The Monte Carlo Method: Versatility Unbounded in a Dynamic Computing World
Chattanooga, Tennessee, April 17-21, 2005, on CD-ROM, American Nuclear Society, LaGrange Park, IL (2005)

CAD INTERFACE FOR MONTE CARLO PARTICLE TRANSPORT CODES

H. Tsige-Tamirat and U. Fischer
Association FZK-Euratom, Forschungszentrum Karlsruhe
Institut für Reaktorsicherheit, Postfach 3640, 76021 Karlsruhe, Germany
e-mail: tsige@irs.fzk.de; fischer@irs.fzk.de

ABSTRACT

The modeling of a complex geometry for Monte Carlo (MC) particle transport problems is a time consuming and error-prone task. Using a Computer Aided Design (CAD) system the effort for the geometry modeling can be reduced significantly. This requires however a suitable interface between CAD systems and MC codes which besides data exchange performs the conversion of CAD data into a representation appropriate for MC codes.

The paper describes the concept and the implementation of a CAD Interface for MC Codes. The conversion of CAD models into a representation appropriate for MC codes is described. Practical considerations with regard to suitability of CAD models, the correction of modeling errors, and the completion of the MC model by voids are discussed. An implementation of a CAD interface for the MC code MCNP with an application for a tokamak fusion device is presented.

Key Words: Geometry modeling, CAD data Conversion, Monte Carlo Codes, CAD Interface, MCNP, Fusion device

1 INTRODUCTION

The Monte Carlo method for particle transport enables the use of complex 3-dimensional geometry models. However, describing and verifying a complex geometry model are among the most complex tasks of MC particle transport problem setup. A more efficient way is to shift the geometric modeling into a CAD system and to use an interface for MC codes. One clear advantage of this approach is that it allows access to full features of a modern CAD system thereby facilitating the geometric modeling. Other advantages are the use of available CAD data after adequate preprocessing steps and the exchange of data with other analysis codes.

An independent interface between CAD systems and MC codes requires access to the data of a CAD system. This can be provided through export of data via neutral file format or by the Application Programming Interface of a specific CAD system. The latter method is CAD system dependent and therefore not considered here. Further, a conversion method for CAD data into a representation appropriate for MC codes is needed. This requires a suitable conversion algorithm. The need for conversion arises due to the difference in the representation schemes used in CAD systems and MC codes.

The paper describes the concept and the implementation of a CAD Interface for MC particle transport codes and is organized as follows. The next chapter is concerned with preliminaries; mathematical framework and geometry representation. The conversion method is presented in section 3. Some practical issues related to the use of CAD data for MC codes are discussed in section 4. Section 5 provides an implementation of a CAD interface for the MC code MCNP [1] with an application to the Joint European Torus (JET). It is shown that a CAD model of a JET torus sector consisting of all significant components is preprocessed and converted into MCNP geometry. The conversion process reproduces fully the CAD geometry with moderate increase in the complexity. Section 6 concludes the work presented.

2 PRELIMINARIES

There are many geometric representation schemes for the shape of a physical object in computers. If the boundary of a shape is algebraic, then a semi-algebraic representation of the shape is among the natural one. The representation of the evaluated boundary of a semi-algebraic set is mostly used by CAD systems.

2.1 Semi-algebraic Geometry

A subset S of Euclidean n-dimensional space $\mathbb{R}^n$ is said to be semi-algebraic if it can be represented by a boolean combination of polynomial equations and inequalities as follows;

$$S = \bigcup_{i}^{s} \bigcap_{j}^{r_i} \{x \in \mathbb{R}^n \mid P_{i,j} *_{i,j} 0\}, \tag{1}$$

where $P_{i,j} \in \mathbb{R}[X_1, \ldots, X_n]$ and $*ij \in \{<, =, >\}$, for $i = 1, \ldots, s$ and $j = 1, \ldots, r_i$. Semi-algebraic subsets of $\mathbb{R}^n$ form the smallest family of subsets that contain all sets of the form $\{x \in \mathbb{R}^n | P(x) \geq 0\}$ and are closed with respect to set-theoretic operation of finite union, intersection, and complement [2]. Semi-algebraic sets represented only by product terms are called basic Semi-algebraic sets.

Any semi-algebraic set can be represented by Equation 1. Such a representation is however not unique. It lacks also any geometric information. Semi-algebraic sets are more descriptive than algebraic sets. Solids in 3-dimensional space such as polyhedron, cylinders, spheres, cones, and boolean combinations of them are semi-algebraic sets.

Further, the closure and interior of a semi-algebraic set is semi-algebraic. Every semi-algebraic set has finitely many connected components, each of them are semi-algebraic. Every semi-algebraic set is locally connected if it is not the disjoint union of two non-empty closed semi-algebraic sets. Connected components have the same sign sequence generated by the polynomials $P_{ij}, \ldots, P_{r_is}$ in the defining formula. If points $X^1 = (x_1^1, \ldots, x_n^1)$ and $X^2 = (x_1^2, \ldots, x_n^2)$ are in the same connected components, then the sign sequence of $P_{ij}(X^1), \ldots, P_{r_is}(X^1)$ is the same as $P_{ij}(X^2), \ldots, P_{r_is}(X^2)$, the sign sequence being 1, 0, or -1.

A point $x \in S$ is said to be a regular point of the semi-algebraic set S if there exists a neighborhood $U \subset S$ of x which is homeomorphic to a manifold of some-dimension. The dimension of non-empty semi-algebraic set is defined to be the maximal dimension of its regular points. Points which fail to be regular in a given dimension are called singular. Every non-empty semi-algebraic set contains at least one regular point. The complexity of a semi-algebraic set is measured by its dimension, the number of defining polynomials and their degree.

A suitable mathematical modeling space for the representation schemes used in CAD systems and MC codes is given by bounded, closed and regular semi-algebraic sets. Any physical object which is rigid, homogeneously three dimensional, of bounded volume, and has algebraic boundary can be represented by such sets. Under the standard Boolean set operations, such sets are not algebraically closed. However, by discarding singular point sets in the Boolean set operation they form an algebraically closed system.

The representation scheme used in CAD systems deals with the evaluated boundary of a semi-algebraic set (a manifold solid). The representations used by MC codes are semi-algebraic cell decompositions. Both representations are closely described in what follows.

2.2 Geometry for MC Particle Transport

The purpose of geometry in MC Particle Transport is to represent physical objects in order to analyze transport processes. Usually, the geometry used for MC particle transport is a decomposition of the problem space into a finite collection of disjoint regions (cells) of constant ray attenuation coefficient whose union is the problem space. Most MC codes represent it by boolean form of primitive solids or algebraic half-spaces.

The geometry for MC particle transport can be formalized as semi-algebraic cell decomposition. To show this, let the problem space $S \subset \mathbb{R}^n$ be represented by Boolean functions with atomic formula of the form $P_i \geq 0$ for $1 \leq i \leq s$, where $P_i \in R[X_1, \ldots, X_n]$ is a polynomial. Such a representation is always possible for sets with algebraic boundaries. The disjunctive normal form

$$\bigcup_{j=1}^{2^s} \bigcap_{i=1}^{s} f_i, \tag{2}$$

where f_i either $P_i \geq 0$ or $P_i < 0$, is a partition of $\mathbb{R}^n$. The product terms $C_j = \bigcap_{i=1}^{s} f_i$ are disjoint, and either empty or connected semi-algebraic cells. If all polynomials $P_1, \ldots, P_s$ are linear, then the product terms C_j are convex polyhedra. It follows from $S \subset \mathbb{R}^n$ that S is a disjoint union of some C_j. A pruning of excessive terms in the product results in the usual representation as used in most MC codes. In fact, every semi-algebraic set is the disjoint union of a finite number of semi-algebraic sets. Every cell in a decomposition has a finite number of faces and adjacent cells touch each other at common faces. Evidently, the representation of a cell in a decomposition is not unique.

The geometric problem solved in MC particle transport, i.e., the particle tracking problem, can be formalized as a point location in a cell decomposition of semi-algebraic sets. To show this, let the set $S \subset \mathbb{R}^n$ be the problem space. Let $S/\mathcal{D}$ be a semi-algebraic cell decomposition of S. The particle tracking problem, which is essentially a point location problem, can be stated as follows. Let r be a ray segment starting at $r_0 \in D_j \in S/\mathcal{D}$ and ending at $r_q \in D_q \in S/\mathcal{D}$. The problem is then to find all cells $\{D_i, \ldots, D_r\} \in \mathcal{D}$ which contain r. The computational problem is then: given a cell $D_i \in \mathcal{D}$ and a query point $r_q = r - r_0 \in X$, decide whether $r_q \in D_i$. The solution yields then the set of cells traversed by the ray and the cell containing it.

2.3 CAD Representation of Geometry

Among several representation schemes used in CAD systems, the boundary representation (B-rep) is widely used in most commercial CAD systems. It provides unambiguous representation of solids through a convenient data structure. For the current application, only manifold solids are considered. A manifold solid is assumed to be a compact, regular point set whose boundary is composed of closed oriented manifolds of dimension up to 2 [3].

A solid can be represented unambiguously by its evaluated boundary and their topological orientation. The boundary ∂S of a solid S is given by

$$\partial S = \bigcup_{i=1}^{s} (S \bigcap P_i) \tag{3}$$

where $P_1, \ldots, P_s$ are the algebraic sets supporting the solid S . The sets $(S \cap P_i)$ are called faces of S. The boundary ∂S of S is a union of algebraic sets and therefore semi-algebraic. Its unique up to the reordering of the boundary supports. For a manifold solid, the boundary supports are smooth surfaces given by irreducible monic polynomials among with normal vectors pointing to the interior of the solid. Faces in ∂S are either disjoint or intersect at common edges. Three faces intersect at isolated vertices. Edges intersect at vertices. An edge may have two vertices or none.

The B-rep data structure stores the evaluated boundary which includes the geometry of the entities in the the boundary and the topological description of their connectivity and orientation. There are many implementation variants for the data structure of B-reps. In practice, they form a hierarchical structure the solid at the top and vertices at the bottom and each entity is linked with its boundary.

3 CONVERSION BETWEEN THE REPRESENTATION SCHEMES

This Section presents an algorithm to convert a manifold solid given by a boundary representation as used in CAD systems into a semi-algebraic cell decomposition which can be used by most MC codes. The conversion process fully reproduces the CAD geometry. However, there is an increase in the complexity. The resulting cell decomposition is in a normal form as given by Equation 1.

3.1 Problem Definition

As can be seen from the above, the representation schemes used in CAD systems and MC codes are different. In order to use CAD generated geometry for MC particle transport, one has to solve a conversion problem which can be formulated as follows. Given a solid by its boundary representation, find its semi-algebraic description. The algorithm which solves this problem is described in detail in [4]. The solution of the problem proceeds in two steps. The first step is to determine the definability of the solid by the available boundary support set and if this is not the case to enlarge the set so that definability is achieved. For manifold solids supported by algebraic half-spaces, it is always possible to enlarge the boundary support set so that definability can be guaranteed. In the second step, a cell construction is performed for all cells in the solid. The availability of a defining set implies the existence of a sign constant decomposition of the solid by the boundary support set.

3.2 Conversion Algorithm

The algorithm takes as an input a manifold solid in B-rep data structure. It traverses the B-rep data structure and extracts the boundary support set. The first stage of the conversion process is the determination of definability. The problem is to find a set of supports sufficient for semi-algebraic description of the cells defining the solid. In general, the polynomials supporting the boundary of a solid in its boundary representation are not sufficient for its semi-algebraic description except when all are linear polynomials, or equivalently when the solid is a polyhedron.

Let a manifold solid be given by a boundary representation. Note that such a solid is a regular closed set with a compact boundary. Therefore it has a finite set of polynomials supporting its boundary. It is assumed that if two supports intersect, their intersection is transversal. A subset of the boundary is said to be definable if it can be represented as a semi-algebraic set by the available elements of the boundary support set. The whole boundary is definable if every subset is definable.

If a boundary support set is sufficient for a representation of the boundary then it is also sufficient for a representation of the interior of the solid. If the support set is not defining, it can be enlarged to be so. Note that the polynomials occurring in the definition of a solid are usually planes, quadric surfaces, etc. which are irreducible monic polynomials. In this case, a defining set can be generated by adding partial derivatives and resultants to the available set. Applied to the current problem, this gives a constructive method to achieve a sufficient set of polynomials for the semi-algebraic description of cells involved in the representation of a solid. In the present case, both partial derivatives and resultants can be approximated by available data, therefore they need not be computed explicitly.

After definability is established, the next stage is the construction of the cell representation. The availability of defining set of boundary supports allows a sign constant decomposition of the solid. This is performed using sample points in each connected component of the solid. If two points are in the same connected component, then the sign sequence generated by the evaluation of the boundary supports is the same. Thereby the the boundary supports are grouped around the sample points into sign constant sets representing basic semi-algebraic sets containing the said sample points. The resulting representation of the solid amounts to be a union of product terms by the enlarged boundary support set.

Obviously, the conversion process increases the complexity of the representation, measured in terms of number of cells, number of boundary supports, and the degree of the boundary supports. The number of cells generated by the conversion is about double the number of sign changing boundary supports. The number of boundary supports increases by the number of additional supports needed to guarantee definability. There is no increase in the degree of the boundary supports. However, the complexity of the resulting representation is comparable to the complexity of the solid represented by Boolean form of primitives.

4 PRACTICAL CONSIDERATIONS

There are many practical constrains that need to be accounted for when using CAD design models for MC transport analysis. MC codes have inherent limitations with regard to geometry and model complexity. Also the implementation of the conversion algorithm is restricted to manifold solids given by boundary representation with algebraic boundary supports such as plane, quadrics, and torus.

The use of CAD models for MC codes may therefore require several preprocessing steps depending upon the initial CAD model. This effort has to be taken into account specially when using available design models. Among the preprocessing steps are model simplification, model repair, and completion of model by voids. The effort needed to generate a suitable model can be considerable depending on the initial model. Also suitability of a model seems to be difficult to quantify other than limitation posed by MC codes and the conversion algorithm. The following subsections discuss these practical considerations in detail.

4.1 Model Simplification

CAD design models are constructed mainly for visualization and manufacturing purposes and are often over detailed for the need of transport analysis. Hence, a simplification of design models is often necessary. The need for simplification is not limited to the present application. Design models need to be simplified often for applications such as mesh generation and visualization.

With regard to the current application typical simplification tasks include geometric simplification, detail suppression, and decomposition of complex parts. A geometric simplification is needed to replace non algebraic boundary element descriptions by an algebraic one. Among such are free form surfaces which can not be treated by both the implementation of the conversion algorithm as well as most MC codes. Also the conversion algorithm underlying theory is limited to algebraic boundary supports. The geometric simplification need to be performed manually in the CAD system. This process can be expensive and go as far as a new modeling of the part.

As mentioned above, CAD design models are usually loaded with details that are irrelevant for transport analysis while severely complicating the geometry. The suppression of irrelevant details is among the essential steps in model simplification for the envisaged application. Obvious details such as blends, notches, fillets, protrusion, and chamfer are suppressed manually in the CAD system. This step requires also extensive user interaction. In general, there are no measures as how to quantify which detail to suppress. Further recognition of details to suppress is left to the analyst.

It may happen that even after detail suppression the model may be still too complex and require a further simplification by decomposition. In view of complexity limitations posed by most MC codes, this step seems more indispensable.

4.2 Model Repair

CAD models may have errors which are insignificant for applications such as visualization or are not checked for at all and become visible in other applications. There are many possible error sources in CAD models most obvious being tolerance mismatch and limited numerical precision. Among relevant errors for the current application are geometric and topological errors such as gaps and overlaps among boundary entities and small boundary entities. Other types of error mostly caused by modeling error are gaps and overlaps among solids and parts.

In practice, CAD models are not checked for such errors. The current application is aware of this fact and performs error checks and model repair. Algorithms for the identification and repair of geometric and topological errors in B-rep data structure are available from most CAD kernels. For modeling errors which affect the conversion process, identification and repair algorithms are implemented.

4.3 Void Modeling

CAD models include parts (solids), a collection or an assembly of them. The void space between the parts is traditionally not available in the models, though there is no principal problem to do so. On the other hand, MC codes need the whole problem space to be defined. Therefore, it is necessary to find a method to include voids in the CAD models or to generate them somehow. A straight forward method would be to perform the void modeling in a CAD system. This has proven to be not practicable due to limitations in CAD systems and also due to the unusualness of the problem. It was therefore necessary to find a practical method to solve this problem. After evaluating different approaches, an algorithm is implemented which automatically completes a CAD model by voids after the conversion step.

The implemented algorithm exploits the normal form (NF) representation of the converted geometry. That is, the conversion process generates NFs of the solids whose terms are basic semi-algebraic sets, which are product forms. The union of all NFs of a collection or an assembly of solids is again a NF. Applying DeMorgan’s law on the final NF results in the description of the void by the union of the complements which is again a NF. Straight forward Boolean evaluation leads however to complicated void including small voids. Also, the resulting model is anymore in a NF. Further, the Boolean evaluation itself can fail and produce overlaps between a void and a part.

The present implementation of the algorithm solves both problems by term-by-term Boolean evaluation with volume control and failure tolerance. This is achieved with the cost of discretization of the model. The algorithm works as follows. It traverses the data structure of the decomposed solid and extracts the boundary supports. The boundary supports are sorted relative to the universe in a shelling like order. Then a term-by-term Boolean evaluation is performed against the universe with volume control. Failure in the Boolean evaluation is allowed. After this step, the model including the results of the Boolean evaluation is discretized by adaptive rectangular mesh points. Afterwards, an overlap detection is performed on the discrete model. If an overlap is detected, then it is added in to the Boolean equation as a difference in the final result without being evaluated. The final result is now a combination of a normal form with or without differences depending on the existence of an overlap.

The main advantage of this algorithm is its failure tolerance during the Boolean evaluation. Further, it avoids small void through the volume control and generates a normal form. The main drawback is its dependence on discretization which makes it slow and resolution dependent.

5 IMPLEMENTATION AND APPLICATION

5.1 CAD Interface for MCNP

An CAD Interface has been developed for the MC particle transport code MCNP [1]. A preliminary interface program is in test phase currently. The method underlying the implementation is depicted in figure 1.

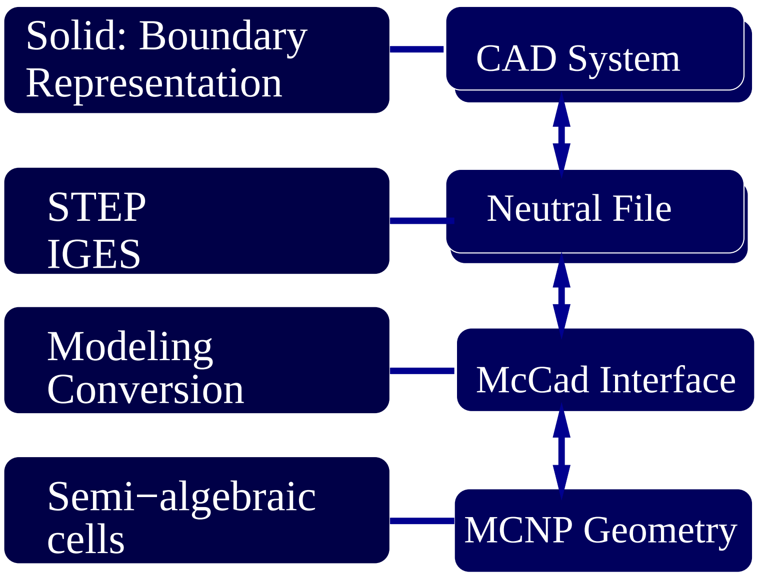

Figure 1: Outline of the method used for CAD Interface for MCNP.

The interface program is CAD system independent and works roughly as follows. A suitable model is generated by a CAD system. The suitability of a model is determined by limitations from MCNP and the conversion algorithm. MCNP supports limited types of surfaces and complexity of cells. The only limitation from algorithmic point of view is that the model should represent a manifold solid or a collection or an assembly of them given in a B-rep data structure. The CAD model is transfered via neutral file format to the interface. Currently, there are two neutral file formats which support the transfer of solids. These are IGES (version 5.3) and STEP. Both file formats are able to transform B-rep data structures accurately. The Interface program imports the model from the neutral file given in either format. It then performs model suitability and error check and if possible repair. Suitability check is limited to geometric properties, i.e., the check if the boundary supports are algebraic. Geometric and topological errors are checked for including check for gaps and overlaps between boundary entities as well as small boundary entities. The next step is then the conversion of the data which is followed by check for overlap among solids and their repair. Finally, the model is completed by voids and output in MCNP syntax.

The interface program integrates a CAD kernel, a C++ GUI application framework, and the conversion algorithm. The CAD kernel provides core data structures, algorithms, and data exchange interfaces for both neutral files IGES and STEP. The GUI framework provides data structures for visualization and user operations. The conversion algorithm heavily relies on the CAD kernel for its geometric and related computations.

The implementation of the interface program is realized in a framework like library. The design approach followed for the interface program is based on object oriented design patterns. The Model-View-Controller (MVC) Pattern is used to divide the interface into three components: The model contains the core functionality and data, views display information to the user, and the controller handle user operations.

Capabilities for visualization, modeling, and data exchange are already implemented. The generation of MCNP geometry representation is treated as data exchange operation. Beside the automatic generation of MCNP geometry representation, it supports visualization and interaction with the model. It is also capable to generate CAD geometry from MCNP geometry representation. This feature is useful for the visualization of MCNP geometry models by CAD systems.

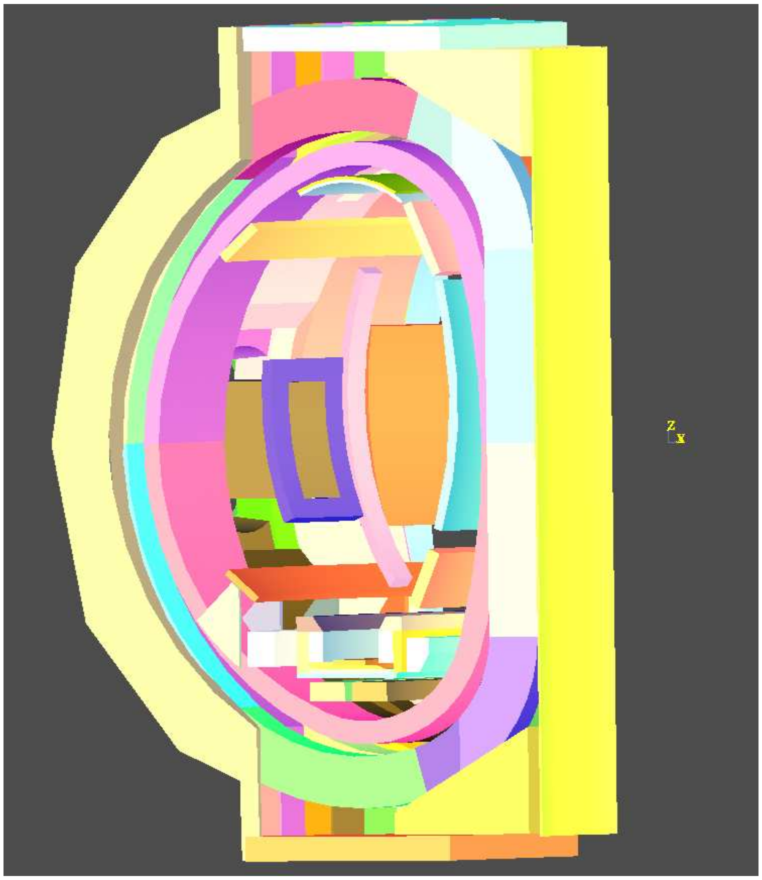

Figure 2: CAD model of the JET torus sector.

5.2 Application to a JET Torus sector

The Joint European Torus (JET) is an experimental tokamak fusion device. Starting from available design models, a suitable CAD model of JET torus sector (octant 3) was generated with the CAD system CATIA v.5. The preprocessing of the model including geometric simplification, detail suppression, and decomposition of complex parts were performed in the CAD system. The final CAD model is shown in figure 2. It contains all major components such as vacuum vessel, mechanical structure, magnets, divertor, etc. There are about 21 parts composed of 80 solids.

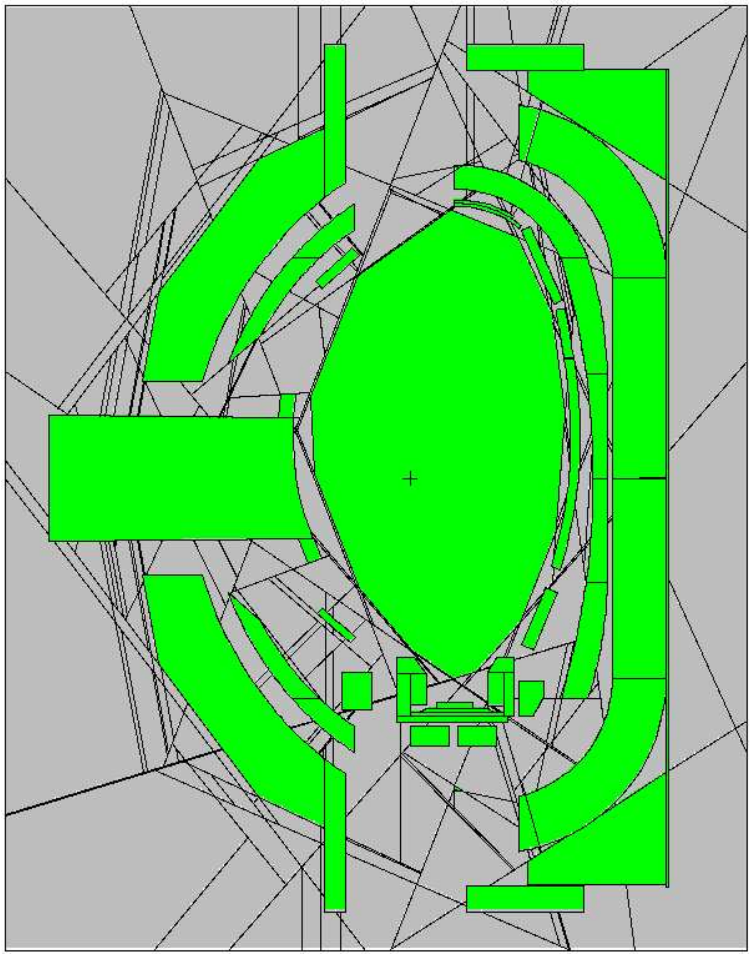

Figure 3: MCNP model of the JET torus sector.

The model has not been tested for geometric and topological errors in the CAD system. There were several errors in the model detected later by the interface. Among them are gaps and overlaps among parts that are dominating. These are repaired during the conversion process while other type errors are repaired prior to the conversion process.

Once a suitable CAD model is generated, it can usually be automatically converted. The conversion process is followed by overlap correction and completing of the model by voids. As mentioned above, the later step is necessary since the CAD model contains only parts and the space among them is not defined. On the other hand, MCNP requires the definition of the whole problem space.

The converted MCNP models of the JET torus sector is shown in figure 3. The parts are shaded in green and the voids in grey. The conversion process does not introduce an approximation. Therefore the converted geometry is fully equivalent to the original one. There is however a modest increase in the complexity of the model in terms of cells and defining surfaces. From a total of 80 solids, 210 sub-cells were produced. Sub-cells refer to product terms in a disjunctive normal form. The increase in the number of surfaces is moderate. After the deletion of redundant faces, the number of surfaces is significantly reduced. The number of voids generated varies depending on the minimal volume admitted. Note that the voids are defined by available surfaces so that there is no contribution to the number of surfaces from void completion phase.

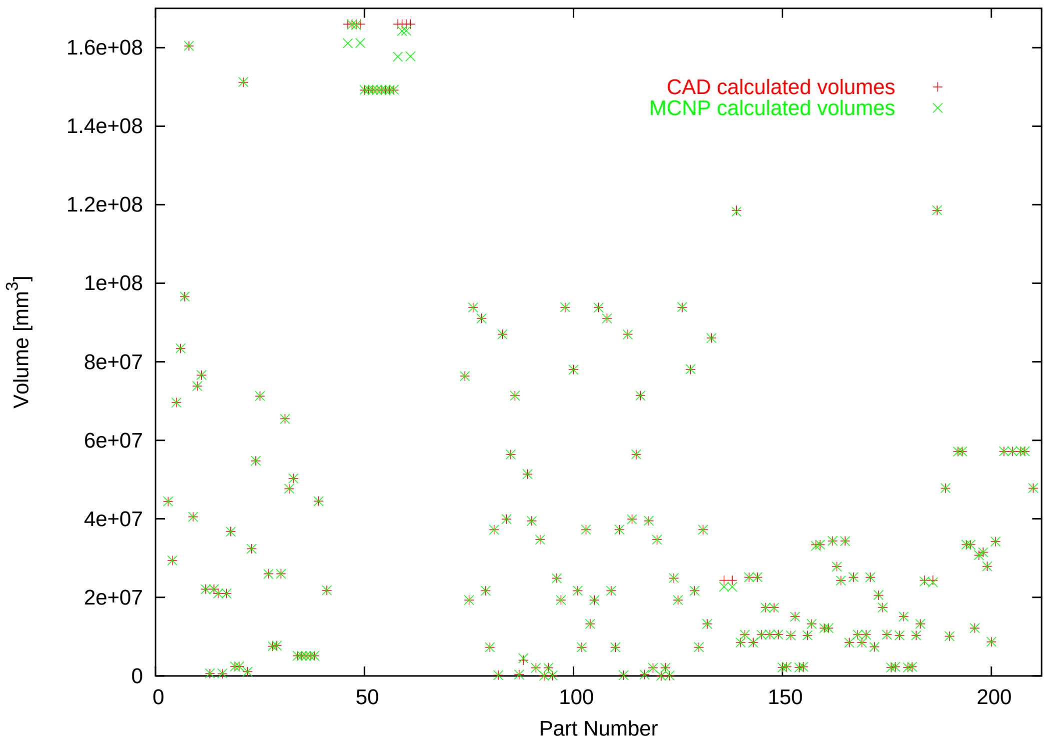

Figure 4: Volumes of parts calculated by MCNP and CAD system.

Using the converted geometry model, volume calculations were performed with MCNP to validate the model. Figure 4 displays the volumes calculated by the CAD kernel used by the interface and MCNP. As can be seen, there are slight differences in the values. The main source of these differences are overlaps among parts which are present in the CAD model. Minor contributions come from statistical errors due to the stochastic volume calculation by MCNP. However, the volume property is well reproduced which confirms that the conversion process reproduces the CAD geometry properly.

6 CONCLUSIONS

A method for the use of geometric data generated by CAD systems in MC Particle Transport codes is presented. The method is CAD system independent and uses neutral files for data transfer. An algorithm is given for the conversion of a CAD geometry into a representation appropriate for MC particle transport.

The method has been implemented into an interface programme for the MC code MCNP. An application is given for a JET torus sector. The converted geometry is shown to be equivalent to the CAD geometry. This has been expected and is in addition to that confirmed by MCNP volume calculations of the converted geometry. There is however a slight increase in the complexity of the model.

A potential bottleneck of the approach is the preprocessing needed to generate a suitable model from a CAD design model. This is not typical for the current application. Most analysis models require similar preprocessing steps. This problem does not arise if a dedicated model for MC codes is generated in parallel with the design model.

Practical experiences with the interface show that it provides a significant speedup in geometry preparation for MCNP. The interface program is currently available only in a test version. It is aimed in a next step to develop a release version.

7 ACKNOWLEDGMENTS

This work has been performed in the framework of the nuclear fusion programme of Forschungszentrum Karlsruhe and is supported by the European Union within the European Fusion Technology Programme.

8 REFERENCES

1. J.F. Briesmeister (ed.), *MCNP - A General Monte Carlo N-Particle Transport Code, Version 4C*, LANL, Report LA-13709-M, 2000.
2. J. Bochnak, et al., *Real Algebraic Geometry,* Springer Verlag, 1998.
3. C. M. Hoffmann, *Geometric and Solid Modeling: An Introduction* , Morgan Kaufmann Publishers Inc., 1989.
4. H. Tsige-Tamirat, *On the Use of CAD Geometry for Monte Carlo Particle Transport*, Proceedings of the Monte Carlo 2000 Conference, Lisbon, 23-26 October 2000, Springer Verlag, 2001.