The Efficiency of Track Reconstruction Using the Vertex Detector as the Primary Tracking Device

N.B.Sinev
University of Oregon, Eugene, OR 97403-1274 USA

Track reconstruction, using the Vertex Detector as a primary tracking device, has been developed for the Silicon Detector (SiD) design. Some preliminary results on the efficiency of such a reconstruction are presented.

1. INTRODUCTION

At the time the Silicon Detector (SiD) for the International Linear Collider (ILC) was suggested, it was obvious that the reconstruction of charged particle tracks in such a detector will be not a trivial task. The problem for the reconstruction lies in the small number of the layers of the position-sensitive devices (silicon microstrip detectors), and in poor spatial resolution of such devices in the beam direction (at best we can implement breaking the strips into segments of about 10 cm in length, this would correspond to about 3 cm resolution in the Z-coordinate). It was anticipated that the main role in the reconstruction of the charged tracks will be played by the vertex detector, while the microstrip devices will be used for achieving a better momentum resolution. However, large density of the charge particles close to Interaction Point (IP) and, again, the small number of layers (5) in the vertex detector, may cause significant deterioration of the tracking algorithm performance. It may reduce the efficiency of the track reconstruction and generate fake tracks - assembling of the charge particle hits belonging to different particles into one "track". To test the performance of the track reconstruction algorithm based on the vertex detector information, a set of software modules was created within the hep.lcd software package, used together with the Java Analysis Studio for evaluating future ILC detector performance. Here we are going to report some initial results characterizing the performance of such tracking code.

2. SOFTWARE

2.1. Algorithm

We will consider our tracking detector to be inside a magnetic field. The magnetic field vector is directed along the beam axis, which will be the Z axis of our coordinate system. As it is well known, the track of a charge particle in such a magnetic field will be a helix. Its projection on the XY plane is a circle with radius depending on the magnitude of the XY projection of the charged particle momentum (so called transverse momentum, referred to as Pt below), while its step along the Z axis depends on the angle between the total and the longitudinal momentum (the projection of the particle momentum to the Z axis). We will call this angle Λ in our discussion. We will also use the angle Θ which is the angle between the full and the transverse momentum of the particle. The reconstruction algorithm may be described as follows:

1. A combination of 3 layers of the vertex detector is chosen as a pattern recognition triplet. The code will repeat reconstruction attempts with different combinations to eliminate the possibility of loosing tracks because of one of the layers in the triplet missing a signal (layer inefficiency). One point from each layer in the triplet is selected and a circle in the XY plane connecting projections of these 3 points into this plane is calculated. This will be repeated for all possible combinations of the points from 3 layers. To reduce waste of the time on such calculations, a preliminary check on the Z coordinates of these 3 points is made, requiring that the Z position of the track hit point in the middle

layer should be between the Z coordinates of the inner and outer layers of the triplet. After circle radius and center are found, further selection of the possible track candidates is done based on the Z position of the middle point, which now can be calculated more precisely.

2. The parameters of the found circle are checked for satisfying certain cuts on possible track parameters. For example, the radius of the circle should exceed a predefined minimum and the circle should pass close enough to the interaction point. If it satisfies these conditions, the angle Λ is calculated based on the Z position of the points in the triplet. This completely defines a track candidate.

3. The track candidate is extrapolated to the other layers of the vertex detector, and a search for the charged particle hits in its proximity is performed. Only the closest hit in each layer is attached to the track, and only if it is within the expected distance, defined by the detector resolution and a possible effect of multiple scattering. The required number of such attached hits for continuing the consideration of a given track as a candidate may be changed. However, accepting tracks with no additional hits in the vertex detector leads to an unacceptably high number of fake tracks.

4. The track candidate is extrapolated further into the microstrip detector layers. Again, hits close to the track are found, but now we are either ignoring the Z-coordinates of the hits (for the microstrip detector design without strip segmentation - so called long strips), or using the length of the strip segments to attach all hits that are close in the XY plane and within the distance of a strip length in Z to the track. Before extrapolating the track beyond the innermost layer of the microstrip detectors, we recalculate the circle parameters, using the attached hit in the innermost layer. This reduces track position errors in the following layers. However, the angle Λ is not recalculated, because the vertex detector provides an error for this parameter that is much smaller than that of the microstrip layers. The number of attached hits is counted and this number is combined with the number of hits attached in the layers of the vertex detector. A track is accepted if this total number of attached hits exceeds a threshold value, which may be set to 5 or 6, or more. Setting it less than five does not make much sense, because too many fake tracks will appear. In fact, 5 is also a very soft condition, as most real tracks will have hits in all 5 layers of the vertex detector and will satisfy such a threshold cut without counting a single hit from microstrip detectors attached.

5. The track parameters are recalculated once more, now using the largest lever arm available for the given track, based on the attached hits in the outermost layer of the microstrip detector and the innermost layer of the vertex detector. Again, only the Pt of the track is recalculated, the angle Λ is not.

6. Finally, once more the track is traced through all layers of the tracking detector and hits are assigned to the track. At this point a check is made to see if this track shares hits with any of the other tracks reconstructed earlier. One shared hit is allowed. Usually it is the generation point which may be common for more than one track. If the track shares more than one hit with another track, it is declared a "duplicate", and a comparison is made with another track. The one which has larger number of assigned hits or better χ^2 is accepted, the other one is discarded.

2.2. Reconstruction speed and limitations imposed by it

The main problem for the algorithm described above is the amount of calculations needed. In general, the algorithm is designed to try all combinations of hits in 3 layers of detector. If each layer has 100 hits, the number of such combinations is 10^6. For clean events without hits from backgrounds and electronics noise, it is a reasonable estimation and it has reasonable processing time of about 1 second per event. But because the number of combinations is the 3rd power of the density of hits, increasing the density of hits by a factor of ten leads to a reconstruction time of about 1000 seconds/event, which is not comfortable. And if the vertex detector is made of CCDs, which are integrating background hits through many bunch crossing, the amount of hits per layer may became even larger. Because CCDs have a huge number of pixels (10^7 per layer), the number of hits in the range of 10000/layer does not create a large occupancy factor (still 0.1%), so such a number of hits should not cause trouble for the identification of the tracks, but the reconstruction time will become absolutely intolerable.

There may be a solutions to this problem, for example by using the microstrip tracker for a preliminary estimation of possible locations of the track hits in the layers of the vertex detector. However, it will require the track to

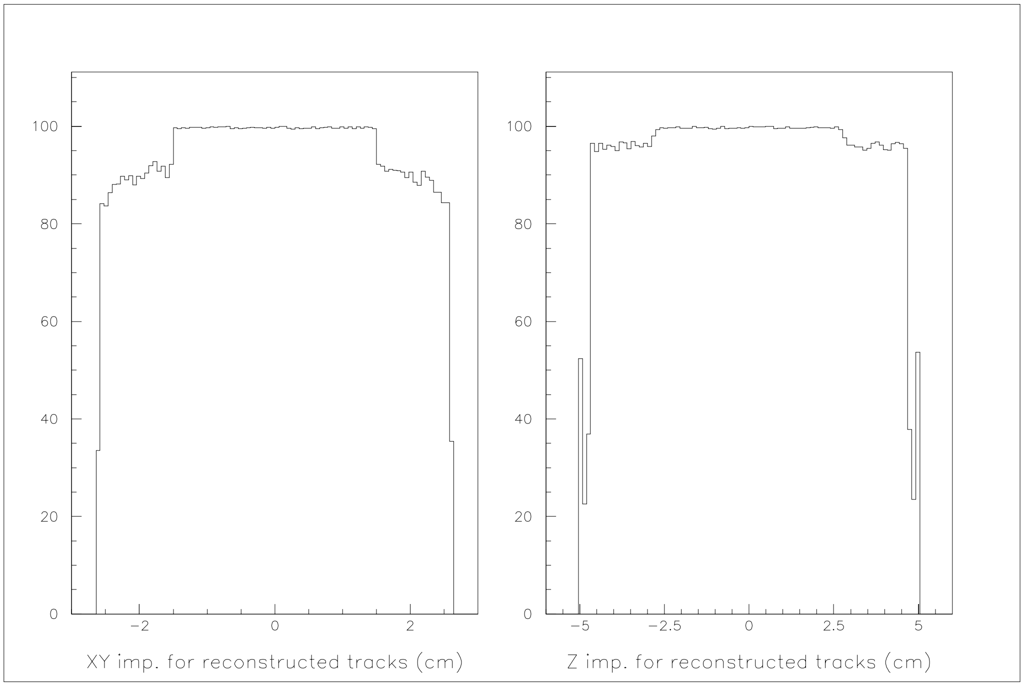

Figure 1: Reconstruction efficiency as function of track impact parameters D and Z.

traverse at least 3 layers of the microstrip tracker. This will raise the lowest transverse momentum of detectable tracks to about 0.5 GeV and this is not desirable. Another solution is to use the track finding algorithm based on a Hugh transformation. Such an algorithm has not cubic, but quadratic dependence of the reconstruction time on the number of hits, and would become faster than the algorithm described above if the number of hits exceed 1000/layer. We have reported on a test of a Hugh transformation based algorithm earlier. However, so far it was not developed into fully operational software, so we are using the more traditional algorithm here. To address reconstruction speed, some limitations on the reconstructed track parameters were made:

We will consider only tracks originating close to the IP. This leads to substantial reduction of the number of 3 point combinations, because after selecting the point in the outer most layer of triplet we do not need to consider all points in the two other layers of triplets but only points having similar values of a deep angle of the vector connecting the point to the IP.

The definition of "close to the IP" may vary. We can set it to a few cm, which allows reconstruction of almost all B decays. Of course Λ or K_s would generally not be reconstructed, but they would not be reconstructed even without such limitation, because we need the decay point to be inside the second inner most layer of vertex detector. The next section will show how far from the IP tracks can originate to be reconstructable.

3. RESULTS

3.1. Reconstruction of the tracks originated far from IP

As was mentioned above, we may need to limit the track origin point to speed up the reconstruction. In addition, as we are using the vertex detector as a tracking device we require that at least 4 layers of the vertex detector have hits from the given track in order for the track to be reconstructed. This leads to the inability to reconstruct tracks originating outside of the second barrel of the vertex detector. In the Z direction, the track origin point should be inside the span of the vertex detector in Z. Reconstruction was tested by generating single tracks originating far from

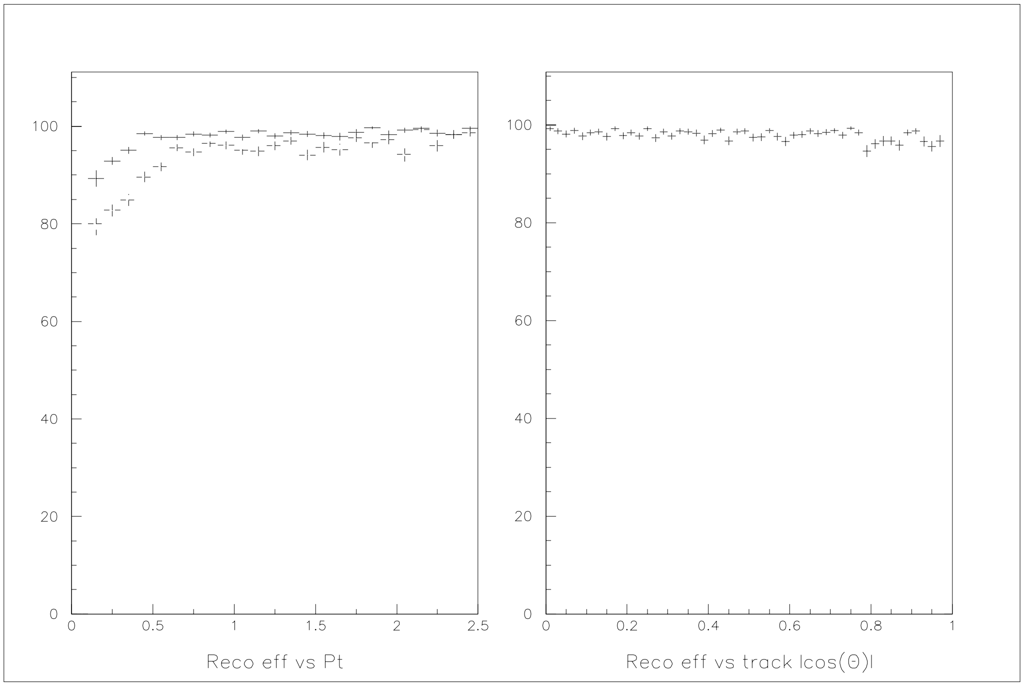

Figure 2: Reconstruction efficiency as a function of the track Pt and cos(Θ).

the IP and reconstructing them. We could not use the generated physics events for such test, as there are too few tracks originating far from the IP in such events.

For all our tests we used the Silicon Detector geometry version of February 2005. This is the so-called "short" detector, where all barrels of the vertex detector have equal length (12.5 cm total, spanning in Z from -6.25 cm to + 6.25 cm), and there are 4 vertex discs for the reconstruction of forward tracks. We also used the "Full CCD Simulation" package [1] to simulate a realistic response of the vertex detector. The efficiency of single track reconstruction as a function of the track origin position is presented in figure 1.

We can see that the reconstruction of the tracks with an origin outside first layer of the vertex detector is possible, but the reconstruction efficiency is noticeably lower than for tracks with the origin inside the first layer. This is understandable, as missing the signal in any of remaining 4 layers immediately makes reconstruction impossible. As a full CCD simulation gives an average layer inefficiency of about 1%, we could expect a drop in reconstruction efficiency for tracks traversing only 4 layers by about 4%. The fact that the observed reduction is larger than 4% is probably due to the additional effect of hits that are too far from the expected position (because of measurement errors or multiple scattering). The behavior of efficiency as function of Z is more puzzling. We set the reconstruction cut on Z to 5 cm. This is well within the vertex detector. However, we see an unexpected (though not too significant) drop in efficiency for Z exceeding 2.8 cm, and absolutely puzzling dips at 4.8 cm. This requires further investigation

3.2. Reconstruction efficiency as a function of the transverse momentum and the deep angle

Now we will use generated physics events (in particular we used TTbar events for our tests). Each event contains many tracks. Some tracks are close to each other. This leads sometimes to the reconstruction of a fake track - a track made of hits from two or more different real tracks. An efficient way to reduce the number of fake tracks is to increase the threshold for the number of hits attached to a track. A first (naive) expectation would be that such an increase will lead to a higher Pt threshold for the reconstruction, as tracks with low Pt have a curvature radius that is too small to reach the outer layers of the tracker barrels. However, in reality, the inability to reach the outer

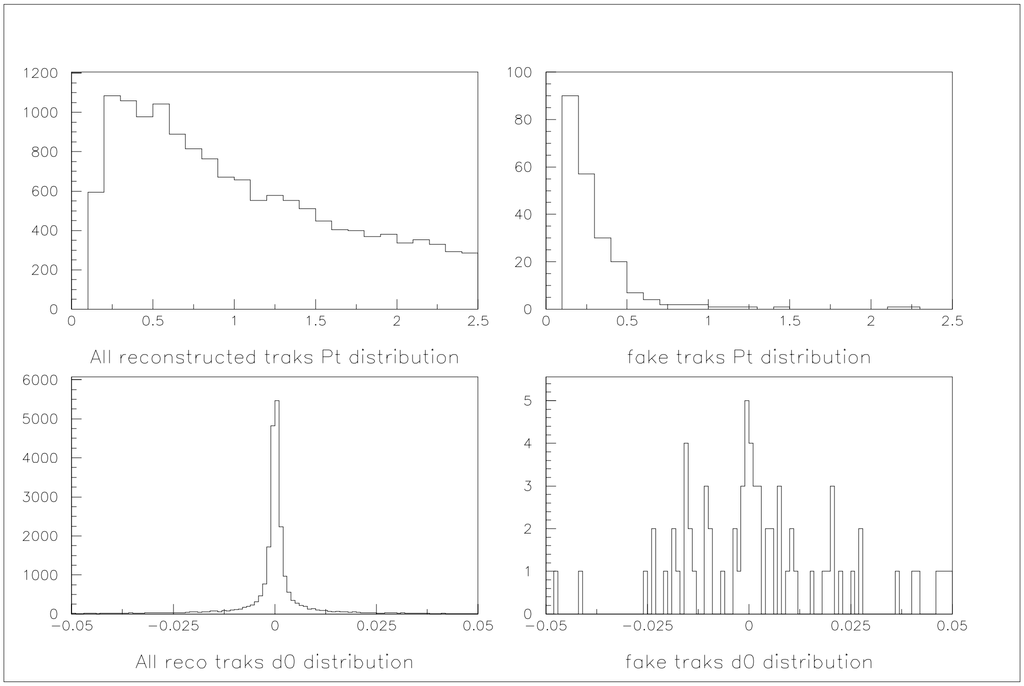

Figure 3: Pt and impact parameter d0 distributions for good and "fake" tracks.

barrels should not make a track non-reconstructable even with a high threshold of the attached hits. Help comes from the endcap trackers. As their inner radius is small enough, essentially any track with a Pt large enough to exit the vertex detector will hit the endcap tracker if it did not hit the inner wall of the calorimeter. So, in principle, we are guaranteed a minimum number of 5 tracker hits for any track: if it hits calorimeter, it passes through all barrels and if it has low enough Pt not to hit calorimeter, it will hit 10 layers of the endcap tracker.

In real life, of course, the picture is not so nice. Tracks making many turns on the way to endcap will hit a lot of material and will scatter so much that their hits will appear too far from the expected position to be successfully attached. All this can be seen from figure 2.

The reconstruction efficiency as a function of Pt is presented for 2 different cuts on the required number of hits attached to the track. The higher curve has this requirement set to 6, while for the lower one it is set to 8. Tracks with Pt less than 0.54 GeV cannot reach the third barrel of tracker. However, we can see that although the reconstruction efficiency for such tracks declines, it still does not have a step at this point.

The reconstruction efficiency vs. $\| \cos\Theta) \|$ should be uniform up to 0.97. We can see, however, a small but noticeable drop (by about 2%) for $\| \cos(\Theta) \| > 0.8$. This is where reconstructing involves vertex detector endcap discs. Though in principle discs should not be worse than barrels, as they have same space resolution and provide 3-dimentional space points on the track, the drop in efficiency is probably due to much larger amount of material traversed by the tracks in this area (tracks are passing through the barrel endplates).

3.3. Generation of "fake" or "ghost" tracks

Even without backgrounds or noise hits, there is still the possibility that the reconstruction software will pick up hits from different closely positioned tracks and combine them into one "fake" track. In the current study we are calling the reconstruction track "fake" if it meets 2 criteria:

1. The monte-carlo parent particle assigned to this track is also assigned to another track.

2. Track "purity" (defined as a fraction of hits belonging to the same monte-carlo parent of all hits assigned to the track) is less than 0.5

The number of such fake tracks depends on the cuts in the reconstruction algorithm. With wide open cuts (low threshold on Pt, low threshold on total number of hits assigned to track, no cuts on the position of track origin) the number of fake tracks can reach 2.5% of all reconstructed tracks. Setting the Pt threshold closer to the lower limit on Pt which we want for the reconstructed track (reducing margin in Pt) and limiting the origin position to within 0.5 cm from the IP can reduce the number of fake tracks to below 1%. The distributions of Pt and d0 for fake tracks in comparison with real tracks are shown in figure 3

We can see that fake tracks have a very soft spectrum in Pt, so their contribution to the energy balance of the event is small. Nevertheless, we need to remember that they are present. This topic requires further investigation.

3.4. Tracking performance in the presence of backgrounds.

This has been studied in detail for the warm technology, and has been reported on at the Victoria meeting [2]. So far, we did not repeat this study for the cold technology, as background files for such technology have not been generated yet. This is in our plans, and hopefully can be fulfilled by Snowmass.

Acknowledgments

The author wishes to thank Mike Ronan, who created the original reconstruction framework for use in the Java Analysis Studio. We have used his general approach and many of his java classes. We are also grateful to Norman Graf for his support and useful discussions.

References

[1] N.Sinev http://agenda.cern.ch/askArchive.php?base=agenda&categ=a04172&id=04172s60t1/transparencies
[2] N.Sinev http://www.linearcollider.ca:8080/lc/vic04/abstracts/detector/simul/nick_sinev.ppt