Andrew Davis, Janet Barzilla, Alfredo Ferrari, Kerry T Lee, Vasillis Vlachoudis and Paul P H Wilson

FluDAG: A CAD based tool for high energy physics

FluDAG: A CAD based tool for high energy physics

Andrew Davis,[a, b] Janet Barzilla,[e] Alfredo Ferrari,[d] Kerry T Lee,[c]
Vasillis Vlachoudis[d] and Paul P H Wilson[b]

[a]High Performance Computing Group, United Kingdom Atomic Energy Authority, Culham, Abingdon, Oxfordshire, OX14 3DB, UK

[b]Department of Engineering Physics, 1500 Engineering Drive, Madison, Wisconsin, 53706, USA

[c]Johnson Space Center, 2101 E NASA Pkwy, Houston, Texas, 77058, USA

[d]European Organization for Nuclear Research (CERN), Geneva, Switzerland

[e]Leidos, 9801 Westheimer Rd Unit 900, Houston, TX 77042

FluDAG: A CAD based tool for high energy physics

Andrew Davis[a,b,*], Janet Barzilla[e], Alfredo Ferrari[d], Kerry T Lee[c], Vasillis Vlachoudis[d], Paul P H Wilson[b]

[a]*High Performance Computing Group, United Kingdom Atomic Energy Authority, Culham, Abingdon, Oxfordshire, OX14 3DB, UK*
[b]*Department of Engineering Physics, 1500 Engineering Drive, Madison, Wisconsin, 53706, USA*
[c]*Johnson Space Center, 2101 E NASA Pkwy, Houston, Texas, 77058, USA*
[d]*European Organization for Nuclear Research (CERN), Geneva, Switzerland*
[e]*Leidos, 9801 Westheimer Rd Unit 900, Houston, TX 77042*

Abstract

As part of the expansion of the Direct Accelerated Geometry Monte Carlo (DAGMC) toolkit to support other Monte Carlo codes, FluDAG (FLUKA integrated with DAGMC) was developed. There has been increasing demand from the high energy physics community regarding Computer Aided Design (CAD) geometry support in Monte Carlo codes. In this paper, the development and validation of FluDAG is discussed and its application to a number of high energy physics experiments is demonstrated, along with its validity relative to native FLUKA calculations.

Keywords: CAD, Monte Carlo, FLUKA, DAGMC, Radiation Physics
2010 MSC: 00-01, 99-00

1. Introduction

Several modern nuclear, particle physics or other high energy applications such as JET, ITER, LHC or others impose significant challenges on the geometric representation of models used in the Monte Carlo (MC) radiation transport process. Specifically, there usually exists a detailed engineering model typically created with manufacturing, assembly, or other engineering analysis in mind. In

---

*Andrew Davis
Email address: `andrew.davis@ukaea.uk` (Andrew Davis)

tandem there are also usually several physics analysis models created to ensure optimal experimental performance, for example signal-to-noise ratio in the case of particle detectors or nuclear heating performance in the case of tokamaks. It
10 is the case that usually the physics geometric model lags behind the engineering design by several months, for several reasons (1) CAD model preparation is typically done by several CAD analysts working in parallel to make modifications, (2) it takes significant human effort to clean and simplify (defeature) these complex CAD models, (3) conversion of even simplified CAD is error prone, tedious
15 and slow. There is also increasing demand from nuclear analysts to use CAD in their analysis usually driven by the availability of these engineering CAD models and the desire to include as many relevant details in their analysis model so as to better reflect reality.

20 There have been several approaches taken to solve the problem of the use of CAD models in MC simulations, these can be broken down into two branches of solution; (1) translation approaches and (2) direct use of CAD. A non exhaustive list of translation approaches includes McCAD [1], SuperMC [2] (formerly MCAM [3]), CATIA-GDML [4], FastRAD [5], and Geomit [6]. There are also a
25 number of approaches that take the direct use approach such as BRL-EGS Nova [7], BRL-MCNP [8], Direct Accelerated Geometry Monte Carlo (DAGMC) [9] and OiNC [10].

The translation approach takes the CAD geometry as produced and attempts to
30 decompose the geometry into primitive Combinatorial Solid Geometry objects, such as planes, boxes, spheres, cones and surfaces up to the 2nd order with the exception of tori. There are however issues with this approach when the CAD geometry contains high order spline surfaces which have no analytic representation within the 2nd order limited description. For geometries that can be
35 completely decomposed into 2nd order surfaces, finding the intersection of rays with the geometry is mathematically well defined and significant investment has been made in finding computationally robust and efficient methods.

2

The direct use approach allows for the CAD model to be used without re-moving features from the geometry. Within this approach are two methods. One method requires high order root finding in order to determine intersection of rays with CAD geometry. This method allows the CAD model to be used without any simplification of the geometry as all operations are performed in the native CAD kernel. However, the consequence of high order root finding is paid in computation time, and such approaches are typically significantly slower than their native implementations (typically by at least an order of magnitude). A second method, and the one employed by DAGMC, is to use high resolution tessellated (faceted) representations of the CAD surfaces. Doing this eliminates the need to find higher order roots during navigation and a number of accelerations are used to offset the high number of facets (triangles) required for high resolution representations. The penalty here is memory usage, but this is easily overcome with the large amounts of memory available on the workstations of today.

DAGMC has been widely used within the fusion neutronics community on several analyses for the international experiment ITER [11, 12, 13] and on a number of other fusion relevant calculations [14, 15]. One of the major benefits of DAGMC is the ability to use it as a common geometric base for multi-physics analysis, for example [16]. DAGMC has now been integrated with the FLUKA radiation transport package [17, 18] to form FluDAG, a fully featured version of FLUKA that is able to transport particles in CAD geometries.

2. DAGMC

DAGMC is an open source collection of C++ based software libraries that allows for efficient ray tracing of CAD based geometries. It is built on the back of a mesh based library MOAB [19] which is used to store the DAGMC geometry. As discussed previously, DAGMC geometry is composed entirely of

3

a large number of triangular facets. It would be computationally prohibitive to perform navigation with a large number of triangles without relying upon algorithms that eliminate the need to test each facet. DAGMC has employed acceleration techniques to make it computationally competitive with native geometry implementations. These acceleration techniques are as follows:

1. Imprinting is an operation performed within the CGM CAD engine where surfaces that are coincident have their curves imprinted into the opposing surface, merging then takes the two topologically equivalent surfaces and unifies them into one definition. This provides an acceleration because determining the next volume entered upon crossing a surface is always a $O(1)$ operation.

2. By representing the surfaces of the CAD model as a collection of triangles, each ray query is reduced to a planar ray intersection, which individually is quick, and is ultimately handled using the Plücker [20] ray triangle intersection test.

3. The DAGMC geometry is composed of collections of volumes, with surfaces stored as children sets of volumes, where the facets (triangles) are members of the surface set. For each surface in the problem a tree of oriented bounding boxes is built, these trees are then used to determine which triangle a specific ray hits in $O(log(n))$ time.

Using the accelerations above, DAGMC is able to be competitive with a native Combinatorial Solid Geometry (CSG) MC calculation, typically within a factor of 2-3 for a moderately complex geometry. It should be noted that this difference in timing decreases with increasing geometric complexity. It is argued that despite the slower calculation time, an overall reduction in analysis time is possible due to the speed at which a CAD model can be made ready for analysis. This has also been anecdotally realised in several analyses.

2.1. DAGMC Workflow

The DAGMC workflow has evolved over the course of its development, however the common factor is the generation of a CAD based solid model. The

4

most common route for DAGMC based analysis is to begin by importing the solid model into SpaceClaim where most CAD repairs, cleaning, and defeaturing takes place. The model is then exported to ACIS [21] (*.sat) format (although the workflow does support STEP [22] as well as several other CAD formats) and imported into Cubit [23] or Trelis [24]. A DAGMC-based analysis allows a number of attributes of the geometry to be defined within the geometry file. These characteristics generally relate to the physical properties of the volume, for example its material definition or boundary conditions. The overall workflow is shown in Figure 1.
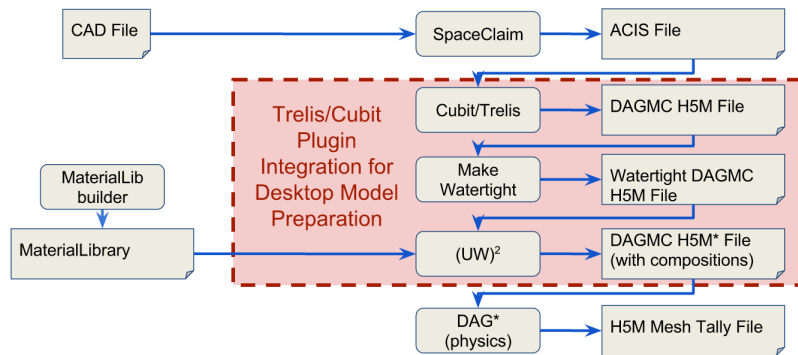

Figure 1: The DAGMC model preparation workflow

The Trelis/Cubit tool is intrinsic in the preparation of DAGMC geometries, as it is used to mark up and produce the faceted geometry. All of the infrastructure required to produce the DAGMC geometries are distributed as plugin objects for Trelis/Cubit. There are a number of standalone command line tools that are run sequentially on a model following faceting. We run make_watertight [25] to seal models to ensure no topological weaknesses exist.

2.2. University of Wisconsin Unified Workflow

The DAGMC workflow is supported in several MC codes and as the number of supported codes grew it became difficult to translate already existant MC code metadata into the newly supported code form, for example MCNP material descriptions to Geant4 materials. Thus the University of Wisconsin Unified

Workflow (UW)² was developed. The workflow encodes material descriptions in code agnostic form storing densities, nuclide mass or atom fractions, comments and other extensible metatdata. These material objects are translated either

₁₂₀ apriori or at runtime depending upon the use case. Particular care is given to nuclides concentration given their particular impact to the neutron based simulations. It is also possible, but in a much more limited fashion to define tallies which can be store in code agnostic form, and again translated to a native format when possible. At the current time, support for tallies/scoring is much

₁₂₅ less developed than material handling. This lack of support is due in part to the complexity of possible scoring mechanisms and geometric fidelity of the specific MC code, e.g. MCNP allows surface based scoring methods whereas FLUKA & Geant4 offer only boundary crossing scores.

2.3. FluDAG Development

₁₃₀ Development of FluDAG was initiated by the National Aeronautics and Space Administration (NASA) having interest in a CAD based workflow for space radiation analysis. Given the complex nature of space craft and the ready availability of CAD models there is an obvious need for the ability to use CAD models in MC calculations. Integration of DAGMC with FLUKA requires that

₁₃₅ the FluDAG wrapper code honours the FluGG [26] (FLUKA with Geant Geometry) API. A schematic of DAGMC interaction with FLUKA is shown in Figure 2.3. The FluDAG layer is used to store all additional state that is required for full functioning of the DAGMC layer, including the last the direction and position of the particle during the last call and if the last step was directly onto a

₁₄₀ boundary beteween regions. This state was introduced to handle some specific issues that were encountered during the development of FluDAG that was not expected based on experience of the integration of DAGMC with MCNP5 [27] or Tripoli4 [28].

₁₄₅ One of the robustness features of DAGMC is the RayHistory object. It is an object used to store the facets crossed in the current ray direction, and is up to
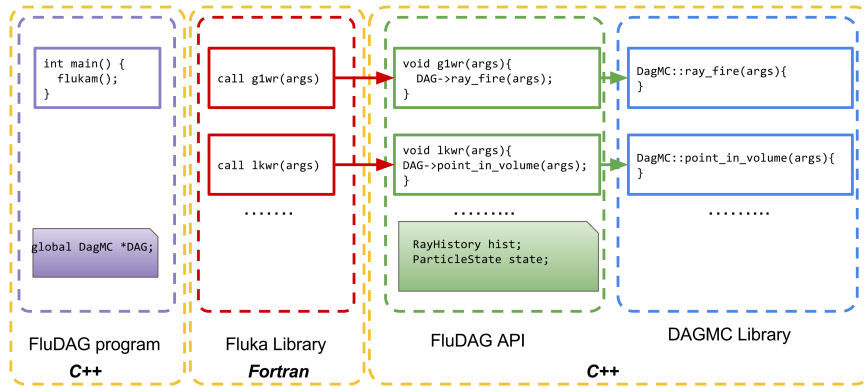
6

Figure 2: Schematic of the program flow and function linkage between FLUKA & DAGMC

the code author to reset the state of RayHistory appropriately. When DAGMC functions are called and are passed a RayHistory object the routines ignore hits from facets already in the object. Thus the RayHistory should be reset at any direction change, since logically we should be allowed to hit the same facet again.

Electron (along with some other charged particles) transport is different from neutral particle transport due to the underlying physics. One of the differences of electron transport (when using condensed history approaches) is that electrons are allowed to change direction at boundary crossings, in DAGMC terms it means that one must be careful about resetting the expected state of particles and checking for this condition. Neutral particles cannot change direction on a boundary (unless undergoing optical reflection) and therefore do not encounter this problem. FLUKA also takes a 'sensing' step to determine how far away potential boundaries are, which must also be handled by the FluDAG layer and be understood to not be a true geometric step and thus no ray state should be retained.

DAGMC had never been previously used for codes that supported magnetic field tracking. FLUKA has specific geometry routines for point inclusion and tracking when magnetic field tracking is on. What this again means from the

7

DAGMC perspective is that the DAGMC wrappers must check to ensure that the RayHistory state is not perturbed incorrectly, since underlying these calls are DAGMC rayfire calls. In conventional CSG geometry repeated calls to the ray-object intersection call will always give the same result, as will DAGMC. However, in the case of DAGMC the RayHistory object can be used to store previous surface hits, potentially being used to preclude the same facet being intersected more than once. Consider the case of an energetic electron traversing a region of uniform magnetic field. As the electron traverses the orbit around its guiding centre, computationally this is performed by breaking the orbit into a number of angular sub-steps, at the beginning of each sub step a rayfire is preformed and the distance to the boundary queried, it is clear to imagine that if there are very many sub-steps in a single orbit, or if there are a few facets in the surfaces of a given volume, then it is very likely the same facet will be intersected, hence it is critical that the history object is reset if the particle does not take the full physics limited step.

By utilising the FluGG API there a number of benefits, (1) consistency of external navigation tools is maintained across different codebases, (2) integration of DAGMC into FLUKA bears significant similarity to other packages reducing maintenance burden, (3) Benchmarking of the API has already been performed by the FLUKA team to have confidence that the API is sufficiently verbose

3. Validation

There are a number of key tests to stress the geometry responses from FluDAG. In terms of high energy physics problems, electrons and neutrons represent the end state of majority of histories, and represents the majority of the CPU time. Indeed for most problems energy is mostly transferred to electrons. A number of the tests performed originated from the development of FluGG and these are reproduced since these tests proved pathological for the case of

8

electron transport. The key comparison in all of the following tests is that when comparing native FLUKA calculations to the exact FluDAG geometry the results are statistically equivalent, i.e. agreement to within $\pm 3\sigma$.

3.1. Aluminium and Gold

²⁰⁰ The AlAuAl (aluminium-gold-aluminium) benchmark originated from the development of FluGG. This benchmark is specifically designed to stress electron transport in very thin layers, especially for the previously mentioned electron behaviour at geometric boundaries. The thin layers with very different densities and atomic numbers strongly affect the underlying electron transport, ²⁰⁵ and thereby photon production by bremßtrahlung, pair production etc. The geometry of the setup is shown in Figure 3.
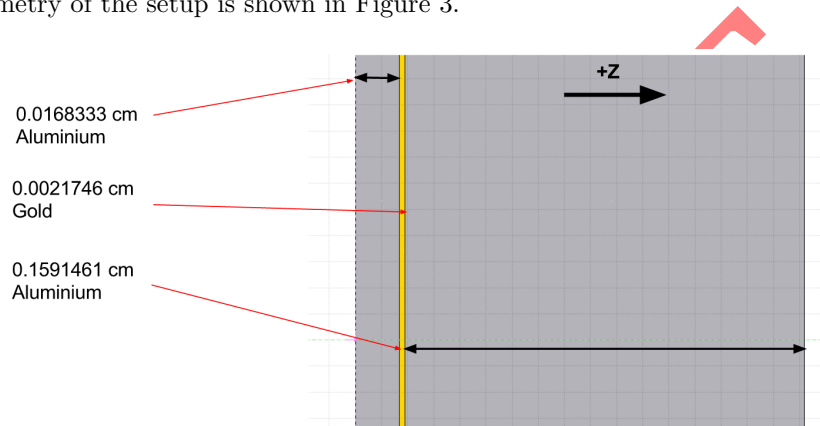
Figure 3: The geometry of the setup for the AlAuAl benchmark (FLUKA geometry shown)

The geometry is composed of 3 material regions, a thin aluminium layer, a thin gold layer, followed by a aluminium layer. The source is a 1 MeV pencil electron beam pointed in the positive z direction, with particles starting at 0,0, ²¹⁰ -10.0 cm. The CAD model for the FluDAG was created by exporting the native FLUKA geometry to MCNP format, then using mcnp2cad [29] the CAD model was created. There were 20 batches of calculation, where each batch contained 5.0×10^5 primary histories.

In each material zone the track length estimate of electron flux is determined,
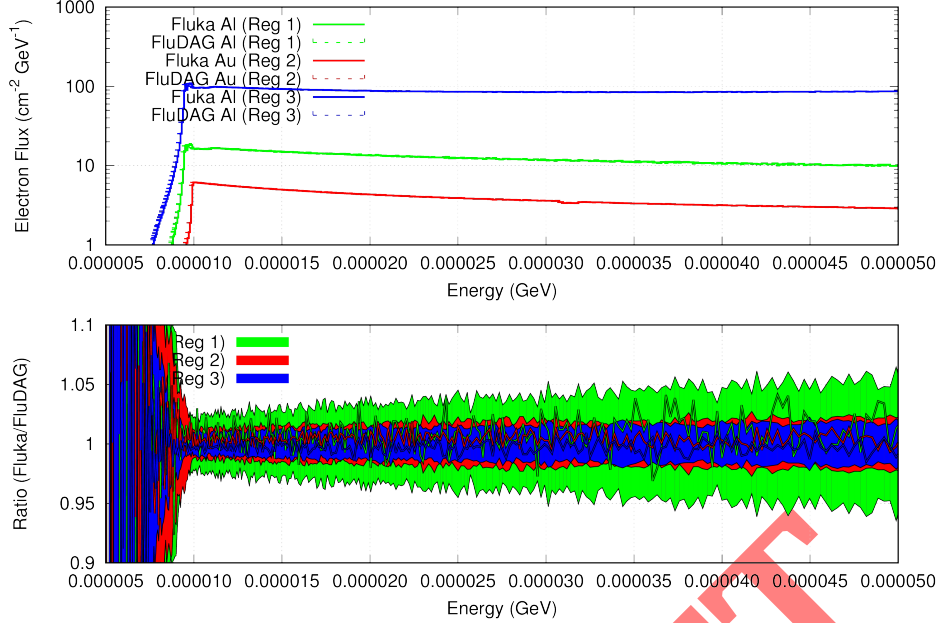
Figure 4: Electron spectra found in the AlAuAl benchmark in the lower energy regime, 5 keV to 50 keV showing the data (upper) and the ratio of FluDAG/FLUKA (lower), the coloured regions denote the $\pm 3\sigma$ zone

as shown in Figure 4. The non-physical undulating artifacts present in Figure 5 are due to the choice of electron physics used, the key take away point is that the features are exactly reproduced in each code across the whole energy range at each stage. This benchmark provided guidance and showed shortcomings in the original development of FluDAG regarding particle direction changes on boundaries, highlighting that despite the simplicity of this benchmark it exposed aberrant electron behaviour, which was subsequently remedied to produce the results shown here.

3.2. Magnetic Fields, Spheres and Cylinders

During the development of FluGG a specific test case gave particularly pathological behaviour for electrons crossing boundaries between cells. The test is subsequently known as "MagnSph" and the geometry is shown in Figure 6. This test is particularly pathological for electron transport due to some
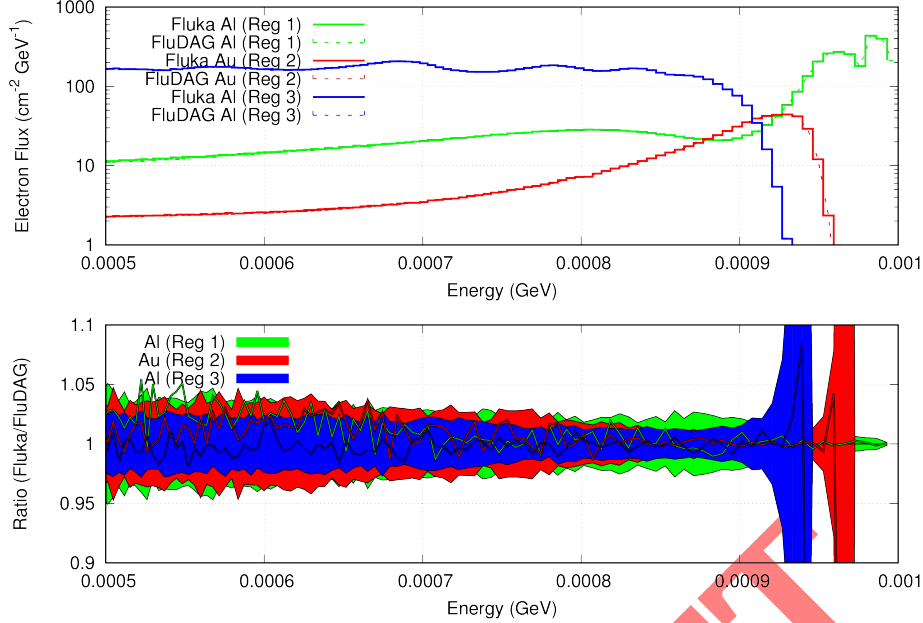
10

Figure 5: Electron spectra in the AlAuAl benchmark in the higher energy regime (500 keV to 1 MeV) showing the data (upper) and the ratio of FluDAG/FLUKA (lower), the coloured regions denote the $\pm 3\sigma$ zone

of the peculiarities of electron transport as discussed previously regarding direction changes on boundaries. The native geometry is composed of cylinders and spheres which are numerically coincident at their extrema, in this case the cylinders have a radius of 0.5 cm and are offset such that adjacent cylinders centres are 1.0 cm from one another.

This benchmark highlights one of the major differences in the geometric capabilities between traditional combinatorial solid geometry and CAD geometry. It is not possible for CAD engines to represent the exact mathematical relationships to the precision possible in CSG. The CSG engines within MC codes have been tuned for accuracy using double precision arithmetic specifically to avoid problems with roundoff. CAD engines are tolerant to a given fixed precision which is many orders of magnitude larger than a double precision number, for

11

example the ACIS CAD kernel treats entities closer than 1.0×10^{-6} cm as being coincident. It is not possible to resolve numerically touching entities where the numerical precision required is higher than that of the CAD engine, for example ACIS can only distinguish vertices as being distinct entities when they are greater than 1.0e-6 cm apart. In this instance we found that the problem as originally defined resulted in several imprint and merge issues. Reducing the radii of the cylinders from 0.5 to 0.49999 and the radii of the spheres from 0.3 to 0.29999 resulted in no issues regarding imprinting or merging. However, changing the radii to 0.49999**9** and 0.29999**9** respectively had the same issues as the unmodified model. Thus, the final model used for the benchmark calculations were has radii of cylinders and spheres of 0.49999 and 0.29999 respectively. This of course means that the geometries are not quite identical, but for the purposes of this benchmark are treated as being equivalent.
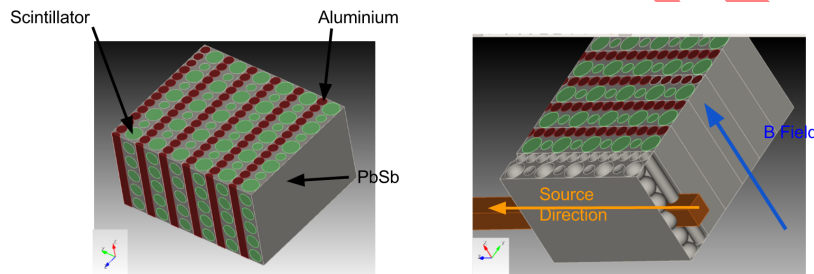

Figure 6: Geometry of the MagNSphe benchmark showing the incident positron source

The particle source is a square cross sectioned beam of side 1 cm, with the primary source particle being positrons at 1 GeV, starting at 2,-4,-1 cm directed into the positive z direction. There is a uniform magnetic field of 60 T directed along the x direction.

The results of electron transport are shown in Figure 7, no statistically significant artifacts can be seen between the FLUKA and FluDAG simulations, thus showing that the FluDAG magnetic tracking routines are performing as expected. The resultant secondary photon distribution is shown in Figure 8, the photon distribution should somewhat follow the electron distribution since
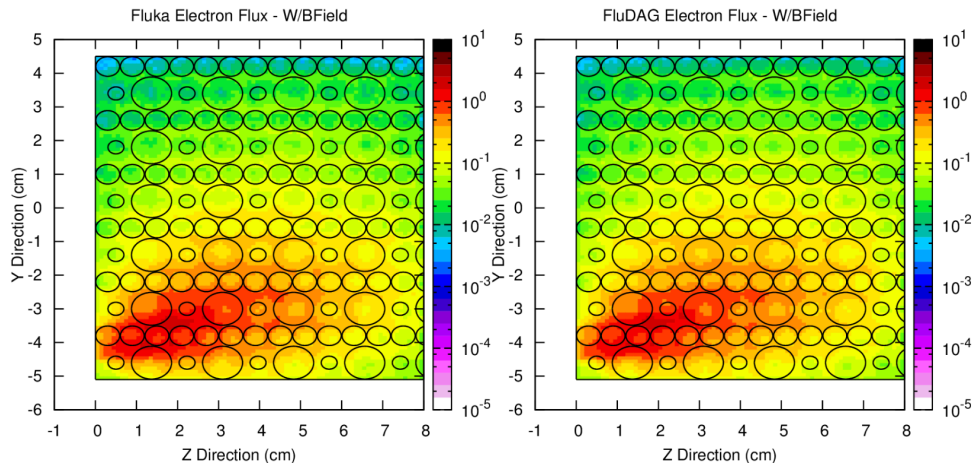
12

Figure 7: Electron flux determined in the FLUKA (left) & FluDAG (right) in the case with magnetic fields
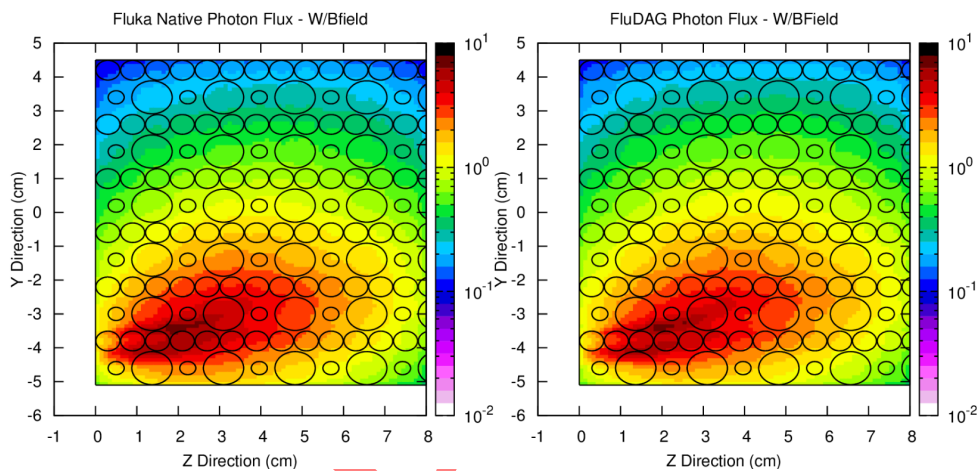

Figure 8: Photon flux determined in the FLUKA (left) & FluDAG (right) in the case with magnetic fields

the primary particles are positrons. The energy deposition, shown in Figure 9, shows excellent agreement between the FLUKA and FluDAG implementations.
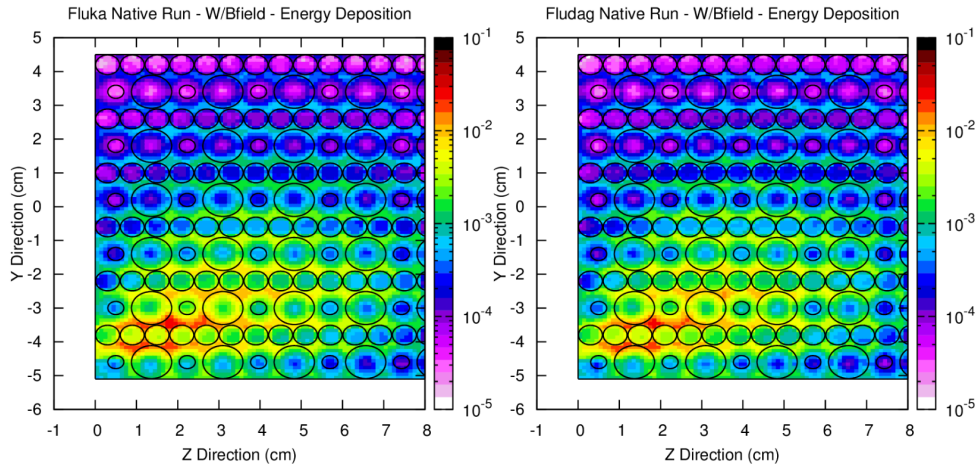
Figure 9: Energy deposition determined in the FLUKA (left) & FluDAG (right) in the case with magnetic fields

3.3. ATIC

The Advanced Thin Ionisation Calorimeter (ATIC) is a high altitude balloon based cosmic ray detector. The detector is composed of several layers of Bismuth Germanate (BiGeO) scintillator, interspersed between them are silicon matrices in order to determine the charge of particles traversing the layer. During the development of the detector several models were created, including native FLUKA models and Geant4 models. The FLUKA model was used as the basis of conversion for this analysis, using the "export to MCNP" feature in Flair, then using the mcnp2cad tool to produce an ACIS file.

This benchmark was used as evidence that the development work done for FluDAG was correct, it was decided by NASA that this geometry, shown in Figure 10, would be used to validate the FluDAG code.

The results were shown to give excellent agreement between FLUKA & FluDAG. For all the particles examined there was agreement within the statistical errors with no outliers beyond 3σ. This along with the previous benchmarks showed that there is a direct agreement between FLUKA & FluDAG
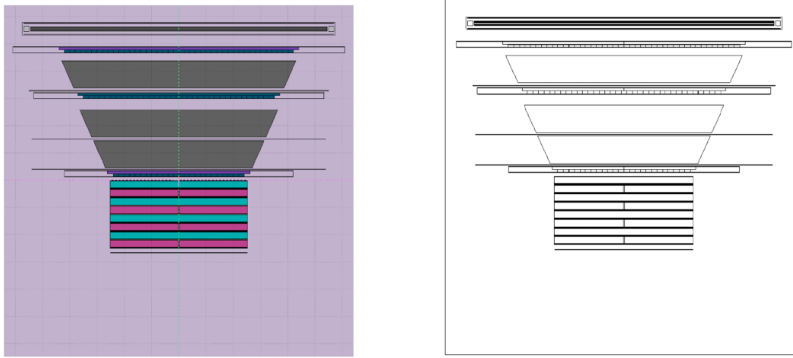
14

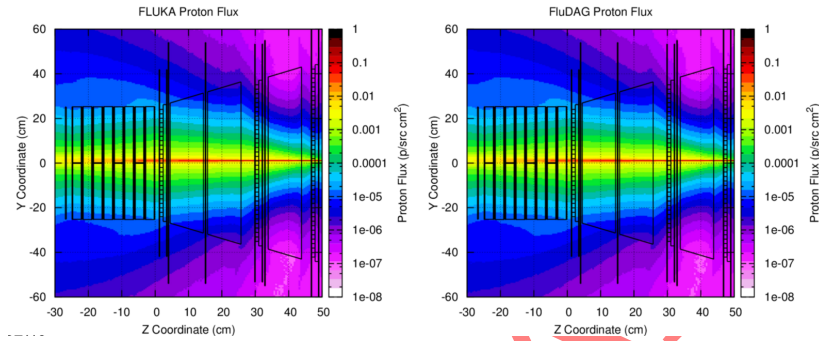Figure 10: Geometry of the ATIC calculations native FLUKA (left) and FluDAG (right)

Figure 11: Proton flux profile determine in the native FLUKA geometry (left) and the FluDAG geometry (right)

when equivalent geometries are used.

3.4. nTOF

The n_TOF (Neutron Time Of Flight) facility at CERN is a pulsed neu-
²⁸⁵ tron source designed to study neutron-nucleus interactions for a wide range of
neutron energies up to several GeV. The study of neutron interactions is of crit-
ical importance to several fields including nuclear technology, nuclear fusion,
and accelerator driven systems. Bunches of $\sim 10^{12}$ protons impinge on a largely
lead target. Combinations of direct (p,xn) and (n,xn) interactions ultimately
²⁹⁰ produce some 300 neutrons per incident proton. These neutrons are moder-
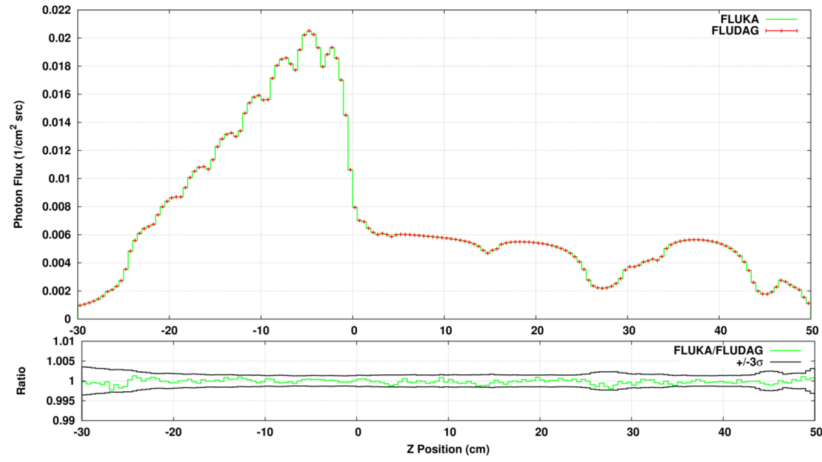ated using water at the rear of the target and then directed along a number of

15

Figure 12: Proton flux profiles, FLUKA as lines, FluDAG as points and the ratio of the profiles (FluDAG/FLUKA)
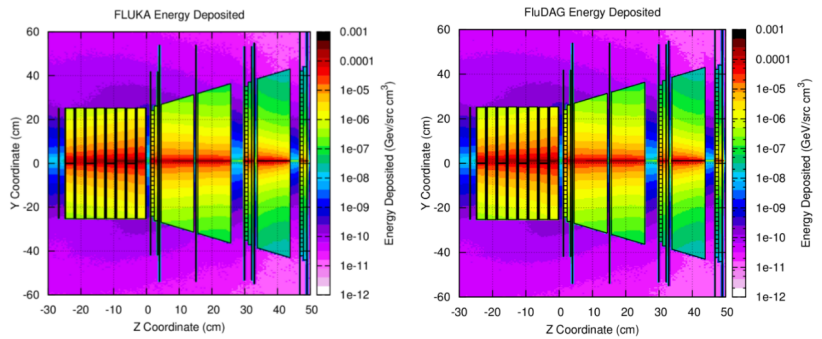

Figure 13: Energy deposition profile determine in the native FLUKA geometry (left) and the FluDAG geometry (right)

beamlines. These neutrons are then used for a number of fundamental neutron science purposes such as cross section measurement and nuclear astrophysics.

In total 3 groups of calculations were run; FluDAG - a full detailed CAD based analysis with the CAD model supplied by CERN, FLUKA - the native FLUKA analysis, and FluDAG-Simple - the literal translation of the native FLUKA geometry into CAD, the geometries used are shown in Figure 15. Similarly to the other cases, the direct translation of the native version of the geom-
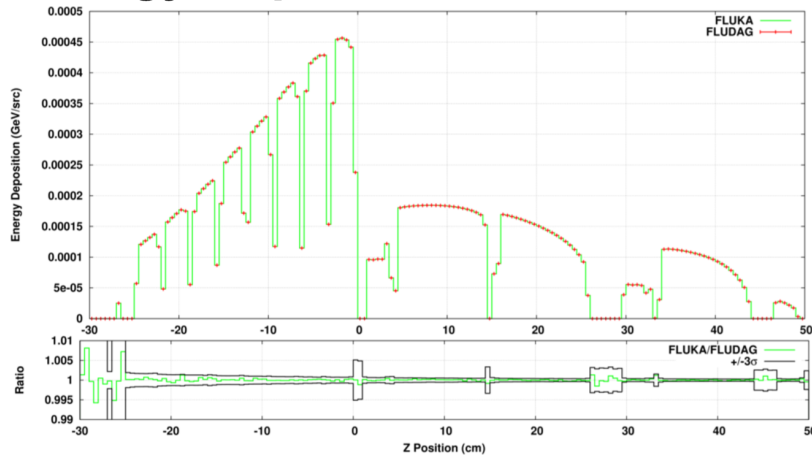
295

16

Figure 14: Energy deposition profiles, FLUKA as lines, FluDAG as points and the ratio of the profiles (FluDAG/FLUKA)
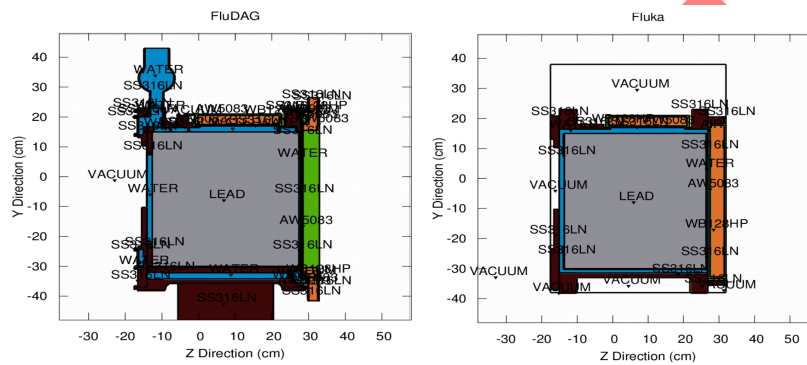

Figure 15: The geometry as used in the nTOF calculation, FluDAG (left) and FLUKA (right)

etry was produced by using the feature in Flair "export to MCNP", then using
300 the mcnp2cad tool to convert the MCNP geometry to CAD. All the calculations run for this section were run on at the Centre for High Throughput Computing (CHTC) system, for 5000 primary histories per calculation, 5 calculations per batch, and 1000 groups of simulation, for a total of 25 million primaries.

305 A detailed comparison of the neutron spectrum leaving the rear of the device was performed. In the three cases a one way angular neutron flux was per-

17

formed considering the neutrons crossing within 15° of the surface normal and was binned into 1000 logarithmically spaced bins between 1.0×10^{-14} GeV and 20 GeV. The purpose in this instance is to a) perform a like-for-like comparison between the native and direct translated geometries and b) determine the effect on downstream neutron production when a geometry representative of the true geometry is used.

The results for the comparison between FLUKA and FluDAG-Simplified are shown in Figure 16. The comparison between the native FLUKA and detailed FluDAG comparison is shown in Figure 17. Good agreement is seen at the higher energies above 400 MeV, however at lower energies there is a statistically significant difference. This difference is due to the heterogeneity of the CAD geometry in comparison to the native geometry, this is confirmed by the statistically equivalanet results of the FluDAG-Simplified geometry results shown in Figure 16.
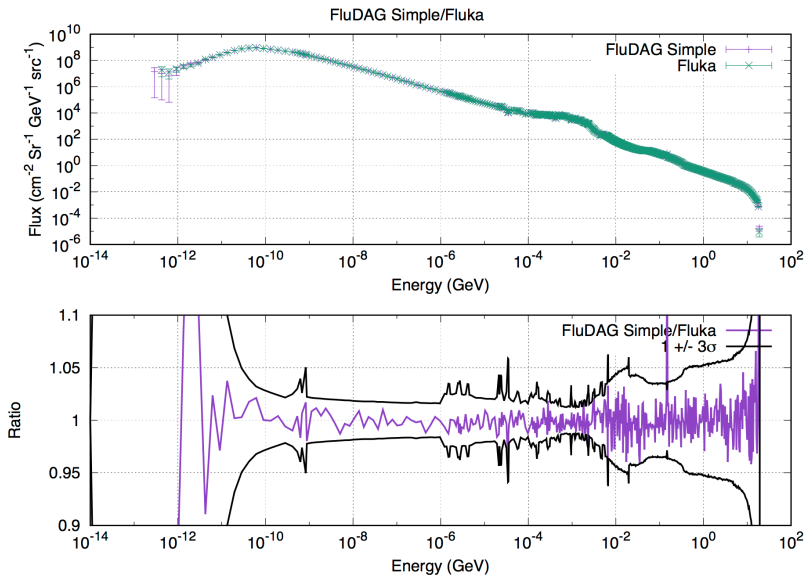

Figure 16: Neutron spectra leaving the rear vacuum surface within 15° of normal for FLUKA & FluDAG simple
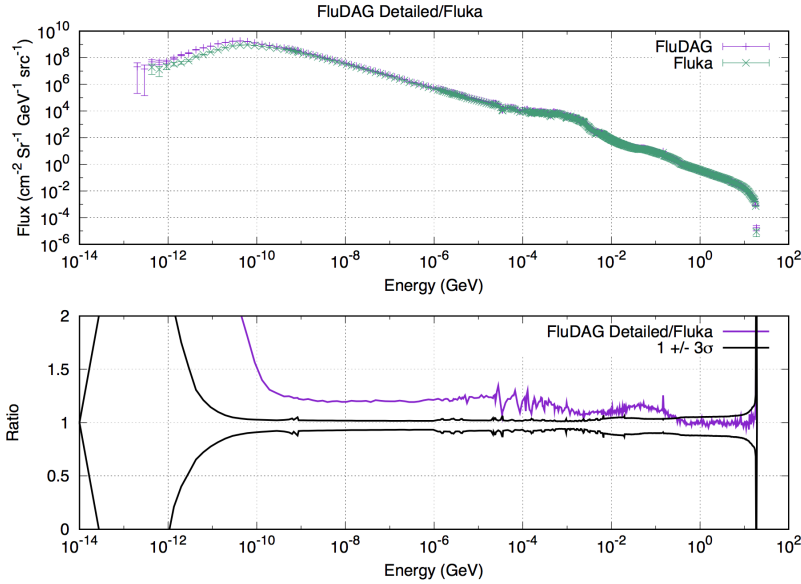
18

Figure 17: Neutron spectra leaving the rear vacuum surface within 15° of normal for FLUKA & FluDAG (detailed CAD model)

The comparisons shown for the nTOF geometry reconfirms that for directly equivalent geometries produce statistically equivalent results for FLUKA and FluDAG. However, this example also demonstrates the importance of including sufficient details in your model for the analysis at hand.

4. Conclusions

In this paper the validation performed on FluDAG was discussed. The development of FluDAG was discussed, highlighting important lessons that were learned during the process, specifically regarding electron transport and charged particle transport in magnetic fields. Two benchmarks were reported on as part of FluDAG development, one covering very thin metallic layers (with thicknesses of less than 20 μm) named 'AlAuAl', the other was more geometrically complex with magnetic field enabled, and both were shown to give agreement to within statistical error with native FLUKA geometry and navigation. The complex

19

geometry of the Advanced Thin Ionisation Calorimeter was used to compare FLUKA and FluDAG and were shown to give excellent agreement throughout the model to within statistical error. As an example of high energy physics use, the neutron time of flight (nTOF) facility was analysed and when using identical geometries were shown to give excellent agreement, however when using a detailed CAD geometry there were some differences beyond statistics, due to the heterogeneity. FluDAG and DAGMC have been shown to be a viable method for performing analyses allowing the use of complex CAD geometries and eliminating the need to approximate as-built geometries to confirm to MC second order primitive limitations.

References

[1] A. Serikov, U. Fischer, D. Grosse, P. Spaeh, H. Tsige-Tamirat, Use of mc-cad for the generation of mcnp models in fusion neutronics, in: In the proceedings of the International Conference on Mathematics, Computational Methods and Reactor Physics (M&C 2009), 2009.

[2] J. Song, G. Sun, Z. Chen, H. Zheng, L. Hu, Benchmarking of cad-based supermc with {ITER} benchmark model, Fusion Engineering and Design 89 (11) (2014) 2499 – 2503. `doi:https://doi.org/10.1016/j.fusengdes.2014.05.003`.
URL `http://www.sciencedirect.com/science/article/pii/S0920379614003731`

[3] Y. Li, L. Lu, A. Ding, H. Hu, Q. Zeng, S. Zheng, Y. Wu, Benchmarking of mcam 4.0 with the iter 3d model, Fusion Engineering and Design 82 (15) (2007) 2861 – 2866, proceedings of the 24th Symposium on Fusion Technology. `doi:http://dx.doi.org/10.1016/j.fusengdes.2007.02.022`.
URL `http://www.sciencedirect.com/science/article/pii/S0920379607000762`

[4] S. Belogurov, Y. Berchun, A. Chernogorov, P. Malzacher, E. Ovcharenko, V. Schetinin, Development and application of catia-gdml geometry builder, Journal of Physics: Conference Series 513 (2) (2014) 022003.
URL `http://stacks.iop.org/1742-6596/513/i=2/a=022003`

[5] P. Pourrouquet, J. C. Thomas, P. F. Peyrard, R. Ecoffet, G. Rolland, Fastrad 3.2: Radiation shielding tool with a new monte carlo module, in: 2011 IEEE Radiation Effects Data Workshop, 2011, pp. 1–5. `doi:10.1109/REDW.2010.6062530`.

[6] S. Sato, H. Nashif, F. Masuda, H. Morota, H. Iida, C. Konno, Progress of conversion system from cad data to mcnp geometry data in japan, Fusion Engineering and Design 85 (7) (2010) 1546 – 1550, proceedings

21

of the Ninth International Symposium on Fusion Nuclear Technology. doi:http://dx.doi.org/10.1016/j.fusengdes.2010.04.038.

URL http://www.sciencedirect.com/science/article/pii/S0920379610001791

[7] J. Tabary, A. Gliere, Coupling photon monte carlo simulation and cad software, application to x-ray nondestructive evaluation, in: Advanced Monte Carlo for Radiation Physics, Particle Transport Simulation and Applications, in Proceedings of the Monte Carlo 2000 Conference, 2000, pp. 487–492.

[8] O. R. N. Laboratory, U. S. D. of Energy. Office of Scientific, T. Information, MCNP-BRL: A Linkage Between MCNP and CAD Geometry, Oak Ridge National Laboratory., 2009.

URL https://books.google.com/books?id=-tVInQAACAAJ

[9] P. P. Wilson, T. J. Tautges, J. A. Kraftcheck, B. M. Smith, D. L. Henderson, Acceleration techniques for the direct use of cad-based geometry in fusion neutronics analysis, Fusion Engineering and Design 85 (1012) (2010) 1759 – 1765, proceedings of the Ninth International Symposium on Fusion Nuclear Technology. doi:http://doi.org/10.1016/j.fusengdes.2010.05.030.
URL http://www.sciencedirect.com/science/article/pii/S0920379610002425

[10] K. Searson, F. Fleurot, A. Cooper, P. Cowan, Oinc: A comprehensive cad import and tracking system for monte carlo radiation transport calculations, Nuclear Technology 175 (1) (2017) 259–263. arXiv:http://www.tandfonline.com/doi/pdf/10.13182/NT11-A12297, doi:10.13182/NT11-A12297.
URL http://www.tandfonline.com/doi/abs/10.13182/NT11-A12297

[11] T. D. Bohm, M. E. Sawan, S. T. Jackson, P. P. Wilson, Detailed nuclear analysis of {ITER} {ELM} coils, Fusion Engi-

22

neering and Design 87 (5-6) (2012) 657 – 661, tenth International Symposium on Fusion Nuclear Technology (ISFNT-10). doi:http://doi.org/10.1016/j.fusengdes.2012.01.031.
URL http://www.sciencedirect.com/science/article/pii/S0920379612000427

[12] B. Weinhorst, U. Fischer, L. Lu, Y. Qiu, P. Wilson, Comparative assessment of different approaches for the use of {CAD} geometry in monte carlo transport calculations, Fusion Engineering and Design 9899 (2015) 2094 – 2097, proceedings of the 28th Symposium On Fusion Technology (SOFT-28). doi:http://doi.org/10.1016/j.fusengdes.2015.06.042.
URL http://www.sciencedirect.com/science/article/pii/S0920379615300739

[13] T. Bohm, A. Davis, M. Sawan, E. Marriott, P. Wilson, M. Ulrickson, J. Bullock, Detailed 3-d nuclear analysis of {ITER} outboard blanket modules, Fusion Engineering and Design 96-97 (2015) 222 – 226, proceedings of the 28th Symposium On Fusion Technology (SOFT-28). doi:http://doi.org/10.1016/j.fusengdes.2015.06.004.
URL http://www.sciencedirect.com/science/article/pii/S0920379615300351

[14] C. Kessel, J. Blanchard, A. Davis, L. El-Guebaly, N. Ghoniem, P. Humrickhouse, S. Malang, B. Merrill, N. Morley, G. Neilson, et al., The fusion nuclear science facility, the critical step in the pathway to fusion energy, Fusion Science and Technology 68 (2) (2015) 225–236.

[15] J. Menard, M. Boyer, T. Brown, J. Canik, B. Covelle, C. DAngelo, A. Davis, L. El-Guebaly, S. Gerhardt, S. Kaye, et al., Configuration studies for an st-based fusion nuclear science facility, in: Proceedings of the 25th IAEA Fusion Energy Conference, St. Petersburg, Russia, Vol. 1318, 2014.

[16] E. D. Biondo, A. Davis, P. P. Wilson, Shutdown dose rate analysis with cad

23

430   geometry, cartesian/tetrahedral mesh, and advanced variance reduction, Fusion Engineering and Design 106 (2016) 77–84.

[17] T. Böhlen, F. Cerutti, M. Chin, A. Fassò, A. Ferrari, P. Ortega, A. Mairani, P. Sala, G. Smirnov, V. Vlachoudis, The fluka code: Developments and challenges for high energy and medical applications, Nuclear Data Sheets 120 (2014) 211 – 214. `doi:http://dx.doi.org/10.1016/j.nds.2014.07.049`.
URL `http://www.sciencedirect.com/science/article/pii/S0090375214005018`

[18] FLUKA: A multi particle transport code, Tech. Rep. CERN-2005-10 (2005), INFN/TC 05/11, SLAC-R-773 (2005).

[19] Moab: A mesh oriented database, Tech. Rep. SAND2004-1592, Sandia National Laboratory (2004).

[20] J. Ericsson, Plücker Coordinates.
URL `http://turing.acm.org/pubs/resources/RTNews/html/rtnv10n3.html`

[21] D. Corporation, ACIS v7.0 Format.
URL `http://paulbourke.net/dataformats/sat/sat.pdf`

[22] Iso 10303-21, `https://www.iso.org/standard/63141.html`, (March. 2016).

[23] S. N. Laboratories, The CUBIT Geometry and Mesh Generation Toolkit (2015).

[24] CSIMSOFT (2017). [link].
URL `http://www.csimsoft.com/trelis.jsp`

[25] B. M. Smith, T. J. Tautges, P. P. H. Wilson, Sealing Faceted Surfaces to Achieve Watertight CAD Models, Springer Berlin Heidelberg, Berlin,

24

Heidelberg, 2010, pp. 177–194. `doi:10.1007/978-3-642-15414-0_11`.
URL `http://dx.doi.org/10.1007/978-3-642-15414-0_11`

[26] First calorimeter simulation with the flugg prototype, Tech. Rep. ATL-SOFT-99-004, CERN (December 1999).

[27] M. E. Sawan, P. P. H. Wilson, T. Tautges, L. A. El-Guebaly, D. L. Henderson, T. D. Bohm, E. P. Marriott, B. Kiedrowski, B. Smith, A. Ibrahim, R. Slaybaugh, Application of cad-neutronics coupling to geometrically complex fusion systems, in: 2009 23rd IEEE/NPSS Symposium on Fusion Engineering, 2009, pp. 1–6. `doi:10.1109/FUSION.2009.5226505`.

[28] Tripoli-4 version 8 user guide cea-r-6316, `http://www.oecd-nea.org/tools/abstract/detail/nea-1716/`, (Feb. 2013).

[29] Mcnp2cad, `http://www.github.com/svalinn/mcnp2cad`, (Mar 2016).

25