
Subject: Re: alternative to LHE tracking

Posted by [Radoslaw Karabowicz](#) on Fri, 17 Dec 2010 14:49:52 GMT

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

Dear Stefano and All,

I am not sure how to describe the tracking shortly. The track model is a helix, that is a circle in x-y plane plus some pitch along the z-axis. Since this assumes the constant magnetic field parallel to z, the tracking will not work in the Forward Detector, and is not doing very well in the GEM region. Of course some future extensions are possible.

The main part of the program is the loop over hits. The hits in one event are mixed so that there is no correlation possible to MC.

Inside the loop the hit is matched to already found tracks. If it doesn't match to any track, the hit is being combined with previously unused hits to form tracks. If no such combination is possible, it is put in unused hit array.

The hit to hit matching assumes the track is primary, that is one of the 3 needed points to construct a circle in x-y plane is set to origin. The other two points (or circles for STT) are set to (x,y) parameters of the hits. In case of hits in MVD, TPC or GEM, where there is no ambiguity, only one circle is fit to 2 hits. With one STT hit there are two circles found (one including, one excluding STT hit, both tangent to it). With two STT hits there are 4 combinations. All the combinations are stored in the track.

The hit matches to already found track, if its distance to any of the track circles in x-y plane is smaller than some parameter (for now around 1mm, but should depend on hit resolution). If the hit matches to the track, then circles with all other track hits are formed. If other track hits are close enough to the circle, then the circle parameters are stored in the track.

After the track contains more than 3 hits and a lot of possible circle parameters from various hit combinations I try to see if there exists one common circle that would fit all the hits. If such mean circle exists, the criterium to attach next hits will be their proximity to this mean circle.

In fact only after this mean circle is found do I try to attach SKEWED STT hits to the track. When I attach them no new information about the circle in xy plane is obtained, only the pure information about z position of the place where skewed stt tube is tangent to the track circle... If there are enough hits in the track (presently more than 3) that carry the z information then I try to obtain the pitch of the track's helix. Again by checking if there is some mean helix pitch from which all the track hits are not too far.

After the main hit loop is finished, I loop over found tracks and get rid of the tracks that had no mean circle nor mean pitch found. They are usually short tracks formed from hits belonging to different MC tracks. This getting rid of tracks consist of moving track hits into the unused hits array, and removing the track from the list.

The last step is to attach the hits from the unused hits array to existing tracks.

The algorithm may be summarized by following simplified C++ code:

```
noHits = fHitArray->GetEntriesFast();  
noTracks = 0;  
noUnusedHits = 0;
```

```

for ( ihit = 0 ; ihit < nofHits ; ihit++ ) {
    Bool_t hitUsed = kFALSE;
    for ( itr = 0 ; itr < nofTracks ; itr++ )
        if ( MatchHitToTrack(ihit,itr) ) hitUsed = kTRUE;
    if ( hitUsed ) continue;
    for ( iuh = 0 ; iuh < nofUnusedHits ; iuh++ )
        if ( HitMatchesToHit(ihit,iuh) ) hitUsed = kTRUE;
    if ( hitUsed ) continue;
    AddHitToUnusedHits(ihit)
}
RemoveShortTracks();
for ( iuh = 0 ; iuh < nofUnusedHits ; iuh++ )
    for ( itr = 0 ; itr < nofTracks ; itr++ )
        MatchHitToTrack(iuh,itr);

```

WriteTracks();

The tracks are written to the output tree as PndTrack (BarrelTrack) and PndTrackCand (BarrelTrackCand). The last difficulty is to retrieve the momentum information of the track from the helix parameters. For the time, there is some hardcoded values used to make the transformation.

If the explanation is not satisfactory, messy or unclear, please let me know.

yours,
radek
