Fusion Engineering and Design 89 (2014) 1885–1888

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

Improved algorithms and advanced features of the CAD to MC conversion tool McCad

L. Lu*, U. Fischer, P. Pereslavtsev

Karlsruhe Institute for Technology (KIT), Institute for Neutron Physics and Reactor Technology, Hermann-von-Helmholtz-Platz 1, 76344 Eggenstein-Leopoldshafen, Germany

HIGHLIGHTS

- The latest improvements of the McCad conversion approach including decomposition and void filling algorithms is presented.
- An advanced interface for the materials editing and assignment has been developed and added to the McCAD GUI.
- These improvements have been tested and successfully applied to DEMO and ITER NBI (Neutral Beam Injector) applications.
- The performance of the CAD model conversion process is shown to be significantly improved.

ARTICLE INFO

Article history:
Received 8 October 2013
Received in revised form 14 March 2014
Accepted 6 May 2014
Available online 25 June 2014

Keywords:
McCad
Decomposition
Void filling
MCNP

ABSTRACT

McCad is a geometry conversion tool developed at KIT to enable the automatic bi-directional conversions of CAD models into the Monte Carlo (MC) geometries utilized for neutronics calculations (CAD to MC) and, reversed (MC to CAD), for visualization purposes. The paper presents the latest improvements of the conversion algorithms including improved decomposition, void filling and an advanced interface for the materials editing and assignment. The new implementations and features were tested on fusion neutronics applications to the DEMO and ITER NBI (Neutral Beam Injector) models. The results demonstrate greater stability and enhanced efficiency of McCad conversion process.

1. Introduction

The automatic conversion of CAD models into the semi-algebraic geometry representation as utilized in Monte Carlo particle transport simulations is the most significant function of McCad [1,2]. In general, CAD software adopts the BREP (Boundary-Representation) to describe the geometry of a solid. MC transport codes, on the other hand, adopt the CSG (Constructive Solid Geometry) representation for the geometry which employs combinations of simple elements to describe a solid (such as primitive solids and half-spaces of analytic surfaces) [3,4]. Therefore, the conversion between CAD and MC geometry models is actually a BREP to CSG conversion. Furthermore, MC geometry models need to define the whole space of the problem geometry including the regions in and around the material solids. Such void space is usually not included in CAD models and must be filled. The generation of the void space in the converted model is therefore another significant processing step.

Although McCad tool is capable of performing the complete CAD to MC geometry conversion and has been successfully used in many fusion applications, e.g., JET, ITER, IFMIF and EU-DEMO [5], there were the two main disadvantages especially for the processing of models with a large number of entities or with complex geometries: the conversion algorithms were very time-consuming and error-prone. This imposed a severe limitation for the conversion of large scale models. In order to improve the stability and efficiency of the McCad conversion process, an investigation was performed on the conversion algorithms including the decomposition and void filling algorithms. Potential causes which lead to errors were detected and avoided by implementing improved algorithms for both the solid decomposition and the void space generation. The advanced version of McCad utilizing the improved algorithms is presented in this paper. In addition, a new interface for the material editing and assignment was developed for generating complete MC input file.

* Corresponding author. Tel.: +49 7260823760.
E-mail address: lei.lu@kit.edu (L. Lu).

http://dx.doi.org/10.1016/j.fusengdes.2014.05.015
0920-3796/

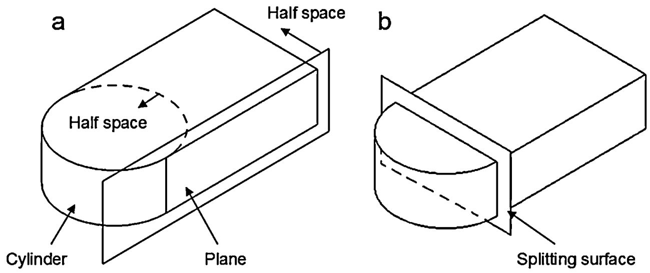

Fig. 1. Splitting surface adding method of original algorithm.

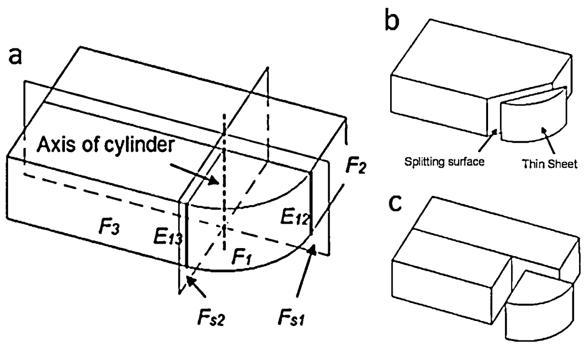

Fig. 2. Scheme for improved decomposition algorithm.

2. Algorithm improvements

2.1. Decomposition algorithm

The conversion from the BREP to CSG representation is the first significant step during the CAD to MC conversion process. Currently, the relatively mature and practical approach is based on the solid decomposition. An input CAD solid with complex geometry is decomposed into a collection of disjoint and simple convex solids, which can be represented by Boolean forms of primitive solids or algebraic half-spaces in MC codes. Generally, half-spaces are adopted more often to describe complex geometries. However, the generation of a sufficient set of half-spaces for a convex solid, based on its boundary surfaces, is a difficult procedure. If the solid consists only of planes, it can be contained entirely by the half spaces of any boundary surfaces. That is, the boundary surfaces of the solid are sufficient to describe the geometry. If the solid contains curved surfaces, such as cylinders, cones, spheres and tori, the boundary surfaces might not be sufficient. Fig. 1a shows an example of a simple convex solid consisting of one cylinder and five planes. The solid can be included by all half spaces of the planes, but only a part of it can be included by the half-space of the cylinder. Thus the boundary surfaces of this solid are not sufficient for describing the geometry and additional surfaces must be added for separating the curved surface from the solid. The original algorithm of McCad generates a splitting surface that utilizes the edges of the curved surface, and then splits the solid into two parts using these surfaces [1], as shown in Fig. 1b.

Theoretically, the principle of this approach is reasonable and available, and any input solid with complicate geometry can be decomposed and eventually described by Boolean forms of half spaces. However, in practice, it was found that the majority of the errors encountered with McCad, including running into an infinite loop and a programme crash, are caused by the curved surface separation. The reason is that when the curvature of the curved surface is small, the original separation approach results in very thin sheets with sharp corners, as shown by the example in Fig. 2b. Unfortunately, the OpenCascade graphic kernel adopted by McCad

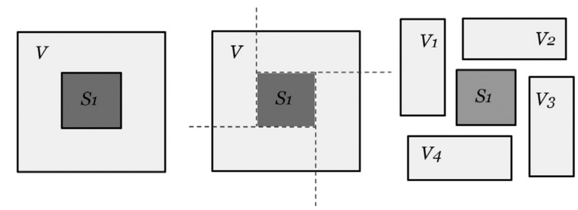

Fig. 3. Scheme for original void filling algorithm.

is unstable and error-prone when performing Boolean operations to such thin sheets. However, curved surfaces with small curvature appear very frequently appeared in CAD models, e.g., the round corners in an engineering model. Therefore, this algorithm can be improved by optimizing the generation of splitting surfaces.

The improved decomposition approach generates the splitting surfaces go not only through the edges but also through the axis of the curved surfaces, which is illustrated on the example of the model shown in Fig. 2a. F_1 is a cylinder that intersects planes F_2 and F_3 at the straight edges E_{12} and E_{13}, which are parallel to the axis of cylinder. Thus a splitting surface F_{s2} for separating F_1 and F_2 can be created through the edge E_{12} and the axis. The normals of F_1 and F_{s1} are perpendicular at the E_{12}, thereby avoiding the generation of thin sheets with sharp corners in this position. The same procedure is performed at the other edge E_{13} for generating the splitting surface F_{s2} that can separate F_1 and F_3. Eventually, as shown in Fig. 2c, the cylinder F_1 is separated from the solid with these two splitting surfaces, and the input solid is then decomposed into three convex solids whose boundary surfaces are all sufficient. Afterwards, they are translated into cell expressions of the MC code and then combined with the union operator (":" in MCNP syntax) for representing the original input geometry.

Currently, the improved decomposition algorithm is capable of processing models with intersections of a cylinder and a plane or two cylinders. Thus most conversion errors resulting in a programme crash or an infinite loop can be avoided. In addition, the improved decomposition scheme gives a less complex final decomposition result which resembles the manual modelling using the same decomposition strategy.

2.2. Void filling scheme

The second fundamental improvement applied to McCad is on the void filling algorithm which, on the one hand, affects the conversion efficiency and, on the other hand, the suitability of the converted model in MC calculations. The original algorithm implemented creates first a large void box which can contain the whole material solids. Next the box is split into a number of convex solids using the boundary surfaces of the material solids. In a final step, the solids that overlap with material solids are removed and the remaining parts are defined as void spaces [1]. The principle of this algorithm is illustrated by a 2D (two-dimension) example in Fig. 3 and can also be extended to 3D problems. Triangle S_1 is an input material solid and V is a box which is used to describe the outer space of S_1. Note that V contains S_1 entirely. In order to describe the void spaces in box V except for S_1, V is split into four parts using the boundary surfaces of S_1. The part overlapping with S_1 is removed and the union of the remaining four parts V_1, V_2, V_3, V_4 define the filling spaces of S_1 in V. Here V_i is convex and thus can be represented by the intersection of its boundary half-spaces. Therefore, with this approach, the filling void spaces, for 2D or 3D problems, can be expressed as $\bigcup_{i=0}^{s}\bigcap_{j=1}^{r}P_{i,j}$, where $P_{i,j}$ are the half-spaces of the boundary surfaces, and s is the number of filling solids generated, which is determined actually by the number of boundary surfaces colliding with box V.

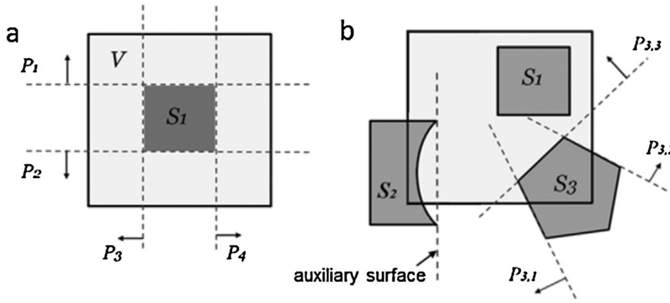

Fig. 4. Scheme for improved void filling algorithm.

Although the void spaces can be filled with this approach adopted by the original version of McCad, it implies two major disadvantages: first, frequent Boolean operations need to be used which are both time-consuming and unstable; second, the number of void solids generated will be larger with increasing complexity of the input material solids.

The new algorithm employs another approach: because the material solids have been already decomposed into a number of convex solids in the first decomposition process, the filling spaces can be described using the complement directly. This is illustrated again on the example of the 2D model shown in Fig. 4. First, the new algorithm also creates a box V that contains S_1. Then it just detects which boundaries of S_1 are colliding with V and calculates the union of the half spaces with respect to these boundaries. S_1 is convex and the intersection of the union and box V can be represented as $V \cap \left(\bigcup_{i=0}^{s} P\right)$ which gives the filling void space in box V. Here $P_1 ... P_i$ are the half spaces of the boundary surfaces in S_1 that are colliding with V. Furthermore, if V contains or collides with multiple solids, as shown in Fig. 4b, the filling spaces of each solid in V are calculated successively. Then the intersection of these spaces defines the filling void spaces in V with respect to the solids. The mathematical expression is $V \cap \left(\bigcap_{i=0}^{r} \left(\bigcup_{j=0}^{s_i} P_{i,j}\right)\right)$. Here r is the number of solids colliding with V, and S_i is the number of colliding boundary surfaces of solid i. However, it is noted that if the colliding boundary surface is a concave curved surface, an additional auxiliary surface needs to be added, as shown by S_2 in Fig. 4b.

If the solids in the box are more complex or more solids are colliding with the box, there are more boundary surfaces colliding with the box. In order to reduce the complexity of the expression for the final void space generated, the box will be decomposed successively into smaller boxes to reduce the number of colliding surfaces with them.

In contrast to the original algorithm, the new algorithm no longer decomposes the solids with Boolean operations, but only employs the collision detection technique to determine the geometric relations between the void box and the boundary surfaces. With this technique, each boundary surface of material solids is first discretized into a group of sample points. If one of them lies in the void box, the boundary surface is colliding with the box. The number of sample points determines the accuracy of the collision detection, which can be controlled by the discretization precision of surface. OpenCascade supplies an efficient function to implement surfaces discretization and sample points calculation. In this way, the programme avoids a large number of Boolean operations and hence is more stable, powerful and time efficient.

3. Material assignment

McCad has been also enhanced with a material management system. To this end, a new user friendly interface has been developed for the material editing and assignment, and added to McCad as a new function. With this function, the user is able to classify the input solids into different material groups, then launch the material editing dialogue box, in which the detailed material information including the name, density and the composition can be edited and then assigned to each group and the solids contained in the group. The materials are saved in an intermediate XML file which is analyzed during the conversion processing and finally output in the MC codes syntaxes to the material cards of the MC input files. Moreover, a database is in the planning stage to enable the user to store and select pre-defined materials directly from an available compilation.

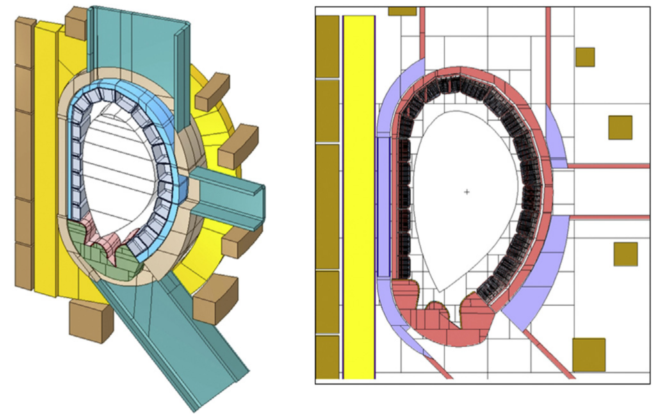

Fig. 5. Application to the generation of a DEMO model.

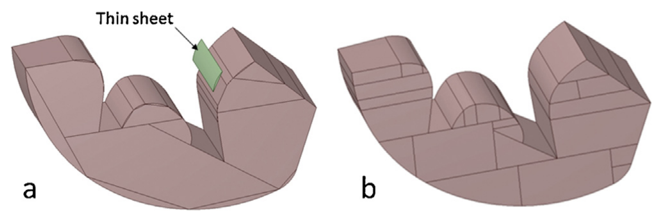

Fig. 6. Comparison of decomposition results with original and improved algorithms.

4. Applications

The improved algorithms and features of McCad have been tested and successfully applied to the CAD to MC conversion of various DEMO models [6,7] performed in the frame of the European PPPT (Power Plant Physics and Technology) fusion power programme, and the NBI (Neutral Beam Injector) port section of ITER.

The improvement of the efficiency achieved with the improved algorithms is demonstrated on the example of the latest DEMO model generation. Fig. 5 shows the input CAD model and final MCNP geometry generated with McCad.

Fig. 6 shows, as an example, part of the divertor that comprises many curved surfaces and demonstrates the decomposition results with the original (Fig. 6a) and improved algorithms (Fig. 6b). It is illustrated that the use of the original algorithm generates a lot of thin sheets. In practice, these sheets make the decomposition process unsuccessful, and manually assisted decompositions are required.

Table 1 shows a comparison of the DEMO model conversion results in terms of the complexity and the required conversion times as obtained with the original and improved algorithms employed in McCad. These results demonstrate that the performance and efficiency of the new version of McCad have been greatly improved, especially with regard to the computing time required for the conversion.Fig. 7a shows the NBI model after simplification and adaptation to the neutronics requirements. The components

Table 1
Comparison of DEMO model conversion results.

	Original algorithm	New algorithm
Number of material cells	14,190	14,190
Number of void cells generated	1031	262
Number of surfaces	3267	3203
CPU time required for the conversion	2 h 40 min	11 min 30 s

Applied PC configuration: CPU: Intel Xeon 2.4 GHz, RAM: 48,000 M.

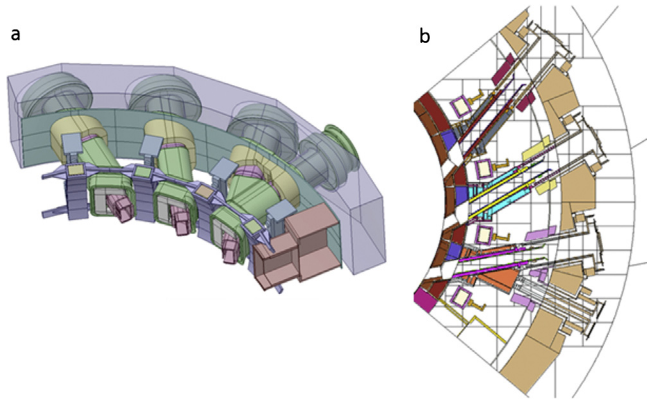

Fig. 7. Application of ITER NBI model.

were divided into several groups according to the different materials they consist of, and afterwards, were imported into McCad. Then the model was decomposed into a combination of a number of convex solids. The void filling function was then used to generate the void spaces and translate them into MCNP representation. Afterwards, the generated input file was merged with the ITER torus sector model for MCNP calculations. Fig. 7b shows a section of the final MCNP input file as produced with the MCNP plotter.

5. Conclusion

McCad is a conversion tool developed to enable the automatic generation of Monte Carlo geometry models from CAD data. McCad utilizes the OpenCascade as graphics kernel and is entirely based on open source software. In this work, improved algorithms have been developed for the decomposition and void filling, and implemented in the McCad code. In addition, an interface for the materials editing and assignment has been designed and added to the GUI. These improvements have been validated and successfully applied to DEMO and ITER. The performance of the CAD model conversion process was shown to be significantly improved. The future development of McCad, on the one hand, will further optimize the CAD to MC conversion algorithm; on the other hand, the development effort will focus on the improved usability of the code for external users including additional interaction features.

Acknowledgments

This work, supported by the European Communities under contracts of Association between EURATOM and the Karlsruhe Institute of Technology, was carried out within the framework of the European Fusion Development Agreement. The views and opinions expressed therein do not necessarily reflect those of the European Commission.

References

[1] D. Grosse, Current status of the CAD interface program McCad for MC particle transport calculations, in: International Conference on Mathematics, Computational Methods & Reactor Physics, M&C, 2009.
[2] D. Grosse, U. Fischer, K. Kondo, D. Leichtle, P. Pereslavtsev, A. Serikov, Status of the McCad geometry conversion tool and related visualization capabilities for 3D fusion neutronics calculations, Fusion Eng. Des. (2013), http://dx.doi.org/10.1016/j.fusengdes.2013.02.146.
[3] X-5 Monte Carlo Team, MCNP – A General Monte Carlo N-Particle Transport Code (Version 5, Vol. I). Report LA-UR-03-1987, 2003 (Rev. 10/3/05).
[4] Open CASCADE Technology, 3D Modeling and Numerical Simulation, 2013, http://www.opencascade.org
[5] U. Fischer, C. Bachmann, B. Bienkowska, J.P. Catalan, K. Drozdowicz, D. Dworak, et al., Neutronic analyses and tools development efforts in the European DEMO programme, in: Presented at 11th Int. Symp. on Fusion Nuclear Technology (ISFNT-11), Barcelona, September 16–20, 2013.
[6] P. Pereslavtsev, L. Lu, U. Fischer, O. Bitz, Neutronic analyses of the HCPB DEMO reactor using a consistent integral approach, in: Presented at 11th Int. Symp. on Fusion Nuclear Technology (ISFNT-11), Barcelona, September 16–20, 2013.
[7] F. Moro, U. Fischer, L. Lu, Applications of McCad for the automatic generation of MCNP 3D models in fusion neutronics, in: Presented at 25th Symp. on Fusion Engineering, San Francisco, 2013, June 10–14, 2013.