Rob’s Part of the Report

## Automated Optimizer for MCNP Input Decks

The Automated Optimizer for MCNP Input Decks, hence forth referred to as the application, uses User supplied data, points of interest, parameters of interest, and a prepared MCNP Input deck in an attempt to create an experimentally observed detector. It accomplishes this task by iterating through all defined points for each parameter and keeps the ones that minimize the chi squared value, outlined below. Once it has determined this is the best value it can achieve from the iterations it moves on to the next parameter and repeats this process.

The application’s secondary purpose is to automate the collection of relevant data for the purpose of recording what other parameters could be of interest based on how it affects the overall data. This is accomplished by printing out the relative error ref, the average error ref, and plots of the data superimposed upon the experimental data ref in an easy to read format. To put it simply this application reads through the MCNP Output file to retrieve the wanted information and stores it a more human readable fashion. From these outputs the User can determine if an interested parameter is behaving as expected or if it even affects the model in a meaningful way. This should make it easier to determine what values need to be adjusted as well as what could be occurring to cause a difference in the model versus the experiment. An example of this will be discussed in the next section.

For a list of current assumptions, how to change it to work with other detectors, as well as a more detailed outline of how the application works see the User Manual Appendix C.

1. Conclusion

[I need to proof read the conclusion again in order to make sure things aren’t stated twice as well as to fix a few calls to make it match the flow I created above.]

As for the application it currently works as designed, but has too many assumptions to be marketable product. One of the glaring assumptions it makes is that the User will know enough about MCNP to create valid input decks and define parameter ranges that will not break the model.

*Equations*



Where,chi squared, is equal to the sum over all the data points of interest, is the analytical derived point created by the application, is the captured experimental data that the application is trying to match, and  is the uncertainty of that experimental data. The closer chi squared is to one the better fit it is to the experimental data.



Where is the relative error of each pair of points, is the analytical derived point created by the application, and is the captured experimental data that the application is trying to match.



Where  is the mean of the relative error, is the relative error of each pair of points, and is the total number of relative error points that were calculated.