Subject: concept of stt reco
Posted by Anonymous Poster on Fri, 03 Aug 2007 11:47:04 GMT
View Forum Message <> Reply to Message

Hi Stt-ers,

we have just finished writing digitization for dch and now wanted to start writing reco. Since dch and stt are very similar in use, it's probably more efficient if we reuse your concept (or even partly your code). Could you, in a few sentences, explain
which steps one has to make to go from digis to a track and
which classes are employed?
which dir should we search through (stt1, stt2, stt)?
what kind of objects are needed to feed the global track fitter?

thanks in advance,

piotr

Subject: Re: concept of stt reco
Posted by StefanoSpataro on Tue, 07 Aug 2007 08:50:56 GMT
View Forum Message <> Reply to Message

Hi Piotr,
I read your message about stt reco for drift chamber.
Stt developers are on holydays now, so probably they will not reply you before
In each case the correct directory should be "stt", which is a merge of the Pavia code and what implemented Ralph (we should decide if it is the case to remove stt1 and stt2 from or main compilation, if obsolete).

If you want I can tell you what we did for HADES, for tracking.
We had 4 planes of drift chambers, place two before and two after the magnetic field. Each chamber was composed by 6 wire layers arranged with different angulation, so it is very close to our design (only two chambers missing).
What we did was first to have the position information from each chamber separately, we developed a cluster finder algorythm per chamber based on projections(I can explain you what we did), and when we find the group of wires per chamber we fit the times to have the x and y information on the plane (we did even merging different planes, but this is a bit more complicated).
After we mixed the informations coming from different planes to have momentum. You can find some informations on the presentation I gave to the last pattern recognition meeting in Ferrara:

http://panda-wiki.gsi.de/pub/Computing/PandaPatternRecognitionFerrara2007/spataro-20070509-Ferrara.ppt

I wrote some line of codes for Panda that, from x and y of each plane, you have real pattern recognition and momentum reconstruction with a resolution of about 2%, that you can give as prefit value to the kalman filter code (using the kickplane angorythm, if you read my slides).

To the kalman one could in theory give the single layer information, or the xy of the plane, this has to be decided. In each case with the different layers information it should be not so difficult to solve the ambiguousity left/right, while this is not the case for stt where the situation is a bit more complicated.

Subject: Re: concept of stt reco
Posted by Pablo Genova on Tue, 21 Aug 2007 10:20:07 GMT
View Forum Message <> Reply to Message

Hi Piotr,

as Stefano pointed out - thank you Stefano - we were on holidays, so sorry for the late reply.

The currently up-to-date directory for straws is the stt one: stt1 contains the old Pavia code and stt2 the old Juelich code.

The stt package contains the merged code (stt1+stt2) and we are developing this package, so you should look in it to have the the most recent code.

Concerning the reconstruction in stt, it is divided in different steps and for each step a task is called (see in the macro file macro/stt/runreco.C).

The tasks are:

1) CbmSttFindTracks
2) CbmSttMatchTracks
3) CbmSttFitTracks

Tasks 1 and 2 come (almost) directly from stt2, and they perform an ideal pattern recognition: the MC thruth is used to assign points/hits to the tracks (in CbmSttFindTracks task, which calls CbmSttTrackFinder) and a matching between reconstructed tracks and MC tracks is performed (in CbmSttMatchTracks, calling CbmSttTrackMatch).
Then the track fitting is performed (step 3) in the CbmSttFitTracks task, which calls CbmSttHelixTrackFitter.

The real track fitting code is contained in CbmSttHelixTrackFitter: it performs a fit with a helix (as the name suggests), separating the problem in a circle fit in the plane orthogonal to the magnetic field and a straight line fit in z - track length plane.

In brief the fitting procedure is the following - for a more detailed explanation or visualization we can send you our presentations:

----- for the transverse momentum:

1 - pre-prefit: using the conformal mapping technique we fit the centers of the firing tubes to have a preliminary hypothesis for the track parameters.
2 - minuit: then we call Minuit to re-do the fit on the center of the tubes, using the results of the pre-prefit to initialize the parameters.
3 - intersection finder: we find the points on the drift circles which are the nearest to the

reconstructed circle (obtained at point 2.). To do this we simply join the centres of the reconstructed circle and the drift circles and find the intersections between this segments and the drift circles (we can send you a plot to show this).
[By 'drift circle' we mean the circle with radius = drift radius measured by the straw, centre = wire centre]
4 - refit: these "new" points are fitted again using the conformal mapping technique.

By this procedure the parameters
R = radius of curvature
d0 = distance of the point of closest approach to the vertex
phi0 = corresponding angle
are found.

------ for the last two parameters:

z0 = z of the point of closest approach
tan(lambda) = tangent of lambda (lambda = dip angle of the track)
they come from the following procedure:

1 - zfinder: from the projection of the skewed tubes in the transverse plane and from the prefit in the transverse plane we infer the z coordinate of the hits. We use the hough transform to decide between the two hypothesis for each point (see the Dubna presentation by Pablo
http://panda-wiki.gsi.de/cgi-bin/view/Computing/PresentationsCMJuly2007
)
2 - zfit: a linear fit is performed in the z - track length plane.

On the issue of the "global tracking" (the last question) we do not know so much, since for the moment we mainly worked with the STT only.

We are currently working on the kalman filter method, for which the previously described fit is just the starting point.

Feel free to reuse our code as you prefer and to ask any question, Andrea and Pablo will attend the next Collaboration Meeting so we can also discuss directly.

Best regards, Pablo & Lia