STC Drift-Time Relation

HEP Group PG Computing Exercise
R. Beuselinck
October 2020

1 Introduction

Programming is about more than knowing where the curly brackets and semi-colons go; that's the easy part. You will need to learn how to understand data from your experiments, how to develop solutions to new problems and how to design and test code to implement your ideas. This exercise is designed to test those skills by developing a solution to a typical high energy physics calibration problem.

The Aleph Inner Tracking Chamber (ITC) can be seen on display next to the lifts on level 5. This drift chamber was designed and built within the group and ran superbly for over 10 years. Before this chamber was built a small prototype test chamber (STC) was built to investigate the drift-time relation of the proposed cell configuration and gas mixtures. This exercise involves determining the drift relation of a simulated dataset corresponding to a simplified model of the STC.

2 STC Description

The detector simulated here is a simplified version of the actual STC. It is an eight layer drift chamber with eight cells per layer. The layers are arranged in parallel planes. Each alternate layer is offset in y by half a cell width. The cell-to-cell spacing is 1cm in y. The layer-to-layer spacing is 1cm in x. Each sense wire is surrounded by field wires arranged to create a circularly symmetric, linear drift-time relation; the drift time is proportional to the closest distance of the sense wire from the track, regardless of direction. This is illustrated in figure 1.

The STC is placed in a test beam of charged particles. A signal is read out from the closest sense wire to the tracks on each layer. The time recorded corresponds to the earliest arriving ionisation along the tracks, from the closest point of approach to the sense wires. The signal arrival times are recorded by 10 bit TDCs with 0.5ns resolution. Thus the maximum TDC count is 1023 and a count of zero represents a drift time between 0.0 and 0.5ns.

The beam intensity is low and the STC is triggered on the passage of one particle per event, leading to a single wire hit on each layer. Using the detector geometry and the TDC counts alone it is possible to determine an estimate of the average drift velocity and to reconstruct the equation of the straight line for each charged track.

3 Raw Data Format

The simulated raw data for this problem is provided as packed binary data as is typical for a real data acquisition system. The data for each recorded hit consists of a TDC count and a wire address packed into 16 bits. There are 8 hits per event, so one event occupies exactly 16 bytes.

The layers are numbered from 0 to 7 by increasing x. The sense wires within each layer are numbered from 0 to 7 by increasing y. The layer number is packed into the lowest order 3 bits of a 16 bit datum. The wire number within the layer is packed into the next 3 bits. The 10 bit TDC count is stored in the remaining bits.

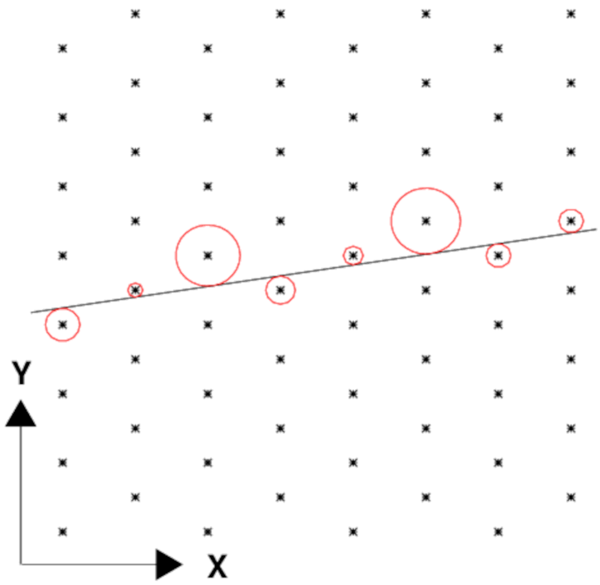

Figure 1: End-on view of the STC detector. The sense wire positions are represented by the asterisks. The red circles illustrate the drift distances to the charged track shown traversing the detector.

4 Your Task

Your assignment is to write a C++ program to analyse the simulated raw data described above. The program should run on the group's general linux machines and compile and run without errors using the standard compiler, g++. The final version of your working code should be emailed to me, along with any plots generated, the numerical results obtained and a suitable description of your method — preferably as a gzipped tar file. Please do not send compiled code or root files; I will expect to rerun your code and regnerate these files myself. Do not include the raw data in your submission; I do not want any 16MB emails!

I recommend you use ROOT to perform any statistical analysis of your results and produce histogrammes as appropriate, but your main analysis code to determine the drift relation must be able to be compiled as a standalone C++ program. Your determination of the unknown parameters for each event should rely only on code you write yourself, i.e. you cannot use any fitting or minimisation packages for this.

Your analysis should provide the following numerical results, preferably with estimates of the errors:

- the average drift velocity in microns per nano-second.
- the mean inclination of the particle beam from the horizontal in degrees.
- the one sigma spread of the beam angle in degrees.

To complete this task successfully you will need to figure out how to read and unpack the binary data. You will have to figure out a suitable algorithm to determine the drift relation from the raw TDC data. A few of the events will be problematic, in that they will not be able to provide a useful solution due to ambiguous geometrical configurations. You will need to be able to reject such problem events.

In marking your solution (at least) the following criteria will be used:

- Does it work? I should be able to compile and run your code "out of the box" without having to modify anything.
- Does it give the right answers for the datasets provided?
- Is your solution method appropriate to the problem? Your algorithm should work well enough without having to be hideously complicated or time consuming. (My code to solve this exercise runs in about 5 seconds on my ancient desktop PC.)
- Your code should not require excessive amounts of memory. (Hint: do not read the whole file in one go; do not store all the events in memory.)
- Is the code well structured and adequately documented? It should be clearly laid out and easy to follow. The algorithm should be clearly described somewhere.

The data files you should use for this exercise can be copied from
`/home/hep/bslnckr/onetrack.raw` and `/home/hep/bslnckr/manytracks.raw`.

The first file contains a single track only. You should find the drift velocity and track angle for this single track and report that in your results. The second file contains 1,000,000 tracks with a *different* drift velocity and a distribution of track angles around some mean value. You should produce histogrammes of the distribution of drift velocity and track angles you find for this sample.