Subject: Re: PID package
Posted by Bertram Kopf on Wed, 12 Aug 2009 13:36:07 GMT

Hi Soeren,

Jens Soeren Lange wrote
1.) The "set methods".
They are by intention.
It was one of the PandaRoot principles from the beginning.
It is intended to be a protection mechanism, that one class is not able to simply overwrite data members in a another class
(example "the pT of my track is negative, which package did that?")
Or, in other words, the avoid "global variables".

I don't see any protection mechanism by using hundreds of set methods. To provide such set methods speaks completely against the encapsulation of the data members. It is possible to manipulate all private data members from outside by using these set functions. The encapsulation of the data member is one of the basic principles in oo programming. Of course, in some specific cases one can make use of set functions, but for the initialization of an object one should/has to use the constructor.

Jens Soeren Lange wrote
2.) It seems you would like to strictly separate
the reco part (track fitting) on one side
and the PID part (e.g. track extrapolation) on the other side
(and then assign PID into analysis instead of reco).
And your point seems to be (please correct me, if I'm wrong),
that you think that Stefano's approach (i.e. the track extrapolation
and filling PID information directly after the track fitting,
i.e. in the reco part) is in your opinion not a good approach.
So, what you have in mind, is
a.) tracks as reco objects
and
b.) tracks as PID (in analysis) objects,
and in most cases they are probably the same.
Now you are proposing to work with references to the track objects.
However, what then?
-> do you want to modify the tracks then in the PID part?
or, in other words, overwrite the tracks?
If yes, I would strongly vote for "hard copying" all the tracks.
Sure, as you say, these are a lot of data and that was actually
my counter argument when you brought up your proposal
(of the separation of reco and PID)
on the tracking hands-on meeting at GSI in March 09.
. . .

No, the track fitting, the track matching as well as the PID is definitely part of the reconstruction. What I wanted to point out is that one has to disentangle these different things

within the reconstruction. It is of course not a good idea to do all these things within one method.

Concerning the track object: I don't want to have different track objects for one track. One, and only "one" abstract track object. The only point is to get access to all public functionalities by using references to this object, instead of hard copies of specific informations. To avoid manipulations on this object one can define it as a constant reference.

Jens Soeren Lange wrote
3.) ...
It seems that you would like to do a track fitter refit
in the PID part (i.e. then at a quite late stage in analysis).
But we decided on the PID Mini-Workshop at GSI in Sep 2007 that
track fitting will be done for 5 different masses (pi,k,p,e,mu)
anyhow.
...
That's even at a step before the PID.
The PID then tries to make a decision, which particle it was,
but using the momentum from the (already before finished) track fitting.

I don't want to do a re-fitting in the PID part. I totally agree with you that one should do track fitting for all 5 hypotheses (at least for low momenta particles where one expects different results). This, we have also done in the reconstruction for all our Physics Book studies. And, of course, at a later stage of the reconstruction the PID should be done. BTW: The track fitting probability for the different mass hypotheses could also be a helpful PID information.

Jens Soeren Lange wrote
4.) a re-fit is still possible for the TCandidates.
That's for example the vertex constraint fit or the mass constraint fit
as Klaus and Dipak implemented it anyhow (shown e.g. in
the Torino tutorial).

What you meant here is the vertex fitter which makes only use of the obtained track parameters and covariance-matrices obtained by the global track fitter. This has completely nothing to do with the track fitter. For the vertex fitter you combine different tracks (at least two) and try to find a common vertex. What I meant instead is the re-fitting of the tracks in the analysis part. This would make sense when e.g. the alignment, calibration etc. have been improved. In the Physics Book software for example it is possible to switch between a cache and a refit mode where parts of the reconstruction can be re-done.

Cheers,
Bertram.