SURFACE AND INTERFACE ANALYSIS
Surf. Interface Anal. 2005; **37**: 1006–1011
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/sia.2093

SIA

A new Monte Carlo application for complex sample geometries†

Nicholas W. M. Ritchie*

National Institute of Standards and Technology, Gaithersburg, MD 20899-8371, USA

Received 29 November 2004; Revised 27 April 2005; Accepted 27 April 2005

NISTMonte is a new application for Monte Carlo simulation[1,2] of electron transport, X-ray generation and transmission in complex sample geometries. NISTMonte uses the Mott cross section[3,4] to model elastic scattering and the Joy–Luo expression[5] to model energy loss. The ionization cross section is modeled using the empirical expression of Casnati.[6] The mass absorption coefficients are those of Heinrich,[7] and the fluorescence yields are tabulated experimental values.[8]

A defining characteristic of the application is the flexibility with which complex sample geometries may be defined. The application provides a set of basic building blocks that may be combined to build complex sample shapes. Furthermore, shapes may be embedded within shapes to model inhomogeneous samples or structured samples. The application imposes no artificial limits on the number of sample regions or on the number of elements within a material.

A detailed description of the implementation and the results of various tests, including backscatter yields, will be presented to demonstrate the range of validity of the model and the implementation.

The application has been developed in Java J2SE 1.4 for platform independence. The source code will be made available on the NIST web site or by contacting the author. Published in 2005 by John Wiley & Sons, Ltd.

KEYWORDS: electron transport; Monte Carlo; microanalysis; simulation

INTRODUCTION

Monte Carlo simulations of electron transport provide a valuable tool for understanding the various signals that are generated when energetic electrons interact with matter. One field in which they have proven particularly valuable is X-ray microanalysis. The interaction of an energetic electron beam with matter and the resulting generation and subsequent detection of characteristic X-rays is a complex, nonlinear process. Extracting accurate quantitative results from X-ray spectra requires a sophisticated understanding of electron scattering, X-ray generation, absorption and fluorescence. For simple sample geometries, the dependence on mean atomic number, absorption and subsequent fluorescence can be modeled accurately with analytical expressions. For more complex sample geometries, analytical models break down, and it is necessary to approach the problem from a different perspective. Monte Carlo models, while less computationally efficient than analytical approaches, are often a viable alternative.

*Correspondence to: Nicholas W. M. Ritchie, National Institute of Standards and Technology, Gaithersburg, MD 20899-8371, USA.
E-mail: nicholas.ritchie@nist.gov
†Certain commercial equipment, instruments, or materials are identified in this document. Such identification does not imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose.

Many researchers have implemented Monte Carlo simulations for modeling X-ray microanalysis. Some of the better-known implementations are NBSMonte,[9] Joy's Monte Carlo,[10] PENELOPE,[11] Electron Flight Simulator,[12] CASINO[13] and Win X-Ray.[14] Each of these implementations has proven valuable; some continue to be refined and enhanced.

Despite the availability of a range of high-quality Monte Carlo simulations, there remain questions that the current implementations are not well suited to answer. In particular, the current simulations tend to excel at bulk samples or layered samples. Some can handle basic spherical particle samples. The model presented in this article is designed to handle samples of arbitrary complexity. Second, this model will be made available as a library in multiplatform compatible source code. While NBSMonte and David Joy's Monte Carlo models are available in source code, they are more limited in the range of problems they can handle. Electron Flight Simulator, CASINO and Win X-Ray are not readily available in source code form. PENELOPE is capable of complex geometries and is available in source code form but is far more complex and sophisticated.

OVERVIEW OF THE MODEL

NISTMonte uses a single scattering model to track the trajectory of energetic electrons as they interact with matter. While in a material, the electron may ionize a constituent

atom, potentially leading to the emission of an X-ray. NISTMonte also models the absorption and detection of characteristic X-rays by modeling the absorption of X-rays as they travel between the point of generation and user-definable detection point.

NISTMonte separates the process of modeling the electron trajectory from modeling X-ray generation and absorption. The electron trajectory is modeled as two independent processes – elastic scattering and continuous energy loss. The elastic scattering is modeled as a series of straight segments, the lengths of which are determined by an exponentially distributed random variable with scale determined by the mean free path. At the end of each straight segment, the direction of the electron is perturbed by a random scattering event. Following each scattering event, the energy of the electron is decreased to account for the average energy lost as the result of inelastic collisions over the straight segment.

Because X-ray generation events occur infrequently, modeling discrete events is extremely inefficient. Instead, the model accumulates the probability of X-ray events. The probability of an event can be readily interpreted by multiplying the resulting predictions by the ratio of the number of electrons in a realistic measurement by the number of simulated electrons. The number of electrons that can be simulated on a Pentium 4–based desktop computer is approximately 6 orders-of-magnitude less than the number of electrons in a typical electron probe.

As the incident electron travels through the material, it may undergo an inelastic scattering event in which the incident electron transfers energy to a core electron, thereby ejecting the core electron from the sample atom. The probability of ionization is a function of the ionization cross section and the path length. The path length is nominally the distance between scattering events. If a boundary between material regions intersects the segment, then one or more scattering points may be replaced by the intersection between the boundary and the segment for computing X-ray generation.

An X-ray may be generated at any point along the segment. Rather than arbitrarily assign the locus of the X-ray generation to the scattering point (and thus subtly bias the spatial distribution of generated X-rays), the model chooses a new random number, r, from a uniform distribution on the interval [0,1). The X-ray generation probability is assigned to a point that is a fraction r of the distance along the current path segment. Over an ensemble of equivalent scattering events, the X-ray generation probability will be evenly distributed over the length of the segment. This method was selected for its simplicity. NBSMonte uses an alternative mechanism to assign the X-ray generation probability. NBSMonte divides the generation probability among the $\Phi(\rho z)$ measurement bins according to fraction of the length of the path segment in each bin. For a fixed number of trajectories, the $\Phi(\rho z)$ curve generated by the NBSMonte mechanism will be less noisy, but because this mechanism introduces the measurement process into the trajectory model, it is programmatically less clean and less flexible.

It is important to handle the interface between materials correctly. Horiguchi[15] presents a mechanism to calculate the mean free path for a trajectory that passes through one or more materials. While their algorithm is correct, a simpler method is appropriate for Monte Carlo models.

It is sufficient to initiate each trajectory segment only on the basis of the properties of the material at the beginning of the segment. If the segment remains in the initial material, this method is trivially correct. If the segment crosses into a new material, the transition must be handled as follows. The point of intersection between the segment and the boundary must be calculated. The segment must be prematurely terminated at the intersection point. The X-ray generation probability for the completed portion of the segment must be computed on the basis of the properties of the initial material and on the distance traveled. The electron trajectory at the boundary point must be restarted exactly as it would be after a scattering event with one exception – the direction of the trajectory does not change at the boundary. The segment length is chosen at random only on the basis of the properties of the new material and will, on average, have length equal to the new material's mean free path. No compensation is necessary to account for the incomplete segment in the previous material.

This simplified procedure works because, like the proverbial coin flip, past events have no influence on future events or, expressed another way, scattering is a Markov process.[11] Equivalently, the mean distance to a scattering event from any point on the electron trajectory is equal to one mean free path, regardless of whether the electron has scattered recently or not. However, it should be noted that, if the locus of X-ray generation is not selected at random as described in an earlier paragraph (or the probability otherwise uniformly distributed) then the X-ray distribution will be biased because the X-ray generation probability is proportional to the length of the path segment.

The family of X-ray lines and Auger electron energies that can be emitted is determined by the shell from which the electron is ionized. The standard X-ray lines result from ionization of the K, L_1 to L_3 and M_1 to M_5 shells when present. The energy of the X-ray is further determined by the valence shell, which acts as the source for the electron that ultimately decays to refill the unfilled core shell. For example, the transition referred to as the $K\alpha_1$ in Siegban notation or more descriptively in IUPAC (International Union of Pure and Applied Chemistry) notation as the K-L_3 is identified as the X-ray resulting from the ionization of the K shell that is later filled with an electron from the L_3 shell.

The fluorescence yield represents the fraction of ionizations that result in X-rays rather than in Auger electrons. Furthermore, the matrix element for electronic transitions between various shells differ; so the relative intensity of X-ray lines within a family will vary with atomic number. The relative intensity of various lines is further complicated by Coster-Kronig transitions. Coster-Kronig transitions are currently not modeled.

Once the X-ray has been generated, it may be absorbed before it can reach a detector. To model absorption of generated X-rays, the X-ray is assumed to follow a ray

trajectory from the point of generation to the point of detection. The ray will intersect the boundary of one or more sample regions between the generation point and the detector. Each intersection with a boundary defines a path length within a region of a specified material. The material composition, density and mass absorption coefficients determine the fraction of the incident X-rays that will be transmitted through the region. The fraction of X-rays that reach the detector is the product of the transmission fractions for each region between the generation point and the detector. To correctly account for complex sample geometries, the model calculates the absorption from each generation point to the detector.

The trajectory of the electron is tracked until it has a kinetic energy of less than 50 eV. This cutoff was selected as this energy has been defined[16] as the arbitrary cutoff between backscatter and secondary electrons. An electron is considered to be scattered outside the sample if the trajectory intersects with a sphere surrounding the sample. The electron source is located at the top of the sphere. Electrons that scatter into the surrounding sphere in the upper hemisphere are considered to be backscattered and the electrons that scatter into the lower hemisphere are considered to be forward scattered.

MODEL PHYSICS

The choice of elastic scattering model has a large effect on the backscatter yield and, thus, on the predictions of the entire model. Early Monte Carlos used a simple screened Rutherford[17] scattering model. More recent models tend to use scattering cross sections based on the more accurate Mott cross section. NISTMonte implements three different elastic-scattering models – a basic screened Rutherford model, the Mott scattering cross section of Czyzeweski *et al*.,[4] and the Mott scattering cross section of Jablonski *et al*.[3]

The screened Rutherford cross section is a simple analytical approximation. The Mott cross sections are based on interpolations between precalculated, tabulated values. The Mott cross sections were only available for a limited range of energies – in the Czyzeweski case, 20 eV to 30 keV and in the Jablonski case, 50 eV to 20 keV. The limited ranges would seem to limit the utility of the model. However, the Mott cross section results are known to approach the Rutherford cross section at higher energies. Thus, NISTMonte reverts to the Rutherford cross section whenever the Mott cross section is unavailable. Backscatter simulations using each of these three different models are presented in the results section.

The energy loss is often modeled using the Bethe energy loss equation that subsumes an array of complex interactions into an equation that depends primarily on the mean ionization potential. The Bethe energy loss equation has divergence problems when the incident electron has energy at or below the sample atom's mean ionization potential. Joy and Luo[5] made an empirical modification to the Bethe expression that eliminates this divergence and makes the equation behave in a more physical manner at lower energies. The Joy–Luo expression asymptotically approaches the Bethe expression at high electron energies. NISTMonte uses the Joy–Luo expression.

Many empirical or semiempirical expressions exist for the ionization cross section.[18] None of them is entirely satisfactory. In particular, while there are substantial quantities of experimental data for K-shell ionizations, such data is sparse for L-shell ionizations and even sparser for M-shell ionizations. On the basis of the available information, Powell[19] concludes that the best available expression for the ionization cross section is that of Casnati.[6] Casnati's cross section was derived from a careful examination of experimental data for K-shell ionization of $6 \leq Z \leq 79$ and $1 \leq U \leq 20$ where U, the dimensionless overvoltage, is the ratio between the electron energy and the K-shell binding energy. While this expression was based on K-shell ionization data, in the absence of a better alternative, NISTMonte also uses it for L- and M-shell ionizations.

The fluorescence yields used in NISTMonte are the values measured by Bambynek.[8] The relative line intensities within a line family are those of Desktop Spectrum Analyzer (DTSA).[20]

ARCHITECTURE OF THE LIBRARY

NISTMonte is implemented in Java. NISTMonte is divided into a library (a 'jar' file) providing the algorithmic implementation and a series of driver applications that configure and call the library. Java is an object-oriented language in which related data items and functional methods are bound together into an organizational structure called a *class*. Functionality that is common to multiple classes can be specified using an *interface*. An interface is a programmatic contract that specifies *what* a class should do without specifying *how* the class should do it. For example, the `MonteCarloSS.Physics` interface specifies (among other things) that the class must implement a method to calculate the elastic-scattering cross section. (Words that refer to programmatic constructs will be formatted in the `Courier` font.) Various different classes can fulfill the interface contract by implementing the elastic-scattering cross section using the screened Rutherford or one of the Mott cross sections. The various different implementations of the `MonteCarloSS.Physics` interface are then readily interchanged to compare physical models.

A novel aspect of NISTMonte is the mechanism by which sample shapes are defined. All the information necessary to define an arbitrary shape with sufficient detail for the purposes of this model can be distilled to two functions. In the Java code, this requirement is encapsulated by the definition of the `MonteCarloSS.Shape` interface. The definition of this interface is shown in Fig. 1. Any class that correctly implements this interface can be used to define a shape and can be used as the foundation for more complex shapes. NISTMonte comes with implementations of the `Shape` interface for spheres, blocks, cylinders and arbitrary shapes bounded by intersecting planes. In addition, NISTMonte provides implementations of the `Shape` interface to represent the difference of two `Shapes` and the intersection of two or more `Shapes`. Additional implementations of

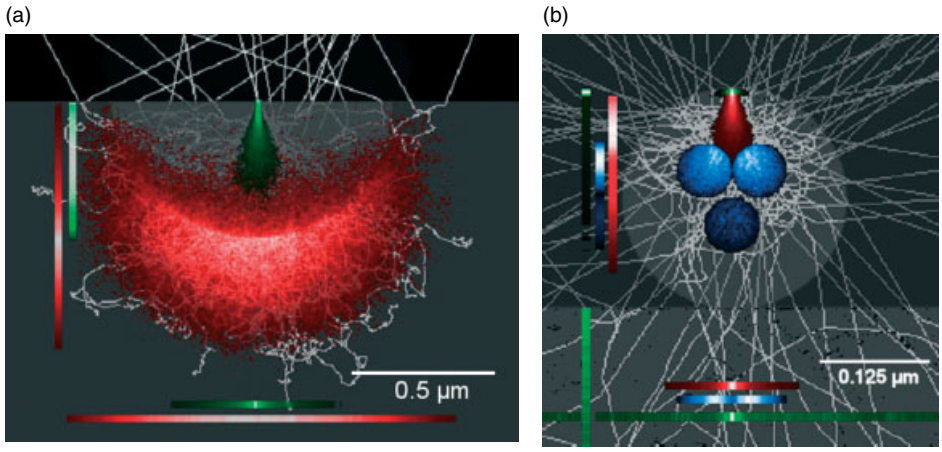

Plate 1. (a) A composite image showing the emission of Mn and Fe X-rays in a 1-µm MnS inclusion embedded in an Fe matrix probed with a 25-keV electron beam. The green and red shading represents the emitted intensity. The brighter the intensity, the more X-rays are emitted from this point. The green corresponds to the Mn K-L_3 (5.9 keV) transition and the red corresponds to the Fe K-L_3 (6.4 keV) transition. The bars at the left and bottom of the image represent the summed intensity across rows and down columns respectively. The gray scale tracks visible around the edges represent the first 100 electron trajectories. The image took 10000 electron trajectories to generate. The relative intensities of each X-ray have been rescaled. The number of Fe X-rays emitted was approximately twice the number of Mn or S X-rays emitted. The absorption was computed assuming a detector located in the upper right of the image at an angle of 40°. (b) A composite image representing 0.25-µm particle consisting of C, Pb, Ba and Sb resting on a C substrate probed with a 25-keV electron beam. The particle consists of a 0.25-µm spherical shell of C surrounding a 0.24-µm spherical Pb particle. Three 0.06-µm Ba–Sb spheres are embedded within the Pb particle. The Pb L_3-M_5 (10.6 keV) transition is represented in red, the Ba L_3-M_5 (4.5 keV) is represented in blue and the C K-L_3 (0.3 keV) transition is represented in green. The gray scale tracks in the background represent the first 100 electron trajectories. The image took 10000 trajectories to generate. The relative intensities of each X-ray have been rescaled. The number of Pb X-rays emitted was approximately equal to the number of C X-rays emitted and eight times the number of Ba X-rays emitted.

```
/**
  * Title: MonteCarloSS.Shape
  * Description: An interface defining a sufficiently rich set of
  * methods to implement a three dimensional volume for the purpose
  * of this model.
  */
 interface Shape {
   /**
    * contains - Is the specified point inside the item represented
    * by this Shape interface?  A point on the boundary between
    * two Shapes is considered to be inside both Shapes.
    *
    * @param pos double[] - a three item array of position coordinates (meters)
    * @return boolean
    */
   boolean contains(double[] pos);

   /**
    * getFirstIntersection - Consider a ray starting at pos0 towards pos1.
    * If the ray does not intersect this shape return Double.MAX_VALUE.
    * If the ray does intersect the Shape, return the u such that
    * p(u)=pos0+u*(pos1-pos0) is the point at which the first intersection
    * occurs.   u must be greater or equal to 0.0.  If u is less than or
    * equal to 1.0, the intersection occurs on the interval [pos0,pos1].
    * u>1 is equivalent to saying that a step from pos0 to pos1 will
    * remain entirely within Shape.
    * @param pos0 double[] - A 3 element array of position coordinates (meters)
    * @param pos1 double[] - A 3 element array of position coordinates (meters)
    * @return double - The fraction of the length from pos0 to pos1 at
    *   which the first intersection occurs.
    * Otherwise Double.MAX_VALUE.
    */
   double getFirstIntersection(double[] pos0, double[] pos1);
};
```

Figure 1. The commented Java code defining the `MonteCarloSS.shape` interface.

the `Shape` interface are planned. For the sake of brevity in the remainder of the document, a class that implements the `Shape` interface is referred to as a `Shape`.

A sample `Region` is defined as a `Shape` made of a `Material`. A `Material` is characterized by its elemental composition and density. `Regions` may be combined to create complex sample geometries. For computational efficiency, there are rules limiting how `Regions` may be combined. `Regions` must be arranged in a hierarchical tree structure with the tree root representing the chamber `Region`. By default, the chamber `Region` is a 20-cm diameter sphere filled with a pure vacuum. However, the chamber may be replaced by a volume of any `Shape` filled with an arbitrary `Material`. The electron trajectory in an environmental scanning electron microscope (SEM) may be modeled by defining the chamber region to have the material properties of a gas of appropriate composition and density. Zero or more `Regions` representing sample volumes may be located inside the chamber. Furthermore, child `Regions` may be embedded within parent `Regions` in a parent–child hierarchy of an unlimited depth. The `Material` properties of a child `Region` supercede the `Material` properties of its parent within the `Shape` defining the child's volume. Child `Regions` must be fully contained within their parent. This restriction is imposed as an optimization. If child `Regions` are fully contained within their parent `Region`, then there are only three alternatives for the location of the end point of any given path segment – the segment may remain entirely inside the current `Region`; the segment may exit the current `Region` and reenter the parent `Region` or the segment may enter an immediate child `Region`. During each step, we need to only test the segment against the current `Region` and its immediate child `Regions`. This can potentially be a small fraction of the total number of `Regions`.

The requirement that child `Regions` are fully enclosed within parent `Regions` might seem to limit the available sample geometries. However, this need not be the case. Consider the example of a `Region` partially embedded in another `Region` like an inclusion embedded in the surface of an iron block. The inclusion is not fully enclosed within the iron block so it cannot be represented as a child `Region`. However, we can use the `Shape` difference class to subtract the volume represented by the inclusion from the iron block. The block and the inclusion are then represented as child `Regions` within the same parent `Region` (e.g. the chamber `Region`.) Other complex shaped `Regions` can be similarly constructed as sums and differences of simpler `Shapes`.

The model structure (including the sample structure) can be defined programmatically (in Java or scripted in Jython[21]) or through XML configuration files. Plate 1(a) and (b) demonstrate some example geometries. Plate 1(a) represents a MnS inclusion in iron. Plate 1(b) represents a Pb–Ba–Sb particle, the forensic marker characteristic of gunshot residue. The intensity of emitted X-rays are

displayed in red, green or blue color scales. The most intense regions are displayed in white with a gradient through red, green or blue to black representing a linear intensity scale. The images are overlaid on an image of the first 100 electron trajectories in grayscale. The electron beam enters the sample from the top center with a downward trajectory. The detector is located in the upper right at a takeoff angle of 40°.

Statistics are collected using an event-driven model. At each scattering point or boundary point, a standard Java-style event is fired. An event is a mechanism by which an event listener class notifies an event source class that the listener is interested in receiving notification when certain events occur. When notified of an event, the event listener is then free to access the state of the model to accumulate statistics on electron or X-ray behavior.

An event-driven model has numerous benefits. First, it loosely couples the statistics accumulators to the electron trajectory model making it easier to develop new accumulators as it becomes desirable to collect different types of model statistics. The base NISTMonte library contains statistical accumulators for the standard $\Phi(\rho z)$ statistics, an accumulator that plots the electron trajectories, an accumulator that collects backscatter statistics, an accumulator that plots X-ray generation intensity and an accumulator that constructs a pseudo-energy dispersive spectrometer (EDS) spectrum by modeling the properties of an X-ray detector. Second, an event-driven model is efficient. By registering only those accumulators that calculate statistics of immediate interest, it is possible to forego potentially expensive unnecessary calculations. The backscatter data discussed in the next section was collected using the backscatter accumulator. The backscatter accumulator does not require X-ray statistics and so the model does not calculate them for this simulation.

EVALUATION

As a partial evaluation of the NISTMonte model, the backscatter yield was modeled and compared to measured values. For the sake of this comparison, an electron was considered to be backscattered if it emerged with a velocity component antiparallel to the incident beam and an energy of greater than 50 eV. The sample was modeled as an infinitely thick bulk sample consisting of a single element with a density typical of a common solid form of that element. (The actual density is not a critical parameter in modeling backscatter yield.) Ten thousand electron trajectories were run for each data point and the number of backscattered electrons was counted. The resulting yields were compared with yields measured by Heinrich.[22] Heinrich attributed a reproducibility of 0.003 and an accuracy of 0.01 to his measurements. Other researchers have made similar measurements.[23,24] These measurements tend to agree with Heinrich's to within a few percent at 10 and 20 keV. The results are plotted in Fig. 2 for beam energies of 10 and 20 keV. The plots compare yields calculated using the screened Rutherford, the Jablonski and Czyzeweski cross sections to Heinrich's measured yields. While the uncertainty in the experimental measurements makes it unrealistic to make definitive statements comparing the two Mott cross sections, the difference between the Rutherford results and the measured values are sufficiently large to suggest that this cross section yields increasingly poor results as beam energy decreases and as atomic number increases. Overall, the Czyzeweski cross section produces results most consistent with Heinrich's measurements across the range of elements at both incident energies. The Jablonski cross section produces consistent results over most of the periodic table but produces yields larger than Heinrich's measurements at low atomic numbers at 20 keV.

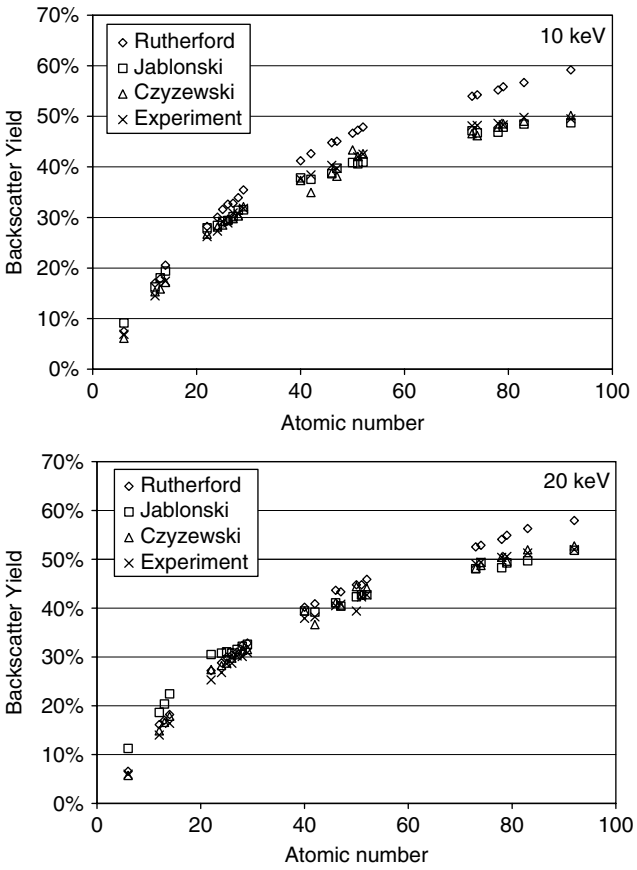

Figure 2. Comparing NISTMonte with different scattering models with Heinrich's experimental measurements of the backscatter yield.

The backscatter yield is a good way to evaluate the electron trajectory model. In particular, if either the total elastic cross section is incorrect or if the shape of the partial elastic cross section is grossly incorrect, then the backscatter yield is likely to be incorrectly modeled. The partial cross sections predicted by the Mott model show nonmonotonic variation with scattering angle. The backscatter yield is not sensitive to this structure as any one backscattering event is most likely the result of many smaller deviations rather than one large deviation. However, the sensitivity of the model to subtle changes in the partial cross section was demonstrated by the modeled yield for the element Rb using the Czyzeweski cross section. The cross section for Rb has anomalously large values at higher scattering angles. This bulge led to a modeled yield that was approximately

20% larger for Rb than for elements with similar atomic numbers.

The backscatter yield does not evaluate the X-ray generation and absorption characteristics of the model. While there are computationally more efficient ways to perform these calculations, NISTMonte can be used to perform the atomic number and absorption correction for quantitative correction of microanalysis data. Preliminary evaluations using NISTMonte to perform standards-based atomic number and absorption corrections on microanalysis data have been promising. This preliminary work suggests that computed $\Phi(\rho z)$ and absorption corrections are reasonable. However, because the work compared an unknown material to reference standards, these tests do not evaluate the ionization cross-section model and fluorescence yield. Further evaluation of the model will be pursued in future research and reported in a future communication.

CONCLUSION

We have presented a new implementation of a Monte Carlo model suitable for microanalysis problems with complex sample geometries. The tests presented in this article demonstrate that NISTMonte can accurately model the backscatter yield for bulk samples. Preliminary tests have shown that NISTMonte can accurately model other microanalytical data. Work continues to further validate the model and to extend it to generate artificial spectra of complex shaped samples as they might be measured on various different types of X-ray detectors.

REFERENCES

1. Berger M. *Methods of Computational Physics*, vol. 1, Adler B, Fernback S, Rotenberg M (eds). Academic Press: New York, 1963.
2. Green M. *Proc. Phys. Soc.* 1963; **83**: 204.
3. Jablonski A, Salvat F, Powell CJ. *NIST Electron Elastic-Scattering Cross-Section Database – Version 3.1*. National Institute of Standards and Technology: Gaithersburg, MD, 2003.
4. Czyzeweski Z, MacCallum DO, Romig A, Joy DC. *J. Appl. Phys.* 1990; **68**: 3066.
5. Joy DC, Luo S. *Scanning* 1989; **11**: 176.
6. Casnati E, Tartari A, Baraldi C. *J. Phys. B: At. Mol. Opt. Phys.* 1982; **15**: 155.
7. Heinrich KFJ. In *Proceedings of the 11th International Congress of X-ray Optics and Microanalysis*, Brown JD, Packwood RH (eds). University of Western Ontario: London, 1986; 67.
8. Bambynek W, Crasemann B, Fink RW, Freund HU, Mark H, Swift CD, Price RE, Venugopala Rao P. *Rev. Mod. Phys.* 1972; **44**: 716.
9. Myklebust R, Newbury D, Yakowitz H. In *NBS Special Publication 460*, Heinrich K, Yakowitz H, Newbury D (eds). National Bureau of Standards: Washington, DC, 1976; 105.
10. Joy DC. *Monte Carlo Modelling for Electron Microscopy and Microanalysis*. Oxford University Press: New York, 1995.
11. Salvat F, Fernández-Varea JM, Sempau J. *PENELOPE – A Code System for Monte Carlo Simulation of Electron and Photon Transport*. OECD Nuclear Energy Agency: Issy-les-Moulineaux, 2003.
12. Small World, LLC: http://www.small-world.net/efs.htm, 2005.
13. Hovington P, Drouin D, Gauvin R. *Scanning* 1997; **19**: 1, 20.
14. Gauvin R, Lifshin E, Demers H, Horny P, Campbell H. *Microsc. Microanal.* 2003; **9**(Suppl. 2): DOI: 10.1017/S1431927603440634.
15. Horiguchi S, Suzuki M, Kobayashi T, Yoshino H, Sakakibara Y. *Appl. Phys. Lett.* 1981; **39**(6): 512.
16. Goldstein J, Newbury D, Joy D, Lyman C, Echlin P, Lifshin E, Sawyer L, Michael J. *Scanning Electron Microscopy and X-ray Microanalysis*. Kluwer Academic Press/Plenum Publishers: New York, 2003.
17. Henrich K. *Electron Beam X-ray Microanalysis*. Van Nostrand Reinhold Company: New York, 1981.
18. Powell CJ. *Rev. Mod. Phys.* 1976; **48**(1): 33.
19. Powell CJ. *Ultramicroscopy* 1989; **28**: 24.
20. Fiori C, Swyt-Thomas C, Myklebust R. *Desktop Spectrum Analyzer*, http://www.cstl.nist.gov/div837/Division/outputs/DTSA/DTSA.htm, 2005.
21. http://www.jython.org/, 2005.
22. Heinrich CFJ. In *X-ray Optics and Microanalysis*. Castaing R, Deschamps P, Philibert J (eds). Herman: Paris, 1966.
23. Hunger HJ, Kuchler L. *Phys. Status Solidi A* 1979; **56**: K45.
24. Reimer L, Tollkamp C. *Scanning* 1980; **3**: 35.