

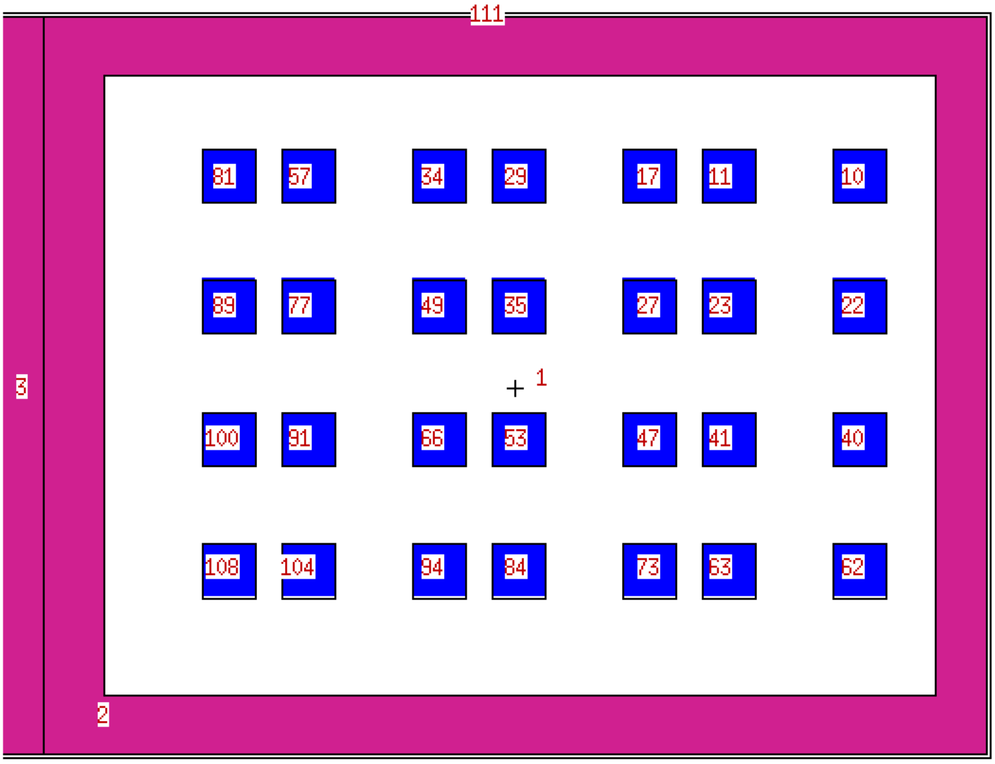

Fig. 7. MCNP model of the room using an envelope. Red numbers correspond with cell ids.

7. Reference the code

The following article is the current reference that should be used in scientific and technical publications:

J. García, J.P. Catalán, J. Sanz, "Development of the automatic void generation module in GEOUNED conversion tool", Fusion Engineering and Design, 168 (2021) 112366, https://doi.org/10.1016/j.fusengdes.2021.112366

8. Contact Information

The contact information for any communication is the following:

Juan Pablo Catalan
Associate Professor/Lecturer in Nuclear Engineering
ETSI Industriales, UNED, Juan del Rosal 12, Madrid, Spain, 28040
Email: jpcatalan@ind.uned.es,
Phone: +34 398 82 09

Patrick Sauvan
Associate Professor/Senior Lecturer in Nuclear Engineering
ETSI Industriales, UNED, Juan del Rosal 12, Madrid, Spain, 28040
Email: psauvan@ind.uned.es
Phone: +34 398 87 31