
Subject: Re: back-propagation with GEANE

Posted by [Lia Lavezzi](#) on Thu, 20 May 2010 13:54:54 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi Anastasia,

your code seems correct.

Actually, as you were saying, there is no big difference between

BackTrackToVirtualPlaneAtPCA(1) and PropagateToPCA(1, -1): with the first one you must use the FairTrackParP and with the second one the FairTrackParH.

They do the following:

1) BackTrackToVirtualPlaneAtPCA(1):

- finds the PCA to the point you chose
- builds a plane there
- backpropagates from the starting point to this plane

2) PropagateToPCA(1, -1):

- finds the PCA to the point you chose
- find the track length at that point
- backpropagates from the starting point to this track length

The only difference stands in the input/output type and in the fact that you have or not a plane defined.

Quote:I set origin of coordinates in point (0,0,0) and used BackTrackToVirtualPlaneAtPCA(1) function for back-propagation. As a results I have point near origin of coordinates and momentum in this point.

The backtracked point should contain the true point within the GEANE error.

Quote:I compared my results with true information from MC generator and I observed significant differences in momentum coordinates, which lead to differences between reconstructed and simulated angles of particles. I made such tests with dipole&solenoid&transition maps as well as without magnetic field. Without magnetic field differences between reconstructed and simulated angles aren't so significant, but they still exist.

Does the term "significant" mean that the differences are outside the GEANE errors? In your plots the differences seem small and compatible with roundoff error.

The problem we know is that in forward propagation the energy is taken at the beginning of the step, whereas in backward the energy is taken at the end of the (forward) step. This can be cured, but in principle the difference is not so big, if the stepping is appropriate.

Concerning the accuracy: GEANE propagates the mean values of the parameters, without taking into account the random processes (dE/dx fluctuations, multiple scattering...). If for example you start from the true value and track 1000 particles with geant3 on to a plane, they will be spread because of these random processes, but if you propagate them with geane you will get 1000 times exactly the same value, the mean value of the distribution.

In your case you start from the MCpoints, which are spread around the mean value, and go back to the true vertex.

The amount of the difference between the GEANE backpropagation and the corresponding MC value can be guessed looking at the error calculated by GEANE: this takes into account also the random processes and tells you how much you are wrong in taking the mean value, neglecting the random effects. It depends basically on the path the particle has to travel from start to end point and on the material amount it has to cross. Hence, the pull quantity (MC -

GEANE)/error_GEANE should be a good gaussian. If it is so everything is OK.

There is however a strange thing in the plots: in the error plots there is always a big peak in 0: the error might be small, but it should not be exactly 0, even if you start from errors all equal to 0.

To answer to your question on errors, they are used only in the propagation of the covariance matrix, and not in the calculation of the mean values of the parameters. There is also an option in native fortran GEANE which allows the user to perform the propagation of the mean values without calculating the associated errors.

When you call Propagate it should return a Bool_t true if the propagation succeeded and false if it failed.

Can you please try putting:

```
Bool_t isProp = fPro->Propagate(fStart, fRes, PDGCode);  
if(isProp == kTRUE) new((*fTrackParFinal)[i]) FairTrackParP(*(fRes));
```

and save only the correctly propagated tracks?

Concerning the Sds planes as starting planes, if you use the PropagateToPlane(1, -1) you can build the FairTrackParH directly from the MARS variables and get rid of them. It is true that the errors you set are not the right ones, but as I was saying this should affect only the error calculations, and not the mean values.

Hope this helps,
Alberto and Lia.
