LA-UR-10-02146

Approved for public release;
distribution is unlimited.

Title:

Implementing MCNP's 21st Century Geometry Capability: Requirements, Issues, and Problems (U)

Author(s):

Roger L. Martz
Tim Goorley
Ryan Clement

Intended for:

American Nuclear Society
RPSD 2010 Meeting
Las Vegas, NV, April 18-23, 2010

Los Alamos
NATIONAL LABORATORY
EST. 1943

Implementing MCNP's 21st Century Geometry Capability: Requirements, Issues, and Problems

Roger L. Martz, Tim Goorley, and Ryan Clement

Los Alamos National Laboratory
PO Box 1663, Los Alamos, NM 87545
martz@lanl.gov; jgoorley@lanl.gov

INTRODUCTION

A new geometry capability has been implemented in MCNP [1] that permits the existence of an unstructured mesh with its legacy Constructive Solid Geometry (CSG) capability to form an hybrid geometry. This new feature enables the user to build complex 3-D models with Computer Aided Engineering (CAE) tools, such as Abaqus [2], and perform a neutron and photon transport analysis on the same geometry mesh that is used for thermo-mechanical analyses.

This paper will present an overview of the issues and problems encountered in implementing the requirements for the hybrid geometry capability in MCNP and a couple applications using this new capability.

ASK THE RIGHT QUESTONS

As 21st Century researchers and code developers, the right questions should be asked to guide us so that future scientific and engineering advances can be made. We know that computer capabilities and capacities will continue to increase, as they have in the past, and will allow us to accomplish more with these superior resources. When will "more" take on the scope of integrated, cross-disciplinary analysis? Is the technology pointing us to "super codes" that will do "everything" in one package? Possibly. But, before that is achievable, don't we need to demonstrate that cross-disciplinary multi-physics analyses are both realistic and worthwhile?

What then is the right direction forward for a code like MCNP to achieve this end? MCNP's CSG capability has served it well for decades and is still of significant use, but is incompatible with the geometry capabilities of mechanical design and analysis programs that rely on finite element methods. Efforts in roughly the last decade have been expended writing CAD-to-MCNP convertors, improving the ability to create complex models, but providing nothing relative to integration with finite element codes. Likewise, effort during this time has been expended on CAD-tracking methodologies so that complex, heterogeneous models can be easily used in MCNP, but no information from this approach is conveniently available to the finite element codes even though the finite element codes rely on solid modeling tools similar to those in the CAD tools for creation of their mesh.

Then, isn't the right question to ask "Should we be developing capabilities to perform radiation transport calculations on the same types of unstructured mesh that the finite element codes use if we expect to one day tightly integrate design calculations across multiple disciplines?" The answer is unambiguously yes. Not only will this approach enable easier multi-physics analysis, it will enable us to use some of the current and future state of the art solid modeling capabilities for the generation of complex, heterogeneous geometry models. Mesh geometries will also present opportunities to improve existing capabilities (e.g., visualization).

HYBRID GEOMETRY

MCNP's hybrid geometry allows the existence of a mesh representation of a solid geometry bounded by surfaces to co-exist with its legacy CSG capability which gives the user the added flexibility of defining geometrical regions from all the first and second degree surfaces of analytical geometry and elliptical tori and then of combining them with Boolean operators. The mesh representation exists inside a MCNP universe with all of its inherent functionality.

In our discussions we often use the term "element' that we have borrowed from the finite element community which uses the terminology "finite elements" to refer to the geometrically simple subdomains into which the geometrically complex domain has been divided [3]. Others may be more comfortable with the terminology "mesh cell" or "cell" when referring to a subdomain in the mesh. However, we will stick with the term "element" since 1) it does not conflict with MCNP concept of a CSG "cell" and 2) it is consistent with the finite element and CAE worlds.

Development of MCNP's hybrid geometry capability has encountered a number of issues and problems that needed to be addressed in order to implement the major requirement of *a geometry capability to handle the needs of 21st Century simulations*. The following five sections discuss these issues and problems while giving the reader an understanding of the progress that has been made.

Code Structure

The MCNP code has been in existence for at least three decades and still carries some of the compromises that were required years ago by hardware and compiler restrictions (e.g., limited memory, common blocks). Efforts have been continuing to *modernize* the code with the newest major version, MCNP6, currently slated for release sometime in calendar year 2011.

The hybrid geometry capability is implemented as a standalone library to MCNP6 with a defined structure and interface. Minimal code changes were made to the existing MCNP Fortran source files with the exception of the addition of mesh control modules to interface between the legacy code and the new mesh library.

Tracking Requirements

Particle tracking in the legacy CSG and CAD geometry capabilities both use a surface-to-surface methodology; that is, until a particle interaction takes place and the collision mechanics routines are invoked. Detailed information or structure on the region interior to the surface is not available unless the user has purposely subdivided the region. An unstructured mesh representation of this region automatically supplies the subdivision; the code only needs to perform element-to-element tracking to obtain results.

Data required for our implementation consist of lists of the node locations, node connectivity for each mesh element, nearest neighbors for each mesh element, elements on each cell surface with the corresponding face(s), material assignment for each mesh element, and mesh elements grouped for tallies. The first two items are supplied by the CAE tool. The second two items are generated by mesh library algorithms. The last two items must be defined by the user in the CAE tool.

The initial implementation pre-processed all elements in the model to generate nearest neighbor lists for all elements. This resulted in a large and noticeable increase in problem setup time when the mesh exceeded ~50,000 elements. The current implementation constructs the nearest neighbor list on the fly when particles enter an element for the first time; thus, amortizing these setup costs with the tracking. An advantage of this is that if a particle never enters an element, the nearest neighbor list is never constructed, thus, saving computer time.

The initial implementation required the user to tag elements on the surface of what is the equivalent of an MCNP cell. In Abaqus, this was accomplished by either creating an element set of mesh elements on the surface or by creating an Abaqus surface; both methods require a specific key word embedded in either the element set or surface name. In the current implementation, we have developed an algorithm that enables the mesh library to do this on its own as part of the problem setup.

Code Integration

Integrating a major new capability into a trusted, gold-standard, legacy code that is feature-rich must be done with the utmost care and attention to detail. A challenging yet desirable design goal for this integration is to allow as many existing features as possible to work correctly with the hybrid geometry. Some features like stochastic geometry may never work inside the unstructured mesh. Although, it may be possible to let a mesh representation be stochastic. This will be an area of future consideration. Other features, such as surface tallies on mesh element faces or mesh descriptions inside lattices, may take additional time to complete.

Every attempt has been made to keep the mesh library features general, but permit them to work seamlessly with MCNP's legacy structure. In a robust CAE tool like Abaqus, it is very easy to collect mesh elements and tag them with an appropriate description. For instance, this is how materials are assigned. It is also possible to tag collections of mesh elements as pseudo-cells so that volume type tallies (e.g., cell-averaged flux, energy deposition, fission energy deposition) can be performed and cell-based variance reduction techniques (e.g., geometry splitting and Russian roulette) can be applied.

Because of this technique of making collections of mesh elements appear to be legacy cells, superimposed features that work on top of the legacy cells automatically work with the unstructured mesh. Among these are the fmesh tallies and mesh-based weight windows. Other features such as point detectors (F5 tallies) and DXTRAN spheres require the completion of a surface-to-surface tracking implementation on the mesh geometry.

Extensibility

The modular mesh library was designed so that it was easily extensible. This has already proven of benefit with the addition of second-order mesh element types. Other mesh types (e.g., optimized Cartesian) require more work, but can be integrated in a straight forward manner.

We chose to work with Abaqus/CAE in this initial implementation due, in part, to its rich and robust set of features and large (i.e., industry standard) user base; Abaqus also contains its own thermo-mechanical solvers which can utilize results from MCNP. Not only does its solid modeling capability allow the easy development of complex 3-D models and the generation of an unstructured mesh using various mesh element types, it permits quick and easy visualization of results. The mesh library structure permits easy integration of other CAE solid modelers through the addition of input modules that translate the solid model description into the data structures needed for the mesh library.

To date, we have been able to take surface-data or shape files of objects (e.g., satellites and asteroids) generated by other means, convert them for use, and work with them in an hybrid geometry calculation.

Results analysis and visualization are achieved with the aid of a special, self-describing output file that the mesh library generates. Both ASCII and binary versions of this output file contain meta-data for ease of reading. Appearing in this file are both the results and a generic description of the geometry.

Performance Issues

Tracking on an unstructured mesh, particularly one with a fine granularity, definitely takes more time than one constructed with the legacy CSG capability. We have found that mesh element type and mesh granularity are drivers for the problem runtimes. Fewer numbers of 2nd order mesh elements are needed to accurately represent objects with curved surfaces and this directly translates into shorter runtimes even though 2nd order tracking algorithms are more complex and expensive to use.

Surface-to-surface tracking has not been fully implemented at this time. When complete, this should greatly reduce runtimes in regions of the phase-space where detailed results are not needed. We will have the surface-to-surface and element-to-element tracking selectable at the pseudo-cell level.

Neither the methods nor underlying coding for the unstructured mesh tracking routines have been optimized. We expect that there are gains to be made here via optimization, but must wait until more of the integration with other features is complete. With the implementation of a deterministic adjoint weight window generator that is integrated with the hybrid geometry, the performance issues may be partially addressed. This is another example of the mesh geometry possessing the potential to improve an existing capability.

As with all features in MCNP, the mesh library supports MCNP's parallel execution with both MPI and threads. As intended with MCNP's modular code development efforts, the MPI is implemented through MCNP's DOTCOMM message passing library and not with direct calls from the mesh library.

APPLICATIONS

The unstructured mesh capability has been tested for two different applications. We will briefly discuss each below.

Urban Consequences

The unstructured mesh capability has been used to represent detailed urban geometries for Los Alamos National Laboratory's (LANL) Advanced Simulation and Computing (ASC) Urban Consequences project. This effort calculates and analyses the effects of improvised nuclear devices (INDs) in urban settings. Specifically, MCNP has transported Fat Man and Little Boy neutron and photon leakage spectra through unstructured mesh models of New York City's Times Square, Figure 1, and Las Vegas, NV. With these geometries, MCNP calculates particle flux, energy deposition, human dose, and neutron induced activation of structural materials. Population dose is immediately useful in planning the response of emergency responders, to help identify the locations where people may be alive and may only survive their radiation injuries with medical attention. Activation from rubble piles and fallout contributes to the radioactivity hazard facing emergency responders.

With the unstructured mesh capability, it is possible to build sophisticated geometric models that need very reasonable memory requirements. The mesh model of Times Square, New York City, consists of 128,232 first order tetrahedrons which explicitly model exterior walls and the hollow interiors of buildings through the use of low density concrete. Individual tetrahedrons vary in size, but are typically 1 - 5 meters. The spatial extent of the model is roughly a square one kilometer (km) on each edge. The model only uses 70 megabytes of memory. The desired goal is to be able to model this level of detail in a square 7 km on edge.

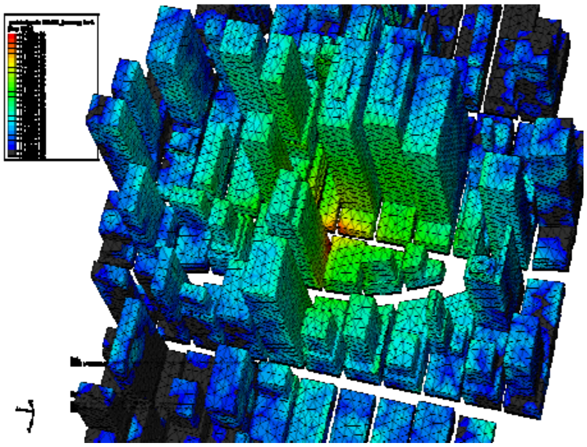

Figure 1. Energy Deposition in Times Square, New York City, buildings. The hypothetical detonation location, at the center of the image, is at the intersection of 7th Ave and 49th Street.

The Abaqus – MCNP link capability also offers the ability to import Arcview shapefiles (.shp) of the major US cities from the U.S. National Geospatial Intelligence Agency's database. Using the program Arcv2CAD [4], shapefiles are converted to AutoCAD files (either DXF or DWG). AutoCAD [5] can then read these files and export them to ACIS SAT files (.sat), which ABAQUS can import. While this process is still being investigated, and complications are expected for increasingly large models, MCNP has transported particles through small (~50 building) models using this process.

Asteroid Mitigation

The unstructured mesh capability has been used by LANL researchers to study asteroid ablation and deflection by calculating energy deposition from the neutrons given off by a Nagasaki-type nuclear device and a hypothetical 14-MeV neutron source. NASA stereolithographic data collected from the Goldstone radio telescope for the Itokawa asteroid (measuring 535 x 294 x 209 meters) was converted to the ACIS SAT files (.sat) that ABAQUS can read from the original NASA (.tab) format. This data provided 20-meter resolution via the ABAQUS unstructured mesh, an accomplishment, which would be practically impossible by a strait-forward application of the standard geometrical primitives native to MCNP. Our first model, shown in Figure2, contained ~129,000 first order tetrahedrons.

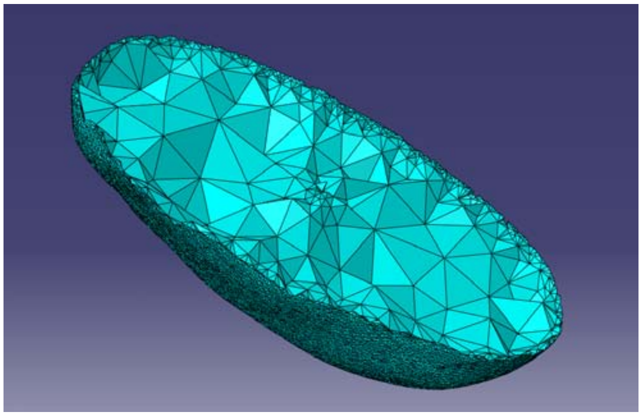

Figure 2. Cut away view of the tetrahedral mesh representation of the asteroid Itokawa, modeled as a solid basalt with a density of approximately 2 g/cc, used in asteroid ablation studies.

In the event of a potentially hazardous object (PHO), such as an asteroid or comet, found to be on a short warning time collision course with the Earth, nuclear explosives are currently man-kind's highest energy density means of preventing a disaster. Further, simply "blowing up" the asteroid could turn one large problem into many smaller problems with equal or greater negative effect. Therefore, the preferred method of deflection is by means of heating the outer volume of the PHO to plasma conditions providing a "gentle" ablation driven push. As gamma and neutron radiation is an important component of the output of a nuclear explosive, an accurate accounting of its effects on the PHO are essential to any credible model.

In addition to accurate transport simulations, we are still faced with the reality that each PHO has a unique geometry, material composition and trajectory. As PHO's come in a large variety of sizes, shapes, and compositions, geometric fidelity is important in accurate models of the energy deposition of prompt and non-equilibrium radiation from a nuclear device. Subsequently, each PHO will respond uniquely to potential deflection efforts. Therefore, we must also be able to perform accurate transport calculations on realistic geometries with sufficient resolution to model the ablation process. Current research models based on the asteroid Itokawa have evolved from our initial work shown in Figures 2 and 3. These current models contain in excess of 2,000,000 first order tetrahedrons with sub-meter sized side lengths. While these models are still being developed and modeling methodologies are still under investigation, we are approaching the resolutions required for accurate, spatially resolved studies.

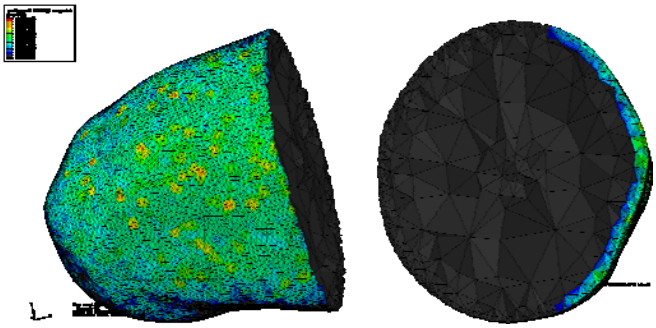

Figure 3. Energy deposition on the surface and into the volume by utilizing the 3D rendering capabilities of ABAQUS and our energy deposition edits. (cut away view).

In addition to geometric considerations we must also be capable of generating enough statistics to have confidence in our energy deposition and momentum deposition estimates. As this capability takes advantage of the parallelization inherent to MCNP, we are also currently exploring the number of particles that are required to obtain sufficient statistics. By example, we can see that the 100,000 particle histories that were used to create Fig. 3 were insufficient as the orange and yellow regions are unphysical. We are currently looking at the effects of greater than 1,000,000,000 histories on our greater than 2,000,000 tetrahedron models. These models are returning dividends as they are pushing the limits of current computing capabilities and improving the overall usefulness of this technology for current and future needs.

This combination of the ABAQUS unstructured mesh capabilities and MCNP radiation transport capabilities provide a crucial new capability in our research efforts toward preventing future asteroid and comet strikes on planet Earth.

CONCLUSIONS

The work to improve MCNP's geometry capability for the 21st century is well under way and positions this tool for use in both stand alone and multi-physics analyses. This endeavor demonstrates that it is possible to integrate a modular code package with a legacy code, with limitations from previous decades, and works successfully with many of the legacy code's existing features. The hybrid geometry capability empowers the MCNP user by allowing the co-existence of different geometry types with different strengths, so that users may select what is best for their application. This capability has been successfully used with several different applications shown here.

ACKNOWLEDGEMENTS

We would like to thank two of the asteroid mitigation team's students, LeAnn Conlon and Catherine Plesko (now a postdoc at LANL), for helping to test this capability and providing the results shown here.

REFERENCES

1. X-5 MONTE CARLO TEAM, "MCNP – A General Monte Carlo N-Particle Transport Code, Version 5, Volume I: Overview and Theory," LA-UR-03-1987, Los Alamos National Laboratory (April 2003).
2. Dassault Systemes Simulia, Inc., "ABAQUS USER MANUALS, Version 6.9," Providence, RI (2009).
3. J. N. Reddy, *An Introduction to the Finite element Method*, Third Edition, McGraw-Hill, Boston (2006).
4. Guthrie CAD::GIS Software Pty Ltd, http:// www.guthcard.com.au/arcv2cad.htm, Version 5.0," Gumeracha SA 4233, Australia (2010).
5. Autodesk, Inc., http://usa.autodesk.com, Version 2010, San Rafael, CA (2010).