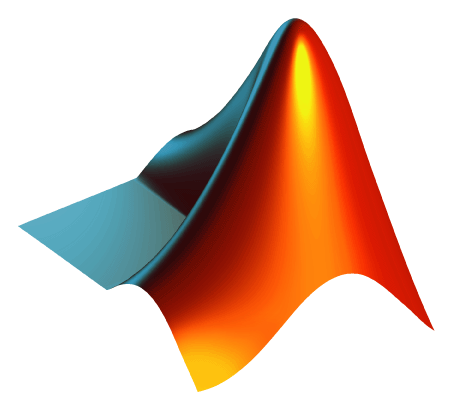
STUDY AND ANALYSIS OF ION INTERACTION AND ITS PROPERTIES USING MATLAB

A Report on the Algorithms and GUI used to produce the simulation.

Jan 2, 2017



Author: S. Mohan Ram

3rd year, ECE DEPT,

Agni College of Technology.

**INTRODUCTON:** Electronics when exposed to radiation tend to behave differently. They are deviated from their ideal characteristics and hence they tend to malfunction. Generally aircrafts and satellites are exposed to radiation caused by cosmic ray showers .To study the effects of such showers and their adverse effect on electronics ,we need to understand them. Matlab is a software tool that allows us to simulate models of cosmic ray showers using simple algorithms. By inserting such algorithms into a user friendly GUI makes it easy for anyone to make use of the software tool.

**OBJECTIVES:**

* Creating different algorithms using Matlab .
* Testing them in various scenarios
* Creating a user friendly GUI to make it easy for anyone to use.

**ALGORITHMS USED:**

1. First Blast based Algorithm.
2. Priority based Algorithm

**Both of the algorithms are scaled down to be visible to the user and to make the computation simpler.** Each algorithm chooses it own path to reach the end , First blast algorithm is considerably faster than priority based algorithm but it comes at a price of loss of valuable data as it considers only one hit per blast, whereas priority consider all the hits hence it is more accurate.

**FIRST BLAST BASED ALGORITHM:**

* The starting point of the blast is defined by the user [It can also be defined as random].
* The user also defines the number of trials, the number of particles to be considered in the atmosphere, the velocity of particle and its energy.
* The particles are plotted at different levels ,in increasing particle density with decrease in altitude.
* A silicon wafer of specified size is considered as the test subject.
* This algorithm simulates a blast at the point specified by the user consisting of a finite number of arms [specified in the program as 10, can be changed].
* Each arm consist of a specified length [specified in the program as 10 points, can be changed].
* The arms stop growing if hit by a particle in the surrounding.
* The arms can grow in all angles within 180° towards the silicon wafer.
* This ensures the area of interest[towards the silicon wafer] is considered and other unwanted possibilities are avoided.
* To do that only the lower portion of the blast is considered hence z axis will be always decreasing randomly. While x and y axis are independent and random.
* Hence only the bottom 4 quadrants are considered.

|  |  |  |
| --- | --- | --- |
| X | Y | Z |
| - | - | - |
| - | + | - |
| + | - | - |
| + | + | - |

* A Point in the direction=[rand(1) rand(1) rand(1)] is taken as the first point ,then it is increased along that direction connecting the starting point the first point and so on till the line length[10].
* To check whether an arm as collided with the particles in the atmosphere a simple distance algorithm is used.
* Two particles separated in 3d space have a distance which is given by the equation
* If the distance is equal to zero which means the overlap each other or in other words collided which each other.
* This process continues till all arms are grown
* At the end the shortest arm is considered as the first blast hence it is the considered as the starting point of the next blast.
* The chain reaction continuous till the altitude where the silicon wafer is kept.
* If the arms of the blast come in contact with the silicon wafer it is considered as a hit .
* A counter variable is updated to make a track of the outcomes.
* If missed another variables is updated to make a track of the outcomes.
* This process continuous tills the specified no of trials as specified by the user.
* Finally all the outcomes along with the probability of hit is displayed on the screen as output.
* The velocity and energy lost in each blast along with its altitude is stored in different arrays for plotting the (Velocity vs. Altitude) and (Energy vs. Altitude) graphs.
* Implementing it in GUI by getting the input data from the user.

ADVANTAGES:

* It is very fast.
* Many trails can be completed in a short period of time ,hence amount of data gathered is more.
* We can keep a track of velocity and energy lost by the particle.
* Easy to understand.
* We do not require a powerful computer.

DISADVANTAGES:

* Only the shortest blast is considered, others are discarded.
* The course of the chain reaction would have been different if all hits are considered.

**PRIORITY BASED ALGORITHM:**

* The starting point of the blast is defined by the user [It can also be defined as random].
* The user also define the number of particles to be considered in the atmosphere.
* The particles are plotted at different levels ,in increasing particle density with decrease in altitude.
* The particles in each levels are stored in separate arrays.
* A silicon wafer of specified size is considered as the test subject.
* This algorithm simulates a blast at the point specified by the user consisting of a finite number of arms [specified in the program as 10, can be changed].
* Each arm consist of a specified length [specified in the program as 10 points, can be changed].
* The arms stop growing if hit by a particle in the surrounding.
* The arms can grow in all angles within 180° towards the silicon wafer.
* This ensures the area of interest[towards the silicon wafer] is considered and other unwanted possibilities are avoided.
* To do that only the lower portion of the blast is considered hence z axis will be always decreasing randomly. While x and y axis are independent and random.
* Hence only the bottom 4 quadrants are considered.

|  |  |  |
| --- | --- | --- |
| X | Y | Z |
| - | - | - |
| - | + | - |
| + | - | - |
| + | + | - |

* A Point in the direction=[rand(1) rand(1) rand(1)] is taken as the first point ,then it is increased along that direction connecting the starting point the first point and so on till the line length[10].
* To check whether an arm as collided with the particles in the atmosphere memory mapping is done.
* To do that the blast particle is compared with all other particles in that level.
* If they have the same x, y and z coordinate it is considered as the particles have collided.
* If collided the position x, y ,z is updated to 3 different arrays(p ,q ,s) of undefined size[because the no. of collisions is undefined].
* This process continues till all arms are grown .
* When one blast is complete the value stored in the arrays( p , q , s) is updated to a , b, c as the starting point of the next reaction.
* The chain reaction continuous till the altitude where the silicon wafer is kept.
* If the arms come in contact with the silicon wafer the program is terminated and output is thrown in the figure window or in the GUI.
* Implementing it in GUI by getting the input data from the user

ADVANTAGES:

* It takes all the points into consideration hence data lost is minimum.
* It is more accurate.

DISADVANTAGES:

* It takes more time.
* It needs a more powerful computer.
* It takes a lot of time to complete one trial hence, it cannot be used for many trials .

**CONCLUSION:**

* We learned to programming using Matlab, and to create a GUI.
* We improved our programming and problem solving skills.