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)

CONVERSION OF COMBINATORIAL GEOMETRY TO VOXEL BASED GEOMETRY IN MORITZ

Kenneth A. Van Riper
White Rock Science
P. O. Box 4729, Los Alamos, NM 87544, USA
kvr@rt66.com

ABSTRACT

We describe volume fraction and material mixing algorithms for converting combinatorial surface or solid body models to grid-based geometries. The algorithms are implemented in Moritz, a geometry editing and visualization program. Support for rectangular and cylindrical meshes includes display, interactive creation by the user, and automatic generation by the program. The volume fraction algorithm computes material fractions in each mesh cell using a ray tracing method. The user has control over the number N and direction (in rectangular meshes) of the evenly spaced rays. The fractions are summed from the ray length in each material. The algorithm can use a fixed number of rays in each mesh cell, N and N + 1 rays, or an iterative method in which the number increases until the fractions have converged to a selected tolerance. With the latter two methods, the difference in fractions between the last two sets of rays gives an indication of the accuracy of the results. An optional first pass with a smaller number of rays may be used to test if the mesh cell contains a single material. The material mixing algorithm defines new materials based on the fractions in each mesh cell. The mixing uses materials defined in MCNP(X) format. The converted voxel geometry can be written in MCNP format, either as a lattice or as cells, for use in the ORANGE code.

Key Words: Moritz, MCNP, MCNPX, Orange, Sabrina

1 INTRODUCTION

Most radiation transport programs describe the problem geometry by one of two basic methods. Combinatorial geometry descriptions, in terms of surfaces and/or solid bodies, are typically used by Monte Carlo codes. Mesh based descriptions are employed in discrete ordinates and other methods. Conversion from one type of geometry description to the other is desirable for analyzing the same model with different transport methods. Initial geometry definition with a combinatorial scheme may be more convenient for some users even if the intended use is in a mesh based transport code. Translation from combinatorial to mesh geometry requires an adequate mesh and the assignment of materials to each mesh cell or *voxel*. When a mesh cell encompasses regions of different materials, the translation algorithm must compute the proper fraction of each material.

Use of a voxelized geometry in a transport code based on combinatorial geometry is of interest in several situations. Many medical physics transport models use voxel geometries based on CT, MRI, and other tomographic scans. Some researchers are interested in comparing results of combinatorial geometries, such as the MIRD [1, 2, 3, 4] human anthropomorphic models, with voxelized representations of the same models. Some specialized variants of the Monte Carlo

transport codes MCNP [5] and MCNPX [6], such as ORANGE [7], offer a significant reduction in computation time for geometries defined on a regular grid. They may offer an advantage when using voxelized approximations of models that are difficult or lengthy to calculate with standard combinatorial geometry. Many geometry models are created using Computer Aided Design (CAD) software. There are many efforts underway to use CAD models directly in transport codes or to convert the CAD designs to combinatorial geometry. Voxelization may be a preferred conversion option for complicated regions of a CAD geometry. For these and other possible uses of voxel geometry, comparison of transport results with the original combinatorial or CAD model will help determine the resolution (e.g. voxel size) required for a desired accuracy.

The Sabrina code [8, 9, 10] can calculate volume fractions on a rectangular mesh. We investigated the efficacy of the Sabrina algorithms [11] and used the results of that study to design the Moritz [12, 13] implementation of a volume fraction feature. We briefly describe the Sabrina algorithm in the next §, followed by a summary of mesh capability within Moritz. § 4 describes the Moritz volume fractions algorithms. § 5 discusses creation of mixed materials and conversion to MCNP voxel geometry.

2 SABRINA VOLUME FRACTION ALGORITHM

The Sabrina volume fraction algorithm calculates the amount of materials in each cell of a rectangular mesh overlying an MCNP geometry model. The algorithm uses a spatial subdivision process combined with ray tracing. Using a test problem with known fractions, we explored the efficacy of both methods and the consequences of various choices and parameters used in the algorithms.

The spatial subdivision method results in high accuracy but at the price of large calculation times. If a mesh cell or consequent subdivision contains more than 1 material, it is divided into 8 equal volume subcells. The subdivision can continue until a user-specified limit is reached. Figure 1 shows an example in 2D. The method tests only the corners and center of a mesh cell or a subdivision to decide if it contains more than 1 material; this single material test can lead to incorrect positive results in a number of cases.

The ray tracing method alone can achieve desired accuracies without the need for the expensive subdivision process. The ray tracing algorithm employs an iterative process where successively more rays are used until the fractions converge to the same values or a user-specified maximum is reached. The iterative process begins with a single ray. Accurate results obtained only after modification of the method used to determine convergence.

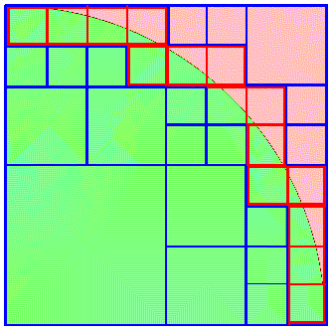

Figure 1. Subdivision of a mesh cell

Using a fixed large number of rays rather than the iterations is more efficient in the presence of curved material boundaries but not necessarily when only planar boundaries are present. Because all rays are tracked in the same direction, whether or not planar boundaries in a mesh cell were parallel or perpendicular to the ray direction affected the convergence and accuracy. Accuracy suffers when the boundaries are parallel to the rays.

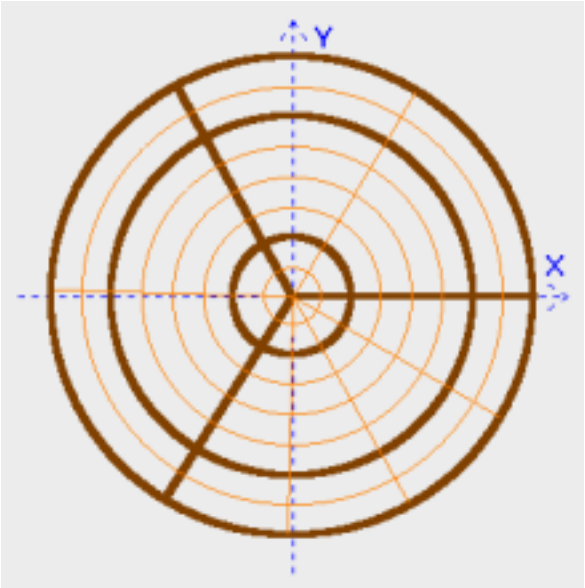

Figure 2. 2D view of a cylindrical mesh

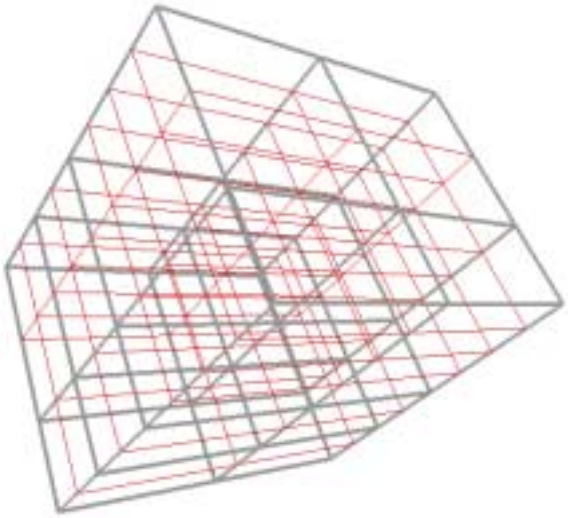

Figure 3. 3D view of a rectangular mesh

3 MESH SUPPORT IN MORITZ

Moritz was designed as an interactive geometry editor. Graphical editing—using the mouse—is available for surfaces, solid bodies, cells, meshes, and other geometry elements. Dialogs in which exact values can be typed complement the interactive editing. Moritz shows the geometry in three orthogonal 2D views and a 3D view. The 3D display is dynamic—the user can rotate, zoom, and pan the image using the mouse and keyboard. Moritz can read and write MCNP/MCNPX input files and convert solid body geometry to MCNP surface geometry.

Moritz includes support for the MCNP superimposed weight window mesh, the MCNPX mesh tally mesh, and the MCNP5 mesh tally mesh. All three mesh types can be defined graphically, on dialogs, or read from the MCNP/MCNPX input specification. Rectangular and cylindrical meshes are supported. We have recently added an automatic adaptive mesh generation algorithm. Meshes are shown in both the 2D (Figure 2) and 3D (Figure 3) views. MCNPX and MCNP5 mesh tally data can be displayed, together with the geometry. Figure 4 shows a 2D mesh tally plot. Figure 5 is a 3D view of multiple mesh tallies, including the data shown in Figure 4. The user has control over which meshes and data sets to show, data scaling, and a number of 3D display options to permit useful views of both the data and geometry.

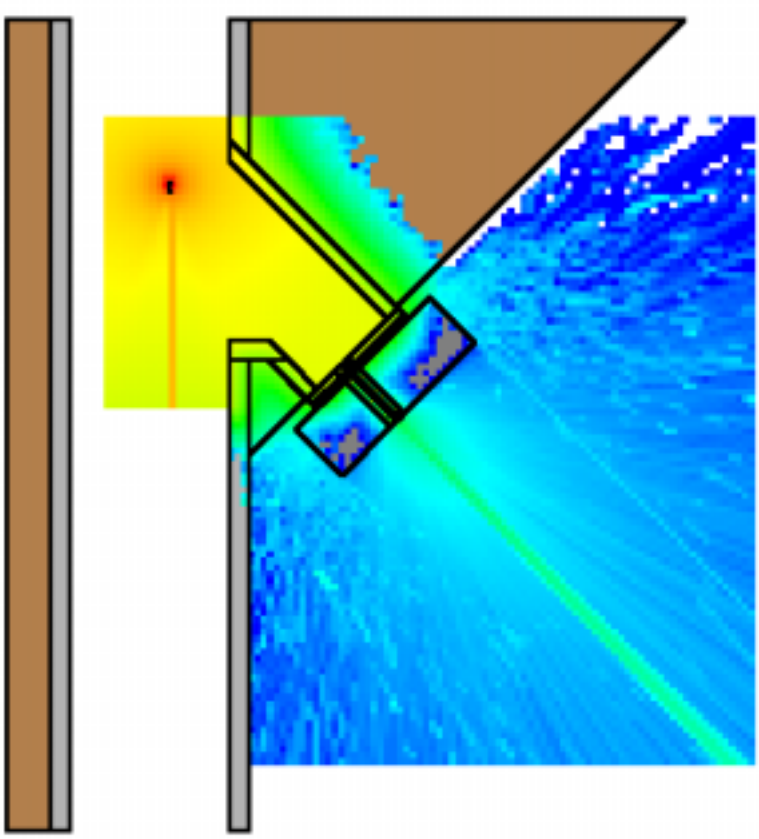

Figure 4. 2D view of a mesh tally

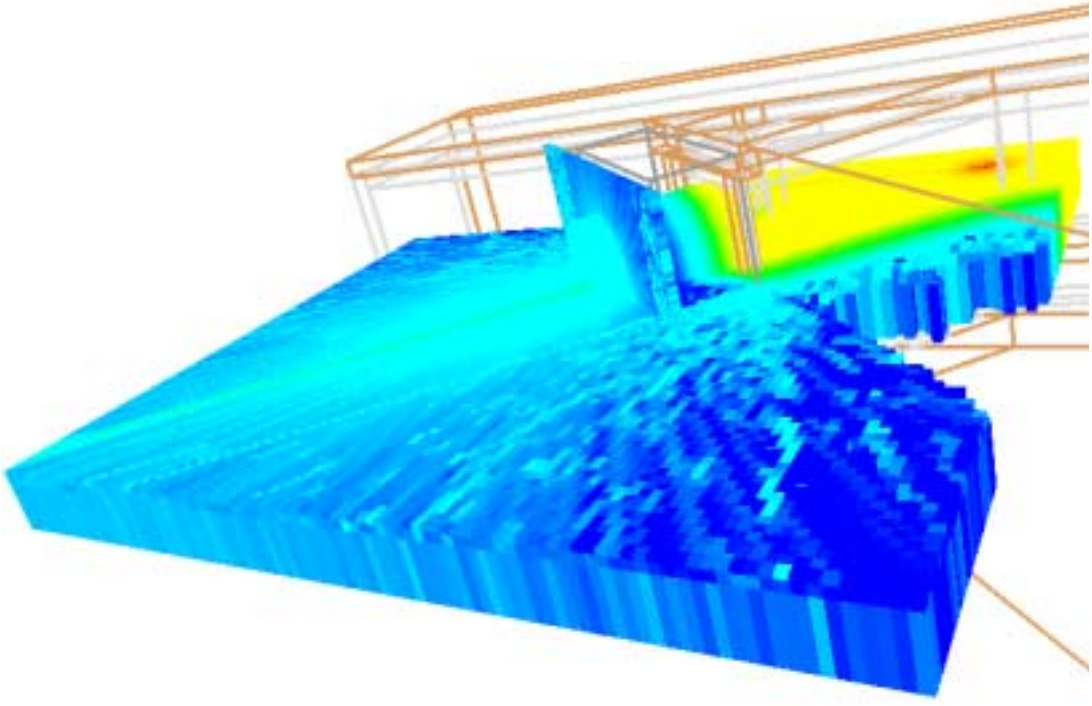

Figure 5. 3D view of a mesh tally

4 MORITZ VOLUME FRACTION CALCULATION

We designed the volume fraction method in Moritz based on the lessons learned from examining the Sabrina algorithms. It is based on ray tracing alone with user control over the number and directions of rays and the strategy to use.

The volume fraction calculation is a numerical integration based on ray tracing across the mesh cell. It works in both rectangular and cylindrical aligned meshes. For rectangular meshes, the rays are parallel to the X, Y, and/or Z axes; 1, 2, or 3 directions can be chosen. For cylindrical meshes, only the axial direction is used for calculating the fractions.

The faction of material *M* for rays in one direction is

$$F_M = \sum_j L_{M\,j} \Big/ D\,J \tag{1}$$

where the sum is over all rays, L_{Mj} is the path length in material *M* for ray *j*, *J* is the total number of rays, and the ray length *D* is the same for all rays in the same direction. When multiple directions are used, the fractions are averaged over directions with equal weighting.

4.1 Volume Fraction Rays

The rays are started at the center of a uniform *N* x *N* grid across a mesh cell face. Figure 6 shows the rays used for *N* = 6 in each direction in a 3D orthographic projection. The mesh cell is outlined in green. The rays orthogonal to the page cross the page at the intersections of the visible rays.

For cylindrical mesh cells, a *N* x *N* grid is created in a rectangle perpendicular to the mesh axis that tightly bounds the axial face of the mesh cell. Each grid position is tested for intersection with the cell. The fractions of grid positions that intersect the cell is used to define a new grid spacing *N'* so that approximately N^2 rays intersect the mesh cell.

To verify that the rays are positioned and directed as intended, during development we added an option to show the rays in the 3D display together with the mesh grid.

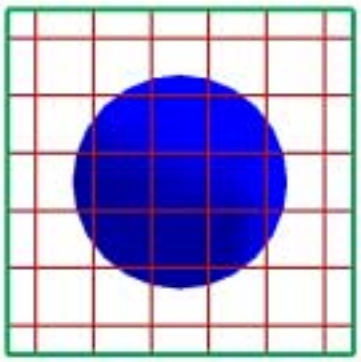

Figure 6. Rays used in a volume fraction calculation

4.2 Single Material Test

One of three methods can be used, optionally preceded by a single material test. The single material test also computes fractions by ray tracing, usually with a smaller grid size N than is used in the main fraction calculation. The ray directions are independent of the directions in the main calculation. For rectangular meshes, any combination of X, Y, and Z directions can be used. For cylindrical meshes, the rays can be in the radial direction as well as the axial direction. Because the radial rays are divergent, fractions based on them are incorrect unless the fraction is 1 or 0. If the single material test finds

only 1 material with a positive fraction = 1, that fraction is used for the mesh cell and no further calculations are made. If not, the fractions calculated for the test are not used in the main calculation.

4.3 Calculation Methods

The main volume fraction calculation uses one of three methods. The *Single N* method uses a single value of *N* in each of the chosen directions. It is the speediest method, but the user must be confident that *N* is large enough for the desired accuracy.

The *N and N+1* method uses two grids, one of size *N* and other of size *N+1*. The difference between the fractions calculated on the two grids gives an estimate of the stability of the calculation expressed in terms of a percent relative error for material *M*

$$\delta_M = 100 \frac{\left| F_M^N - F_M^{N+1} \right|}{F_M^{N+1}}, \tag{2}$$

where F_M^K is the fraction for grid size *K*. If both fractions are 0, $\delta_M = 0$. If $F_M^{N+1} = 0$, F_M^N is used in the denominator of equation (2). The final fraction is the average

$$F_M = \frac{1}{2}\left(F_M^N + F_M^{N+1}\right). \tag{3}$$

The *Iteration* method computes fractions on grids of increasing spacing *n* until the fractions converge for all materials or until *n* reaches a user specified limit *N*. The iteration starts with *n* = 1 or the grid spacing used in the single material test if that test is used. *n* increases by 1 on each iteration. Convergence is reached if

$$\frac{\left| F_M^{n-1} - F_M^n \right|}{\max\left(F_M^{n-1}, F_M^n\right)} \le \varepsilon \tag{4}$$

for all nonzero fractions. The default convergence parameter $\varepsilon = 10^{-6}$. Both the limit *N* and ε can be set by the user.

4.4 Calculating the Fractions

The volume fraction calculations are controlled from a dialog. The calculations can be performed on the entire mesh or on a selected range of mesh cells. While the calculation is underway, the dialog shows the indices of the mesh cell under analysis. The output can be directed to a Moritz window, a file, and/or to the voxel conversion algorithm. In addition to the mesh grid and volume fractions, the output can optionally contain absolute material volumes, uncertainties, parameter values, and calculation time.

It is the user's responsibility to ensure that the various parameters and settings are sufficient to reach the desired accuracy. The uncertainties are a gauge of how much the results vary with changing grid size N; they should not be interpreted as the absolute error of the results. That being said, small errors coupled with a large N is a good indication that the fractions are accurate. The calculated fractions should be compared to fractions calculated analytically where possible.

4.5 Testing

We used several simple MCNP geometry models consisting of 2 or 3 materials, each of which with a known volume. The calculated fractions differed from the actual fractions by no more than the reported uncertainty unless the uncertainty was very large (> 10%).

5 VOXEL CONVERSION

The volume fraction calculation results can be directed to a voxel conversion algorithm that uses the material fractions to define mixed materials where necessary. The current version produces a voxelized geometry in MCNP format.

5.1 Material Mixing

New materials are defined by mixing for mesh cells that contain more than one material. The isotopes of the new material are added with the isotope fraction multiplied by the fraction of the material in the grid cell. The density of the new material is the fraction weighted sum of the constituent material densities.

Before mixing, the composition fractions and density of all materials are converted, if necessary, to weight fractions and to a mass density. The fractions are normalized so that their sum is unity.

Two thresholds can be used to limit the number of mixtures. Materials with fractions below an *ignore threshold* are ignored and the remaining isotope fractions are renormalized so that their sum is 1. The algorithm searches for another grid cell with the same material. Materials are considered to be the same if the difference between each isotope fraction in the grid cells being compared is less than the *same-as threshold*. If the same material is not found in another grid cell, a new material is created.

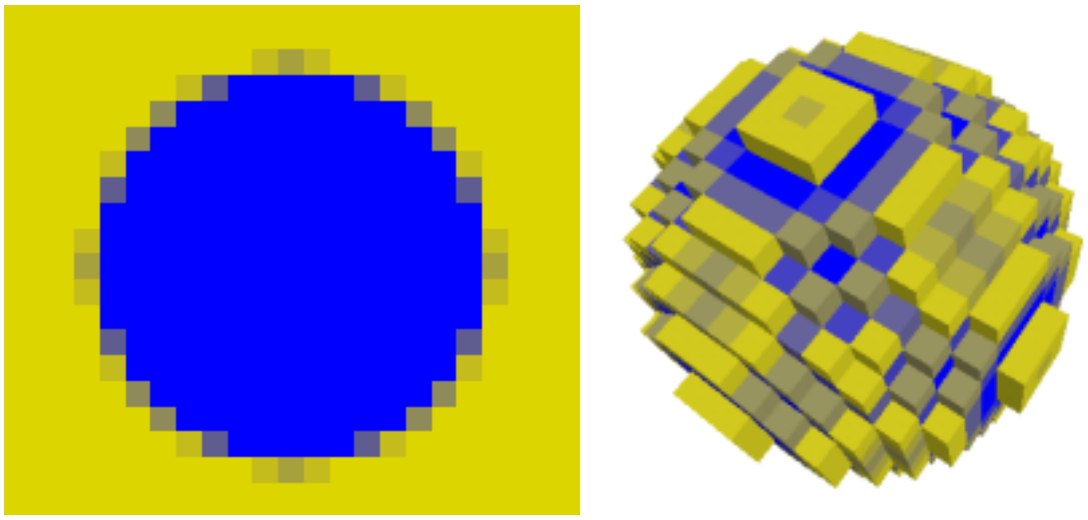

Figure 7. 2D (left) and 3D (right) pictures of a voxelized sphere

5.2 MCNP Geometry

The MCNP geometry may be written as individual cells or as a lattice. The lattice uses a smaller number of cells but adds somewhat to the execution time. When lattice geometry is written, a single spherical cell and universe is created for each material. These cells are centered on the unit lattice cell and are larger than the extent of the unit cell. The lattice elements are filled with one of these single cell universes according to the material at the lattice element position.

Figure 7 shows 2D and 3D views of a voxelized sphere. The colors are mixed between the blue of the sphere and the gold of the surrounding material using the same weighting fractions as used for mixing the materials. Voxels containing pure gold material are not shown in 3D. Figure 8 shows a duct model constructed from portions of 3 tori through a concrete box (green). The blue material inside the duct and at the sides of the box is air. Pure concrete voxels are not shown in the 3D view. Figure 9 is shows the model converted to voxels. Preliminary comparisons of neutron transport through the duct have shown that the voxel mesh is too coarse to give the same result as when using the original model in Figure 8.

Figure 8. 2D (left) and 3D (right) pictures of a duct model

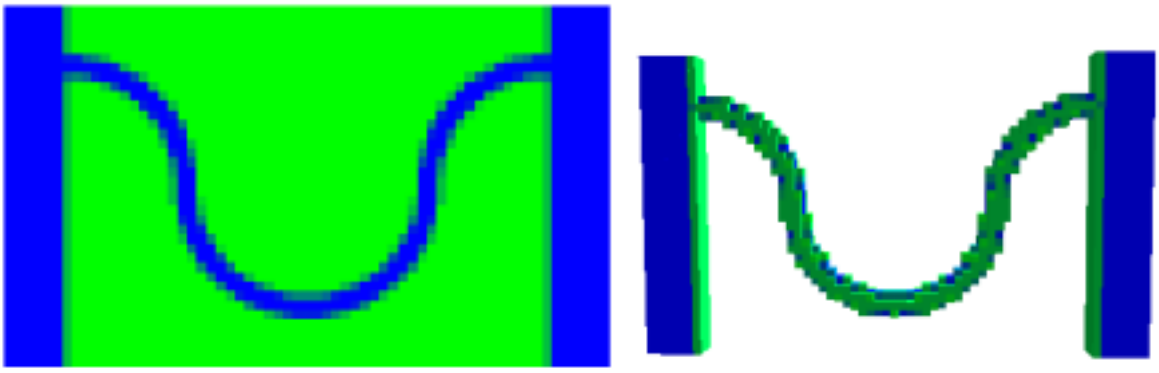

Figure 9. 2D (left) and 3D (right) pictures of voxelized duct model

6 CONCLUSIONS

We have implemented volume fraction, material mixing, and voxel conversion algorithms in Moritz. The user has a choice among several methods for calculating the volume fractions and can adjust the number of rays used to achieve the desired accuracy. The ability to convert an MCNP combinatorial geometry model to a voxelized model in MCNP geometry format will be a valuable tool in future studies of the dependence of a transport result on voxel size. The output of the fraction and material mixing can be formatted for use by deterministic transport codes such as PARTISN [14] that run on a rectangular or cylindrical grid.

7 ACKNOWLEDGMENTS

We are grateful to Steven C. van der Marck and Alfred Hogenbirk for discussions about the ORANGE code and running preliminary calculations of a voxelized model.

8 REFERENCES

1. M. Cristy and K. F. Eckerman, “Specific Absorbed Fractions of Energy at Various Ages from Internal Photon Sources. I. Methods,” *Oak Ridge National Laboratory Report ORNL/TM–8381/VI* (1987).
2. M. Cristy, “Mathematical Phantoms Representing Children of Various Ages for Use in Estimates of Internal Dose,” *U. S. Nuclear Regulatory Commission Report NUREG/CR–1159* (also *Oak Ridge national Laboratory Report ORNL/NUREG/TM-367*) (1980).
3. M. G. Stabin, E. E. Watson, M. Cristy, J. C. Ryman, K. F. Eckerman, J. L. Davis, D. Marshall, and M. K. Gehlen, “Mathematical Models and Specific Absorbed Fractions of Photon Energy in the Nonpregnant Adult Female and at the End of Each Trimester of Pregnancy,” *Oak Ridge National Laboratory Report ORNL/TM–12907* (1995).

4. K. F. Eckerman, M. Cristy, and J. C. Ryman, “The ORNL Mathematical Phantom Series,” *Oak Ridge National Laboratory Report*, available at http://homer.hsr.ornl.gov/VLab/VLabPhan.html (1996).
5. J. F. Briesmeister, Editor, “MCNP - a general monte carlo n-particle transport code,” *Los Alamos National Laboratory Report LA–13709–M* (2000).
6. J. S. Hendricks et al. (15 authors), “MCNPX, Version 2.5.d,” *Los Alamos National Laboratory Report LA–UR–03–5916* (2003).
7. S. C. Van der Marck and A. Hogenbirk, “ORANGE, a Monte Carlo dose engine for BNCT,” *Proceedings of the 10th International Congress on Neutron Capture Therapy*, edited by W. Sauerwein, R. Moss, and A. Wittig, Monduzzi, Bologna, (2002).
8. J. T. West III, “SABRINA: An Interactive Three-Dimensional Geometry-Modeling Program for MCNP,” *Los Alamos National Laboratory Report LA–10688–M* (1986).
9. K. A. Van Riper, “Sabrina User's Guide,” *Los Alamos National Laboratory Report LAUR–93–3696* (1993).
10. K. A. Van Riper, “New Features in Sabrina,” *Proceedings of the Topical Meeting on Radiation Protection for our National Priorities*, Spokane, WA, Sept. 17–21, 2000, pp. 316–323 (2000).
11. K. A. Van Riper, “Analysis of and Refinements to the Sabrina Volume Fraction Algorithm”, *Proceedings of American Nuclear Society Topical Meeting on Mathematics & Computations*, Gatlinburg, TN, April 6-11 (2003).
12. K. A. Van Riper, “Interactive 3D Display of MCNP Geometry Models,” *Proceedings of the ANS International Meeting on Mathematical Methods for Nuclear Applications*, Salt Lake, UT, Sept. 10-13, (2001).
13. K. A. Van Riper, “Creation and Viewing of MCNP/MCNPX Meshes and Grid Tally Data in Moritz,” *Proceedings of the Topical Meeting on Radiation Serving Society*, Santa Fe, NM, April 14-18, (2002).
14. R. E. Alcouffe, R. S. Baker, J. A. Dahl, and S. A. Turner, “PARTISN Code Abstract”, Physor 2000 International Topical Meeting: Advances in Reactor Physics and Mathematics and Computation into the Next Millennium, Pittsburgh (2000).