Subject: Re: SPU problem explanation. Posted by Anonymous Poster on Wed, 07 Oct 2009 11:07:54 GMT View Forum Message <> Reply to Message

Hi Lia,

the solution to this unfortunate inconsistency is to treat spu as a real track parameter. In this way it is only predicted and/or updated consistently with the other parameters describing the track. So here is how I did it:

1) I make the state vector 6 dimensional (parameter in GeaneTrackRep.h and some hardcoded 5->6 in the cxx file) so that spu becomes track parameter number 5 (start counting from 0)

2) Very important: Initialize the 5th row and 5th column of the now 6x6 covariance matrix with 0 3) Due to this initialization, the update step in the Kalman will never change the value of spu which cones out of the prediction (The Kalman gain is 6x2 (for a 2D hit) where the last row is just zero, if the covariance has only 0 in the 5th row and column to begin with. If you take my PandaNote and insert the matrices into the equations this becomes clear very easily.

4) Every time you use spu, extract it from the state vector

5) Delete the parameter as a member of GeaneTrackRep

I have done all this in GeaneTrackRep2 which is the class that I use for the external GENFIT. In principle this class should also work inside PandaROOT (together with TGeoManager and FairMCApplication). I will also make it available in the next weeks, and then you can just take a look for yourself. If you want to look now, get genfit at

https://genfit.svn.sourceforge.net/svnroot/genfit and look in the GeaneTrackRep directory.

If you have any further questions, please do not hesitate to ask me.

Cheers, Christian

