Subject: Fitter Exceptions
Posted by Aleksandra Wronska on Thu, 04 Jun 2009 12:35:56 GMT

Dear Genfitters,

I have another wish to the wish-list for genfit. During my recent simulations when using Genfit I quite often had a break and exit caused by two fitter exceptions: "low momentum" and "small theta" ones. I suggest that in the "lazy" mode the occurence of these should not cause interruption of the program, but proceeding to the next track, with assignment of statusflag=1 for the abandoned track.

Can you introduce it if it does not contradict with the exception philosophy somehow? So far I did it for my own, but I think it would be useful to have it in svn.

cheers,
ola

Subject: Re: Fitter Exceptions
Posted by Anonymous Poster on Thu, 04 Jun 2009 12:56:43 GMT

Hi Ola,

hmmm... something is a little strange here. What happens in case of these exceptions changed a little bit since we started to treat the cov[0][0]==0 cases special.

I think the behaviour you decribe to wish to have, you will get, when you switch of the lazy flag in the trunk version. Am I getting it wrong?
Or could you send me and svn diff Kalman.cxx, so I understand what you mean?

Cheers, Christian

Subject: Re: Fitter Exceptions
Posted by Aleksandra Wronska on Thu, 04 Jun 2009 14:05:14 GMT

Dear Christian,

I thought the lazy flag was supposed to relax the conditions of reconstruction? At least that's why I keep it set to 1.

My modifications to Kalman.cxx are the following:

awronska@lxi009:/d/panda02/awronska/pandaroot/genfit$ svn diff Kalman.cxx
Index: Kalman.cxx
===
--- Kalman.cxx (revision 5641)
+++ Kalman.cxx (working copy)

```
@@ -12,8 +12,9 @@
 #include "FitterExceptions.h"

 #define COVEXC "cov_is_zero"
+#define THETACUT "GEANE propagation not possible for p.theta<THETACUT"
+#define LOWMOM "GeaneTrackRep: PROTECT AGAINST LOW MOMENTA"

-
  Kalman::Kalman():_lazy(0),_initialDirection(1),_numIt(3),_blowUpFactor(2
0.),_nullExtrapolation(false){;}

 Kalman::~Kalman(){;}
@@ -129,6 +130,10 @@
              if(e.getExcString()!=std::string(COVEXC)) arep->setStatusFlag(1);
              continue; // go to next rep immediately
            }
+            if(_lazy && ( e.getExcString()==std::string(THETACUT)
||e.getExcString()==std::string(LOWMOM))){
+              arep->setStatusFlag(1);
+              continue; // go to next rep immediately
+            }
          }
        }
    }// end loop over reps
```

cheers,
ola

Hi,

Sebastian introduced that flag (I never would call my code lazy). The flag is supposed to ignore errors and go on with fitting. This only makes sense if errors are recoverable, like the cov[0][0]==0 error. In your cases it does not make so much sense.

So, to conclude: If you want the fit to be aborted and the statusFlag to be set, do not activate the lazy flag, and the code will do what you want.

Cheers, Christian