CCFE
CULHAM CENTRE FOR FUSION ENERGY

CCFE-PR(17)10

Andrew Turner

Investigations into Alternative Radiation Transport Codes for ITER Neutronics Analysis

Investigations into Alternative Radiation Transport Codes for ITER Neutronics Analysis

Andrew Turner

CCFE, Culham Science Centre, Oxfordshire OX14 3DB, United Kingdom

This paper has been submitted to ANS 2017 Annual Meeting Transactions

The following article appeared in ANS 2017 Annual Meeting Transactions.

Investigations into alternative radiation transport codes for ITER neutronics analysis
Turner A

Investigations into Alternative Radiation Transport Codes for ITER Neutronics Analysis

Andrew Turner

UK Atomic Energy Authority (CCFE), Culham Science Centre, Abingdon, Oxon, OX14 3DB, UK.
andrew.turner@ukaea.uk

INTRODUCTION

As radiation transport models for ITER become increasingly complex, the use of traditional constructive solid geometry (CSG) modelling approaches such as MCNP presents significant disadvantages during model creation and integration. Studies were performed to assess the capabilities of alternative mesh-based neutronics workflows offered by the MCNP [1], DAG-MCNP [2] and Serpent-2 [3] codes.

WORKFLOW ISSUES

Model simplification

Model creation for ITER analysis usually entails the creation of an MCNP model of a specific system (e.g. a diagnostic port) which is then integrated into an existing reference ITER geometry for neutronics analysis (e.g. C-model). ITER geometry is complex and detailed, particularly the blanket and port plug components - often consisting of spline geometry unsupported in CSG, hence requiring extensive effort to create suitable approximations. CAD simplification is therefore a bottleneck to creating ITER MCNP models.

Simplified CAD models are converted to MCNP using automatic conversion software (e.g. SuperMC, MCCAD), which have become mature and effective, with features such as automatic splitting and void creation. However, once an MCNP model is created, additional bottlenecks are encountered during integration.

Model integration

After conversion, tiny surface overlaps that often occur due to CAD numerical precision can result in lost particles in a transport simulation. These require manual effort to repair.

The model is then integrated into the ITER neutronics reference model, which due to complexity is built using a universe structure to simplify modification. However, a geometrically clean model inserted into a geometrically clean universe does not necessarily result in geometry free of errors. In MCNP, surfaces that exist in both the fill model and the universe model can result in geometry errors which are difficult to diagnose and repair. This is particularly the case when fill transforms are used.

For both of these reasons, significant effort can be required downstream of conversion to obtain a model that is ready for analysis.

Consequences

Neutronics analysis is only one of many physics and engineering disciplines that feed into the design of ITER systems. The significant timescales needed to simplify CAD and modify the resulting MCNP model may potentially lead to delays in the design process.

MCNP model modifications downstream of conversion (due to geometry errors, coincident boundaries) have to be performed every time the model is converted from CAD. As such, this discourages CAD-based model modification and it is often impractical to explore multiple design iterations by adapting the MCNP model by hand.

Alternative approaches

Mesh-based workflows are offered by MCNP6, DAG-MCNP and Serpent-2. The potential advantages of such techniques are:

- To proceed rapidly from CAD to a transport model with minimal simplification and accurate spatial approximations.
- To avoid manual repair work after conversion.
- To rapidly make changes to the CAD design, re-mesh, and swiftly create additional model variants ready for analysis.

A CAD-based modelling approach will allow leveraging of the efficient CAD manipulation capabilities of software such as SpaceClaim, and enable design optimisation studies to be carried out effectively.

The disadvantages of such methods are on the computational side – speed and memory requirements. A study was performed on a simplified ITER-like model to compare these approaches in terms of the results, computational requirements, capabilities and limitations of the codes.

METHODOLODY

Octamak model

The model created was developed to be readily convert to CSG as well as being meshed, and it was to have ITER-like features and proportions. The model

('Octamak', due to its shape) consists of blanket modules with inter-modular gaps, divertor cassettes, vacuum vessel with port extension, port plug with penetration, port interspace structures, superconducting coils, and a biological shield. The model was developed in SpaceClaim and is shown in Fig. 1.

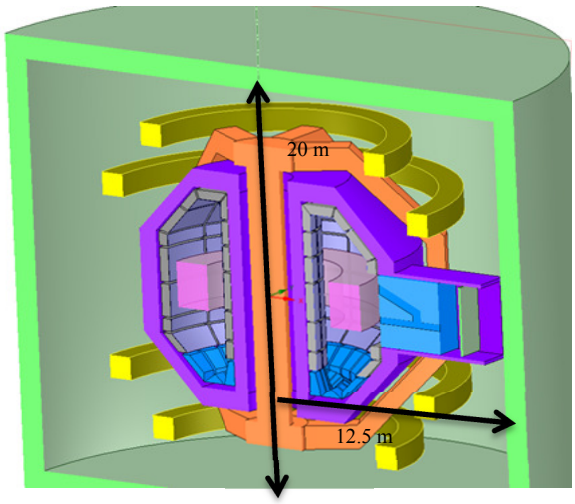

Fig. 1. Octamak model

Codes

The mesh implementations in these codes differ. MCNP6 supports volume meshes (e.g. tetrahedra) whilst DAG-MCNP utilises a surface mesh (triangular facets). Serpent-2 supports both of these, however only the surface mesh was used in this study. To differentiate the mesh types, the term unstructured mesh (UM) refers to a volume mesh and unstructured surface (US) indicates a surface mesh technique.

The following code versions were used: MCNP6.1 (for MCNP-CSG), MCNP6.1.1-beta (MCNP6-UM), MCNP5v1.6 with DAGMC (DAG-MCNP), Serpent v2.1.27 (Serpent-CSG and Serpent-US).

MCNP6-CSG and DAG-MCNP are mature and validated codes, whilst Serpent-2 is in the beta stage of development. The MCNP6 mesh implementation is at an early stage of development, and the beta version of the code was used containing several bug fixes (the upcoming MCNP6.2 release is expected to be reliable). DAGMC is not currently available for MCNP6.

Model conversion

The model was converted to an MCNP-CSG model using SuperMC 2.3.7 [4] with automatic void creation. Serpent-2 has equivalent surface types and surface senses to MCNP, and hence the MCNP-CSG model was converted to an identical Serpent-CSG by simple changes to syntax.

A 14.1 MeV neutron source was defined within the plasma chamber, emitting $1x10^{20}$ n/s. ITER-like materials were used for all components. FENDL-3.0 cross-section data was used in all codes.

In order to create identical mesh geometry for all codes, Attila4MC was used to generate a tetrahedral mesh. From here, Attila4MC was used to output an ABAQUS volume mesh for MCNP-UM and a native 'RTT' format mesh file. The RTT mesh also contains surface mesh data, which was read by the MOAB 'mbconvert' tool and converted to both MOAB 'H5M' format for DAG-MCNP and a series of STL files for Serpent-US. In this way, the same mesh geometry could be compared in the different radiation transport codes. Four meshes of increasing resolution were produced. For brevity, results are presented for the coarsest mesh (mesh #1), with the others used to study speed and memory implications of detailed models. Table I shows the mesh counts of the models compared to CSG.

Table I. Meshed models of Octamak

Geometry	Surfaces/facets	Cells/elements
CSG	541	1795
Mesh 1	$3.14x10^4$	$4.38x10^4$
Mesh 2	$1.82x10^5$	$6.07x10^5$
Mesh 3	$5.91x10^5$	$3.00x10^6$
Mesh 4	$1.27x10^6$	$1.12x10^7$

Tallies/detectors

In order to compare results, tallies were defined as follows: neutron flux and heating in selected blanket cells, neutron flux in the port interspace, neutron heating in superconducting coils. Simulations were run in neutron-only transport mode without any variance reduction (due to the lack of coupled N P transport and weight windows in Serpent-2, both currently under development).

RESULTS

A selection of results are reported below.

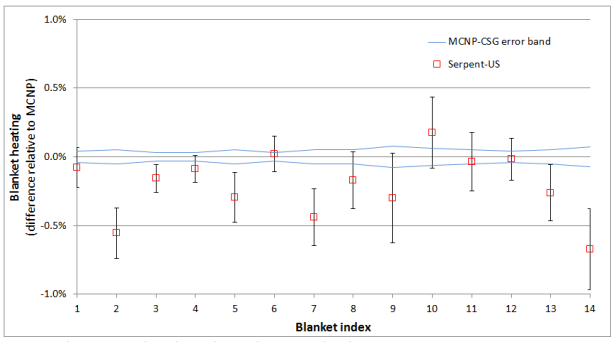

Fig. 2. Blanket heating, relative to MCNP-CSG

Table II. Coil neutron heating results

	Total heating (kW)	Statistical error	Relative to MCNP-CSG
MCNP-CSG	13.49	0.16%	1.000
Serpent-CSG	13.49	0.53%	1.000
Serpent-US	13.56	0.52%	1.005
MCNP-UM	13.52	0.49%	1.002
DAG-MCNP	13.45	0.49%	0.997

Blanket heating results are only shown for Serpent-US relative to MCNP-CSG, as an example (Fig. 2), since no significant differences were observed in any of the modelling approaches. Table II demonstrates excellent agreement for heating of the coils. No significant deviations were noted between any of the methods and the reference result (MCNP-CSG model). All the codes and approaches gave reliable results for this model.

Computational efficiency was compared using the tally figure of merit (FoM = $[err^2 \times time]^{-1}$). The time was that spent performing radiation transport - initialisation time was neglected (which was particularly significant in the case of the MCNP-UM simulations). These are shown in Fig. 3.

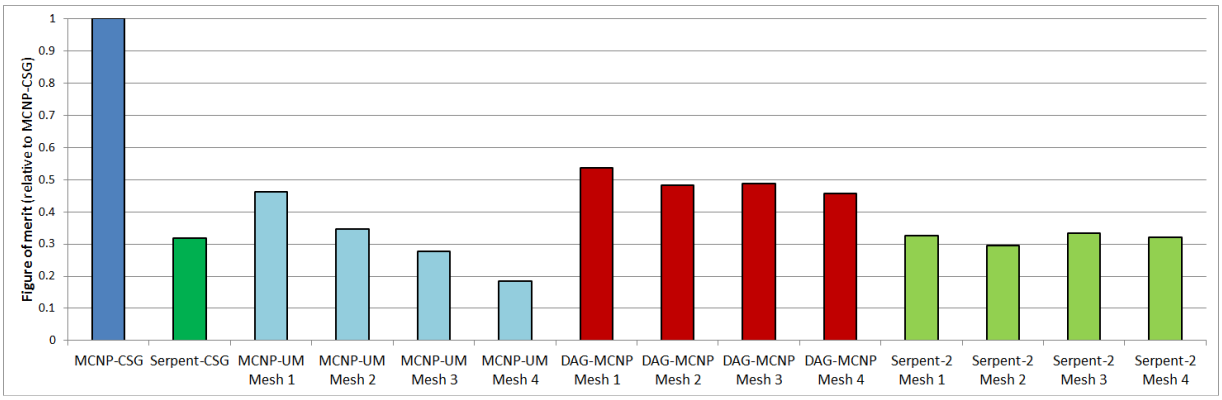

Fig. 3. Figure of merit for blanket heating tally

Table III. Peak memory use

	Memory (GB)				
	CSG	Mesh #1	Mesh #2	Mesh #3	Mesh #4
MCNP	1.3	1.0	4.2	22.2	10.6*
Serpent	11.4	13.2	12.1	12.3	12.5
DAG	-	1.6	1.6	2.1	3.2

**mesh adapted to reduce memory*

Assessing computational efficiency of tallies (FoM) relative to MCNP-CSG, it was found that:

- Serpent-CSG was less efficient than MCNP-CSG with a relative FoM of 0.3.
- DAG-MCNP yielded a relative FoM of 0.5, and for Serpent-US 0.3.
- MCNP-UM relative FoM was in the range 0.2 – 0.5, and was found to decrease significantly with increasing mesh resolution.
- The surface-based mesh methods of DAG-MCNP and Serpent did not show significant slowdown with increasing resolution.

Assessing the peak memory demands of the approaches, the following is noted:

- Memory use of Serpent-2 was significant, 11 GB compared to 1 GB for MCNP-CSG.
- DAG-MCNP memory requirement was 2-3 GB, marginally increased over MCNP-CSG.
- The surface-based methods of DAG-MCNP and Serpent-2 did not show significant memory increase with increasing resolution.
- Memory use of MCNP-UM was found to increase significantly with mesh resolution. This occurred at the initial processing stage, and was related to the presence of parts with a high element count. Mesh #4 originally proved intractable, and the Attila4MC feature was used to automatically subdivide such parts.

OTHER BENEFITS AND LIMITATIONS

Mixed geometry

Both MCNP and Serpent-2 support mixed geometry types i.e. some cells filled with CSG, and others with mesh. MCNP is, however, limited to a single mesh universe which must not be clipped by the boundary (MCNP6.2 will support multiple universes). Serpent-2 is more general in this regard, and supports an arbitrary number of geometry types with no restrictions on boundary clipping. Serpent-2 also does not suffer geometry errors due to coincident boundary surfaces in CSG models.

DAG-MCNP requires that the entire geometry is created in CAD for faceting as a single model. In practice

this means that a full and complete clean reference CAD representation of the geometry is needed.

Implicit background

All mesh-based codes and methods support implicit background, hence there is no requirement to define the space between components unless desired.

Initialisation time

It was noted that the initialisation time for MCNP-UM rapidly becomes significant when individual mesh parts contain a large element count. As per developer recommendations [1], such parts should be split into multiple smaller ones. This was performed using a feature of Attila4MC, which is recommended for use in MCNP-UM applications.

Parallelism

Hybrid parallelism (MPI with OpenMP threads) is beneficial to reducing the memory footprint of parallel applications on multi-core infrastructure. Both Serpent-2 and MCNP6 support this. As yet, DAG-MCNP does not, which may limit the ability to parallelise calculations on larger models.

CONCLUSIONS AND RECOMMENDATIONS

A study has been conducted to assess the potential for alternative mesh-based geometry workflows in ITER neutronics analysis using a simplified ITER-like model. Such techniques offer reduced simplification requirements and faster model creation time in exchange for an increase in computational requirements.

Experience shows highly detailed models of ITER port systems take 1 – 2 months to simplify, convert, repair and implement in CSG. Limited simplification and repair of CAD would still be needed when meshing, however it is assumed this timescale would be at least halved.

Based on these results, computational time will be expected to increase by a factor of 2-5 depending on the method and mesh resolution. A typical ITER neutron transport run requires 5 days on 64 cores and a significant time delay is unlikely to be acceptable, however Monte Carlo codes offer high parallel scaling and increased computation times can be avoided by scaling up to more compute cores. The same cannot be said of model simplification activities, which are often difficult to multitask and available personnel effort is usually limited. In this regard computational parallelisation is more feasibly achievable.

A meshed neutronics model of the full ITER device is likely to require meshes with > 10^8 elements/facets. Due to the slowing down observed in MCNP-UM with higher resolution models, it is likely that surface mesh approaches (DAGMC, Serpent-2) will be better suited to whole-device modelling. A mixed geometry approach, with a single system meshed in a CSG reference model, may be feasible in MCNP6.

DAG-MCNP is highly robust, efficient, and validated; however it requires the entire geometry to be represented as a mesh and is hence most suitable for models where reference CAD models exist, which is not the case for the ITER neutronics model at this time. Once suitable ITER CAD models exist, DAG-MCNP will prove an extremely powerful tool.

Serpent-2 was found to be marginally slower than DAG-MCNP, though it supports mixed CSG and surface mesh geometry. This flexible approach may be better suited to the currently MCNP-centric ITER reference models, where one could re-use existing geometry. The fully general universe implementation, with no constraints on coincident boundaries or clipping of mesh geometry is highly advantageous for large and complex models. Whilst still under development and currently lacking important features, Serpent-2 has the potential to address many issues facing ITER neutronics workflows.

Given sufficient computation, unstructured geometry workflows have the potential for rapid turn-around of results for ITER neutronics analysis. Designs can be easily modified and optimised, and neutronics will become more efficiently integrated into the engineering design process. The creation of a reference neutronics CAD model for ITER C-model is recommended to enable such CAD-based workflows. It is also recommended that ITER create suitable validation models and acceptance criteria, and that developers validate codes for use on ITER analysis. This will ensure the readiness of computational tool for neutronics analysis of ITER and future fusion facilities.

This work was funded by the RCUK Energy Programme under grant number EP/I501045.

REFERENCES

1. R.L. MARTZ, “The MCNP6 book on unstructured mesh geometry: user’s guide,” LA-UR-11-05668 (2013).
2. P.P.H. WILSON, “Acceleration techniques for direct use of CAD-based geometries in fusion neutronics analysis,” *Fus. Eng. Des.*, **85**, 1759 (2010).
3. J. LEPPÄNEN, “Development of a CAD-based geometry model in Serpent 2 Monte Carlo code," *Trans. Am. Nucl. Soc.*, **111**, 663 (2014).
4. Y. WU, “CAD-based interface program for fusion neutron transport simulation,” *Fus. Eng. Des.*, **84**, 1987 (2009).