

CAD_to_OpenMC: from CAD design to particle transport

Erik Bergbäck Knudsen¹ and Lorenzo Chierici²

¹ United Neux, Denmark ² Copenhagen Atomics A/S, Denmark ¶ Corresponding author

DOI: [10.21105/joss.07710](https://doi.org/10.21105/joss.07710)

Software

- [Review](#)
- [Repository](#)
- [Archive](#)

Editor: [Mojtaba Barzegari](#)

Reviewers:

- [@nasserma](#)
- [@jacobmerson](#)
- [@bonh](#)

Submitted: 24 September 2024

Published: 25 June 2025

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC BY 4.0](#)).

Summary

We present CAD_to_OpenMC: a practical tool for allowing geometries designed in CAD systems to be used in Monte Carlo particle transport simulation systems. Written in Python, and easily installable through pip and/or conda channels, CAD_to_openMC does so by importing a geometry in the form of a step-file, a common export format among many CAD systems, identifying objects within that file, and running a triangularization algorithm on all surfaces contained within the geometry. The geometry is written to an HDF5 file in a format defined by the DAGMC library, which is usable by the main particle-transport packages.

Statement of need

Most Monte Carlo neutronics, and general particle transport software packages (e.g., OpenMC, GEANT4, MCNP, Fluka) ([Agostinelli et al., 2003](#); [Böhlen et al., 2014](#); [Ferrari et al., 2005](#); [Kulesza et al., 2022](#); [Romano & Forget, 2013](#)) are ubiquitous in nuclear science. Almost all of these use Constructive Solid Geometry (CSG) to describe the geometry in which computations are to be performed. CSG is computationally efficient but also tricky to work with, in that all object boundaries must be constructed from analytical descriptions of geometrical primaries (cylinders, planes, 2nd-order surfaces, etc.). Most engineering design, however, is performed using CAD tools, which are optimized for ease of design.

These facts form a compelling argument for bridging the gap between these worlds. So far, a number of solutions have been proposed which can mostly be put into two categories.

1. tools that convert CAD models into CSG ([Catalán et al., 2024](#); [Du et al., 2018](#)), and
2. tools that allow ray-tracing on CAD geometries, or close derivatives thereof.

In the second category, one of the most widespread open source solutions is DAGMC ([Wilson et al., 2010](#)), a library with couplings to OpenMC, MCNP, Fluka, and Geant4. DAGMC is proven to be fast and reliable, yet one hurdle remains: DAGMC requires a discretized (meshed) approximation of the CAD geometry. In general, it is not a trivial requirement.

Most CAD software packages are, though often proprietary packages, able to write to the STEP-format, which is backed by an international standard ([Industrial Automation Systems and Integration—Product Data Representation and Exchange – Part 1, 2024](#)). For this reason, the STEP format was chosen as the input format.

The CAD_to_OpenMC tool eases the process of generating a meshed description of a CAD-generated geometry (in the form of a step file) ready for inclusion in transport codes through DAGMC.

Method

CAD_to_OpenMC uses OCCT ([OCCT3D Collaborative Development Portal, n.d.](#)) to interact with CAD geometries, i.e., step-files. Once the geometry has been imported, one of several meshing backends may be called to create a discretized version of the geometry. In the end, the generated geometry is exported into the DAGMC-expected format.

OCCT uses a hierarchical boundary representation model (BREP). In a simplified picture, objects consist of a set of bounded surfaces (faces), formed by a set of boundaries (edges), which themselves consist of line segments connecting points or simply mathematical descriptions of curves. More specifically, the BREP-description used in OCCT also includes notions of shells (the boundary of a volume), loops (a circular set of edges), etc. The BREP concept is fundamentally different from CSG, where objects are formed by boolean operations on halfspaces. For example, the OCCT BREP-description allows several more types of operations for forming a geometry, such as extrusions and rotations.

Thus, a need for conversion always exists, regardless of whether one discretizes the geometry or not.

Triangularization / Surface Meshing

When a curved geometry is to be discretized, the result is an approximation. This leads to potential problems with overlaps between regions if

- a. objects are close to each other, such as for a cylindrical can with thin walls
- b. objects share a surface such as the case for a liquid inside a can.
- c. when objects have surfaces that touch each other.

The first case puts a constraint on the absolute tolerance of the discretization, i.e., triangles have to be small enough not to cause crossing surfaces.

In terms of case b, each surface must only be discretized once. CAD_to_OpenMC handles this by filling a hash table for surfaces upon processing, enabling reuse without reprocessing.

In the last case (c), imprinting has to be performed. This is the process where the boundary curves of one surface are projected onto another, splitting it into two or more sub surfaces.

Meshing backends

CAD_to_OpenMC is made to be flexible in terms of the algorithms, backends, used to generate triangularized surfaces. The present release supports a range of backends: {'gms', 'stl', 'stl2', 'db'}. In most cases stl2 performs well, but sometimes fails to produce a watertight mesh. In such cases, we recommend using the db backend if available. Further, the gms backend consistently produces a practical model, but is memory hungry.

Material tags

CAD_to_OpenMC can use the CAD part names as material tags. The default is to use the first part of the part name as a material tag. This may be changed by supplying a dictionary as the tags-argument. Here, the keys are regular expressions, and the values are the material tags to use. Unmatched parts are tagged as vacuum (the default) but may be tagged by extracted part names. The below example shows how to tag all parts with the name wall in them with concrete and all parts ending with bellows with steel.

```
tag_dict = {".*bellows" : "steel", ".*wall.*" : "concrete"}
```

Implicit complement

CAD_to_OpenMC can add an implicit complement to the output file. That is, the material tag applied to any part of the geometry *not* claimed by any CAD part. This is done by simply assigning the name of the material as a string (with a suffix of `_comp\`) to the implicit complement attribute of the base Assembly object. That string gets picked up by the DAGMC system and is used for the unclaimed volume.

Results

As a test system we have chosen a tabletop reactor (GODIVA IV) model from the ICSBEP-benchmark project ([International Handbook of Evaluated Criticality Safety Benchmark Experiments, 2022/2023](#)) known as case HEU-MET-FAST-086.

GODIVA IV

This reactor consists of a cylindrical core ([Figure 1](#)) held by three sets of clamps set. Additionally, the core has three vertical holes into which control and burst rods may be inserted from below. The rods themselves are similar in composition to the fuel elements.

The benchmark includes five experimental cases, which differ in terms of control- and burst-rod positions. Three geometries are described:

1. a detailed model which includes as-built things (curved clamp etc.),
2. simplified model, where all surfaces are along the principal axis, all corners 90 deg. etc., and
3. cylindrically symmetric model.

The benchmark reports only experimental results for the two first, but contains MCNP geometries for all three in ([International Handbook of Evaluated Criticality Safety Benchmark Experiments, 2022/2023](#)). The 3rd model is created in order to use some legacy analysis tools which are purely 1D/2D.

Corresponding to [Figure 1](#), [Figure 2](#) shows the discretized version of the two reactor models used for our further analysis.

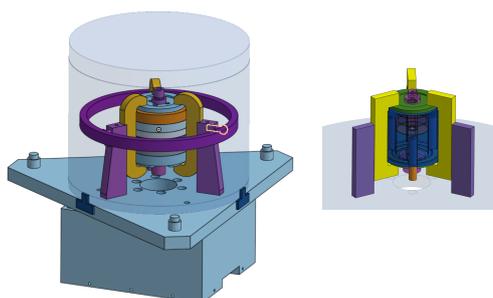


Figure 1: CAD-drawings of the Godiva IV reactor, detailed (left) and simplified (right) versions. Note the rectangular clamps and supports in the simplified version. Visible are also the set of control and burst rods.

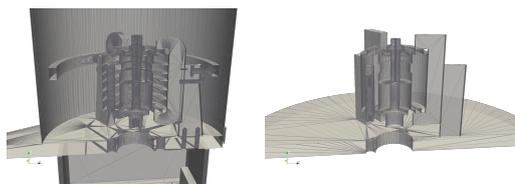


Figure 2: Discretized Godiva IV-models, detailed (left) and simplified (right) versions.

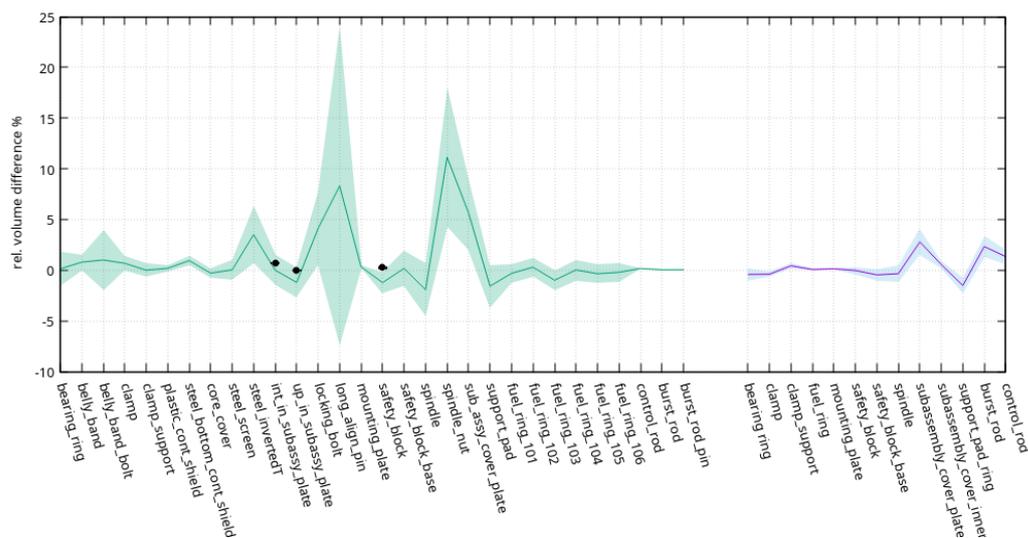


Figure 3: Part-by-part comparison between volume calculations using stochastic volume estimation in OpenMC and volumes reported by CAD-software for the detailed (green) and simplified (magenta) benchmark models. The intervals are the computational error margins, almost exclusively stemming from the estimated error in the stochastic volume computation. The additional black circles denote the relative difference between the extracted benchmark CSG-model (as computed by stochastic volume runs in OpenMC) and the CAD-model.

Figure 3 shows volume differences between discretized and exact CAD models for the various objects in the model.

Differences in volume often influence the neutronics of a reactor more than boundary shifts. Hence, this is a useful measure for performance. The errors found (Fig. 3) are dominated by the error in the stochastic volume estimator, not the volume error itself. This is evidenced by the very small error in burst- and control-rod volume for the detailed model, which were run with smaller tolerances. Additionally, we have used the benchmark CSG-model to verify a few volumes in the detailed model. A CSG model generally yields shorter runtimes, which allows tighter tolerances while remaining practical. The CSG-benchmark deviates slightly from the CAD-model constructed from drawings, suggesting an underlying discrepancy internally in the benchmark.

The considered cases have different settings for the control- and burst-rods (see Tables 1 and 2).

Table 1: Control-rod (CR) and burst-rod (BR) positions for the five cases of the Godiva-IV benchmark/simplified model from HEU-MET-FAST-086 (Goda et al., 2021; *International Handbook of Evaluated Criticality Safety Benchmark Experiments, 2022/2023*). Measures in inches withdrawn from the fully inserted position. The two rightmost columns contain criticality numbers for the device. MC denotes Monte Carlo, whereas Lit. denotes numbers from the benchmark.

case	CR 1 top	CR 2 top	BR top	k_{eff} CAD	k_{eff} CSG
1	-4.001	-0.449	0.0	0.98026	0.98187
2	-1.998	-1.666	0.0	0.98101	0.98185
3	-0.493	-3.794	0.0	0.98124	0.98297
4	-0.469	-0.447	-2.970	0.98745	0.98359
5	-0.319	-0.656	0.0	0.98706	0.98844

Table 2: Control-rod (CR) and burst-rod (BR) positions for the five cases of the detailed Godiva-IV model from HEU-MET-FAST-08 (Goda et al., 2021; *International Handbook of Evaluated Criticality Safety Benchmark Experiments, 2022/2023*). Measures in inches withdrawn from the fully inserted position. The two rightmost columns contain criticality numbers for the device. MC denotes Monte Carlo, whereas Lit. denotes numbers from the benchmark report.

case	CR 1 top	CR 2 top	BR top	k_{eff} CAD	k_{eff} CSG	k_{eff} Lit.
1	-4.001	-0.449	0.0	0.97905	0.98303	0.9880
2	-1.998	-1.666	0.0	0.98390	0.98275	0.9880
3	-0.493	-3.794	0.0	0.97885	0.98330	0.9887
4	-0.469	-0.447	-2.970	0.98352	0.98426	0.9897
5	-0.319	-0.656	0.0	0.98390	0.98969	0.9945

Two minor adjustments (≈ 1 mm) had to be made to the stated measurements; locking bolts and intermediate sub-assembly plate, for the model to fit. Both edits were ~ 1 mm and did not affect the results. We assume the errors are misprints in the drawings.

It is clear from Tables 1 and 2 that the agreement between model and experiment is not perfect, yet the difference between CAD-based and CSG-based models is small (generally on the order of 5×10^{-3}). We take this as proof that the geometry generation works as intended.

Discussion and Conclusion

We submit that the tool presented is a convenient tool for making CAD geometries available for Monte Carlo particle transport. By utilizing the DAGMC-layer, the resulting geometries are not restricted to OpenMC, but in fact may be used also in MCNP, Fluka, etc. Experience has shown that a particularly useful feature is to extract tags from CAD-defined parts and interpret them as material tags for transport. This enables a consistent material naming scheme throughout the entire modeling procedure.

Finally, other active projects exist targeting the problem. In the interest of efficiency and resource management, there are active efforts aiming unification, which may bear fruit in coming releases.

References

Agostinelli, S., Allison, J., Amako, K., Apostolakis, J., Araujo, H., Arce, P., Asai, M., Axen, D., Banerjee, S., Barrand, G., Behner, F., Bellagamba, L., Boudreau, J., Broglia, L., Brunengo, A., Burkhardt, H., Chauvie, S., Chuma, J., Chytráček, R., ... Zschesche, D.

- (2003). Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3), 250–303. [https://doi.org/10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8)
- Böhlen, T. T., Cerutti, F., Chin, M. P. W., Fassò, A., Ferrari, A., Ortega, P. G., Mairani, A., Sala, P. R., Smirnov, G., & Vlachoudis, V. (2014). The FLUKA code: Developments and challenges for high energy and medical applications. *Nuclear Data Sheets*, 120, 211–214. <https://doi.org/10.1016/j.nds.2014.07.049>
- Catalán, J. P., Sauvan, P., García, J., Alguacil, J., Ogando, F., & Sanz, J. (2024). GEOUNED: A new conversion tool from CAD to Monte Carlo geometry. *Nuclear Engineering and Technology*, 56(6), 2404–2411. <https://doi.org/10.1016/j.net.2024.01.052>
- Du, T., Inala, J. P., Pu, Y., Spielberg, A., Schulz, A., Rus, D., Solar-Lezama, A., & Matusik, W. (2018). InverseCSG: Automatic conversion of 3D models to CSG trees. *ACM Transactions on Graphics*, 37(6), 1–16. <https://doi.org/10.1145/3272127.3275006>
- Ferrari, A., Sala, P. R., Fasso, A., & Ranft, J. (2005). *FLUKA: A multi-particle transport code (program version 2005)*. <https://doi.org/10.2172/877507>
- Goda, J., Bravo, C., Cutler, T., Grove, T., Hayes, D., Hutchinson, J., McKenzie, G., McSpaden, A., Myers, W., Sanchez, R., & and, J. W. (2021). A new era of nuclear criticality experiments: The first 10 years of Godiva IV operations at NCERC. *Nuclear Science and Engineering*, 195(sup1), S55–S79. <https://doi.org/10.1080/00295639.2021.1947103>
- Industrial automation systems and integration—product data representation and exchange—part 1: Overview and fundamental principles (standard no. ISO 10303-1:2024)*. (2024). <https://www.iso.org/standard/83105.html>
- International handbook of evaluated criticality safety benchmark experiments* (No. 7645). (2022/2023). OECD Nuclear Energy Agency.
- Kulesza, J. A., Adams, T. R., Armstrong, J. C., Bolding, S. R., Brown, F. B., Bull, J. S., Burke, T. P., Clark, A. R., Forster, R. A., III, Giron, J. F., Grieve, T. S., Josey, C. J., Martz, R. L., McKinney, G. W., Pearson, E. J., Rising, M. E., Solomon, C. J., Jr., Swaminarayan, S., Trahan, T. J., ... Kulesza, J. A. (2022). *MCNP code version 6.3.0 theory & user manual* (LA-UR-22-30006, Rev. 1). Los Alamos National Laboratory. <https://doi.org/10.2172/1889957>
- OCCT3D collaborative development portal*. (n.d.). Retrieved September 18, 2024, from <https://dev.opencascade.org/>
- Romano, P. K., & Forget, B. (2013). The OpenMC Monte Carlo particle transport code. *Annals of Nuclear Energy*, 51, 274–281. <https://doi.org/10.1016/j.anucene.2012.06.040>
- Wilson, P. P. H., Tautges, T. J., Kraftcheck, J. A., Smith, B. M., & Henderson, D. L. (2010). Acceleration techniques for the direct use of CAD-based geometry in fusion neutronics analysis. *Fusion Engineering and Design*, 85(10), 1759–1765. <https://doi.org/10.1016/j.fusengdes.2010.05.030>