SAND2004-2587P

Validating the Integrated Tiger Series (ITS - 5.0)

Lisa Cordova
Simulation Technology Research Department
Sandia National Laboratories

Code Validation:

Code validation is the process by which the calculated results of a code are checked against real data. It is used to ensure the code is producing accurate results. Validation with experimental data is the usual process for determining whether or not a code will be able to deliver simulation accuracy within an acceptable range of error.

The desired method of validation consists of designing experiments to test pertinent code capabilities, selecting appropriate performance metrics that include both experimental and computational uncertainties, and comparing the results of the metrics to predefined success criteria. However, while designing validation experiments is the preferred approach to begin a validation exercise, it is not always practical to do so. Often data from previously run experiments can be used to estimate the performance of a particular code capability.

The Project:

My task was to validate the current version of ITS by comparing calorimetric measurements of electron energy deposition, obtained from laboratory experiments performed in the late 1970's, to code simulations performed with TIGER, the 1-D transport version of the code. Although the experiments were developed to test the calorimetric method of obtaining dose, this experimental suite was an attractive one for validation. It contained measurements from 50 keV to 1 MeV electron beams incident upon both high and low Z materials, which could be used to assess the code's ability to model electron dose for this range of energy. This experimental suite also included measurement uncertainties that could be used in evaluating code performance.

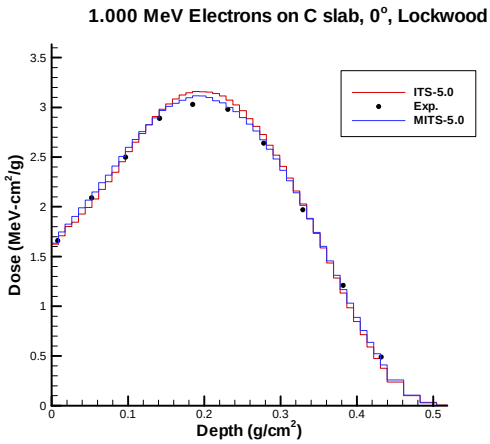
This project was especially challenging for me, because I lacked the basic computing skills I needed for the job. Having very little programming experience and primarily a Windows user, I had to learn Fortran-77 and Linux, and how to run TIGER.

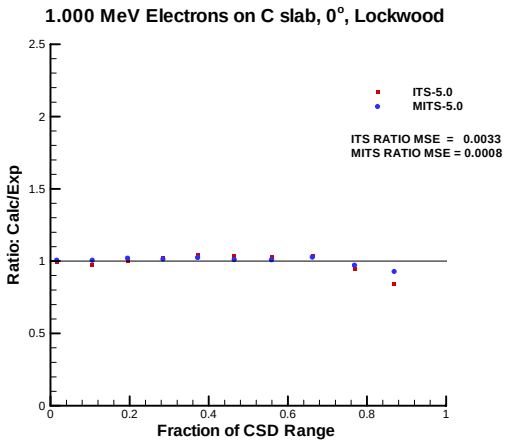
In addition to lacking computer skills, the job was also difficult because the amount of computational error in the code was not known, and finding suitable metrics was an on-going task. Two metrics were used, one graphically comparing a binned simulated dose profile to the experimentally obtained dose profile, and the other metric produced ratios of calculated dose to measured dose at each data point - where the ratios were expected to

be one. Neither one of these metrics contained computational or experimental uncertainties, but it was a beginning.

Because there were nearly fifty experiments to model and because this data set would be used to validate future versions of the code, I wrote shell scripts to run the simulations and to process and archive the results in an automated fashion. One script modified template input files for each job and then ran the jobs sequentially. Another script modified template plots to create graphical comparisons of code results and experimental data.

Below are examples of the two types of graphical comparisons created for this project. The graphs depict how well the two primary executables of Tiger, ITS (uses continuous energy cross sections) and MITS (uses multi-group cross sections), modeled energy deposition from a 1-MeV electron beam incident on carbon.





The graphing utility used to generate the plots was Tecplot. In addition to learning this software and how to write shell scripts, I learned how to use the 1-D Tiger code and how to make minor code modifications to allow my scripts to work with the template input files. This project also gave me an understanding of how important validation is to code development and how difficult it can be to find metrics to evaluate code performance.

Computer Skills Acquired:

Linux Shell Scripting
Programming in Fortan-77
Generating Tecplot Graphs
Running the TIGER Code
Minor Code Modifications

Other Lessons Learned:

As a LINUX novice, it wasn't long before I had a real mess with my files and directories! Consequently, I learned the value of organization and in giving my files meaningful names. I also found that keeping a daily work diary not only helped me to remember the new skills I was learning, but also allowed me to easily return to work I needed to finish.

Acknowledgement
Sandia is a multiprogramming laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy under Contract DE-AC04-94AL85000.