CDF/ANAL/SEC_VTX/CDFR/1924

Fitting the beam position with the SVX

Hans Wenzel

Fermilab

Abstract

To know precisely the position of the primary vertex of an event is quite important for a number of measurements (e. g. b-lifetime, b-tagging, b-cross section....). This note describes which algorithms are used to fit the beam position using SVX track information and it describes how to access the information stored in the CDF database. The track/vertex selection criteria are listed. Finally plots are presented which: show the position of the beam in the first months of data taking, indicate that the calculated values and errors are correct and show that the global alignment between the two barrels of the SVX is quite good and stable.

1 Access to the information and general remarks

The beam parameters are stored in the CDF conditions database. They can be easily accessed by calling the subroutine C$SVX:SVGEBP.CDF. The information is then stored in the common block SVXBPO. A printout of the include file C$INC:SVXBPO.INC is appended to this note. Since the beam position is calculated on a run by run bases this routine should be called in the run initialization part of your AC module.

The files needed to do the beamline fits are produced as part of VALPLOTS to avoid spinning through the tapes again. That means the positions are not available until valplots has been executed on these tapes.

Before the package became part of VALPLOTS one had to go through all the tapes. In some cases the production output was tracked with the wrong alignment constants in this cases the SVX tracking was redone with the proper set of alignment constants.

Part of the data has been tracked using the wrong t_0's to do the CTC fit. This shouldn't affect the measurement of the beam position using the SVX. But it will be checked as soon as a data sample is available retracked with the right CTC calibration constants which can be compared to the results we have now.

Peter Berge points out correctly that now we have two different beampositions in the database. One obtained with the CTC in stage0 and then the SVX fit which is much more precise. Up to now the routines (TRKSVC, CTCSVM I think) doing a beam constrained fit access the CTC database instead of the SVX value.

So these routines should be changed that they access the SVX information or we should decide to have just one database for the beamline which contains the best value available.

2 The algorithms

Different algorithms have been tried. At the moment the track based algorithm using the D-φ correlation is the one used to get the values for the database. The Algorithm based on the primary vertex as estimated by the routine VXGTPR is run in parallel to have a comparison. For the combined fit (combining the two SVX barrels) both algorithms give exactly the same results. The D-φ algorithm needs less events to give a stable result and doesn't require vertexing so it was chosen as the 'standard' fit.

2.1 D-φ correlation

This algorithm is based on tracks, that means every selected track gives one point in the fit. For this algorithm no fitting of the primary vertex is necessary and it requires less data than the fit based on the fitted primary vertices.

To first order the impact parameter D for tracks coming from the primary vertex can be parametrized in the following way:

$$D(\varphi_0, Z_0) = x_0 \cdot \sin\varphi_0 + ax \cdot \sin\varphi_0 \cdot Z_0 - y_0 \cdot \cos\varphi_0 - ay \cdot \cos\varphi_0 \cdot Z_0 \tag{1}$$

where

x_0,y_0	:	position of the beam at z = 0
ax, ay	:	x and y slope of the beam

So the χ^2 to be minimized is:

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{D_i - (x_0 \cdot \sin\varphi_0 + ax \cdot \sin\varphi_0 \cdot Z_0 - y_0 \cdot \cos\varphi_0 - ay \cdot \cos\varphi_0 \cdot Z_0)}{\sigma_i} \right)^2 \tag{2}$$

or with $\vec{x} = (x_0, y_0, ax, ay)$ and $\vec{g} = (\sin\varphi_0, -\cos\varphi_0, \sin\varphi_0 \cdot Z_0, -\cos\varphi_0 \cdot Z_0)$.

$$\chi^2 = \sum_{i=1}^{N} \left(\frac{D_i - \vec{x} \cdot \vec{g})}{\sigma_i} \right)^2 \tag{3}$$

where $\sigma_i^2 = \sigma_D^2 + 2 * \sigma_{Beam}^2$. For the sigma of the beam σ_{Beam} a value of $50\mu m$ is assumed.
The solution for $\vec{x}$ is then:

$$\vec{x} = V \cdot \vec{sg} \tag{4}$$

Where the components of the inverse of the 4 by 4 matrix V are defined in the following way:

$$V_{lm}^{-1} = \sum_{i=1}^{N} \frac{g_l \cdot g_m}{\sigma_i^2} \tag{5}$$

(l,m = 1,4)

and the components of the vector $\vec{sg}$ are:

$$sg_l = \sum_{i=1}^{N} \frac{g_l \cdot D_i}{\sigma_i^2} \tag{6}$$

(l=1,4)

Table 1 summarizes the requirements for the tracks used in the fit. In addition tracks crossing from one barrel to the other were rejected.

The fitting was done iteratively that means after every iteration the impact parameter cut with respect to the fitted beam position was tightened until 60% of the originally selected tracks survive.

P_t of each track	:	>1.0 GeV
Distance from nominal beam position	:	<0.2 cm
Parametrization	:	3D
Nr. of hits in the SVX	:	4

Table 1: Track selection cuts

2.2 Using the primary vertex of the event

This algorithm is an event based algorithm. For each event the routine VXGTPR was called to fit the primary vertex. VXGTPR gives the (X,Y,Z) position of the vertex and the covariance matrix. Then the sigma of the beam is added to the relevant terms of the covariance matrix before doing a line fit through the points (x and z in cm). The line fit is a standard technique and needs no detailed description here. Table 2 summarizes the track and vertex selection cuts for vertices used in the fit.

Minimum Nr. of 3D tracks for the Vertex fit	:	3
χ^2 of the vertex fit	:	< 20
P_t of each track	:	> 0.6 GeV
Nr. of hits/track in the SVX	:	4

Table 2: Track/vertex selection cuts

In addition it was required that all tracks were completely in one barrel. So no tracks crossing from one barrel to the other were used in the primary vertex fit. The reason was that we wanted to check the global barrel-to-barrel alignment comparing the fitted beam position in the two barrels. The third plot of figure 1 shows the distribution of the primary vertex x versus z and a fit through the points. The upper two plots show the variation of the primary vertices with respect to the calculated beam position. We see that the beam profile is pretty gaussian and circular. The sigma of the gaussian is consistent with a sigma of the beam of 36 μm convoluted with the error of the primary vertex fit.

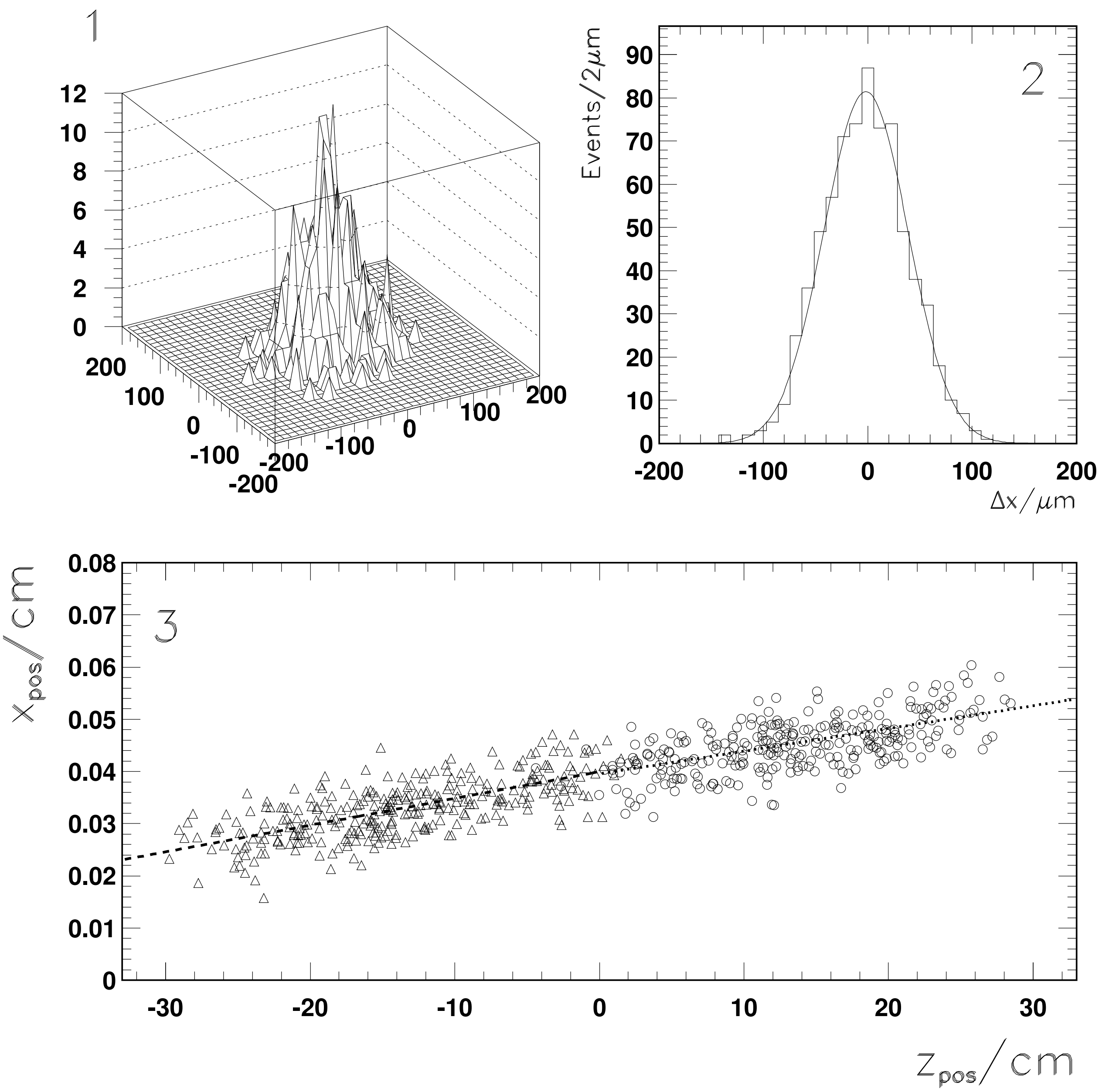

Figure 1: Position of primary vertex (x(cm)/z(cm) and beam profile

3 Checks of the global alignment

To check the barrel-to-barrel alignment the fit was done for each barrel separately. Figures 2 and 3 show the difference between the fitted x0 and y0 in the different modules of the SVX. We see the difference between the barrels is quite stable over the analyzed period. The mean value of the difference is about 5 μm in x and 3 μm in y. The two right plots show the difference divided by the convoluted error. If the errors are calculated correctly one would expect a gaussian distribution with a sigma of 1. This is the case within 10%.

Figures 4 and 5 show the differences in slope between the two barrels. Again the difference is quite stable in time. The mean value is in the order of 0.3 μm /cm. Also for the slopes the errors seem to be calculated correctly within 25%.

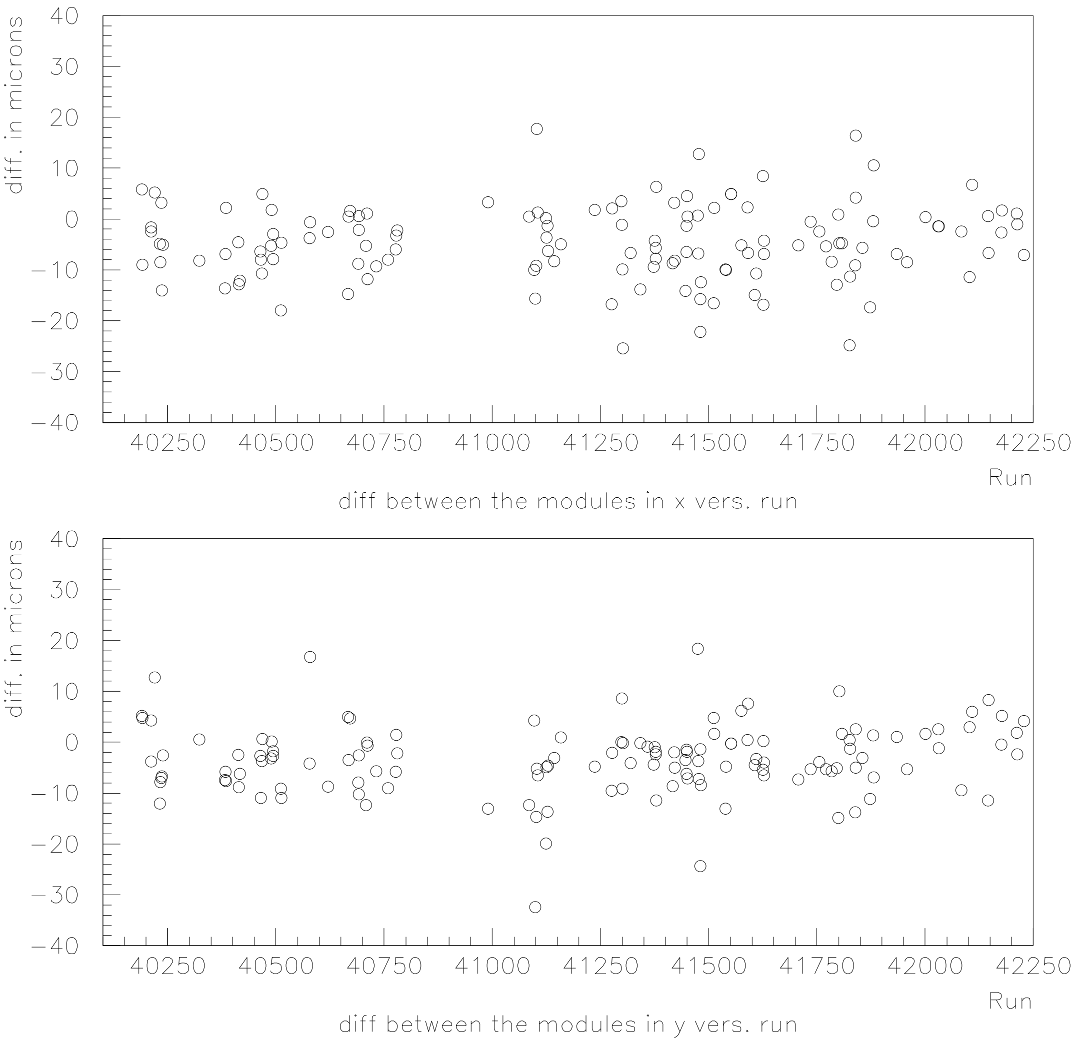

Figure 2: Diff. in x0 and y0 between the two SVX barrels vers. run

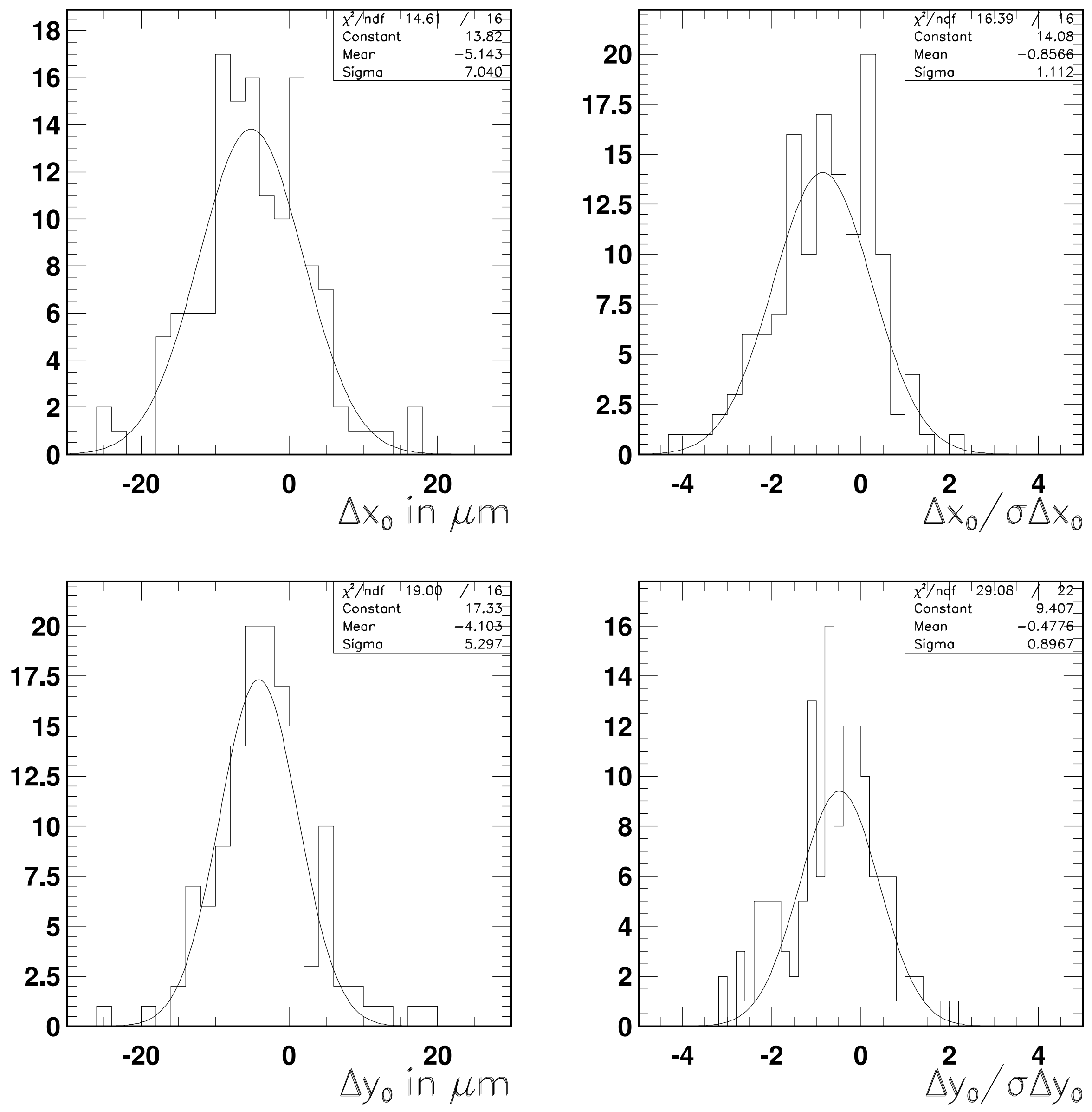

Figure 3: Difference and normalized difference in x0 y0

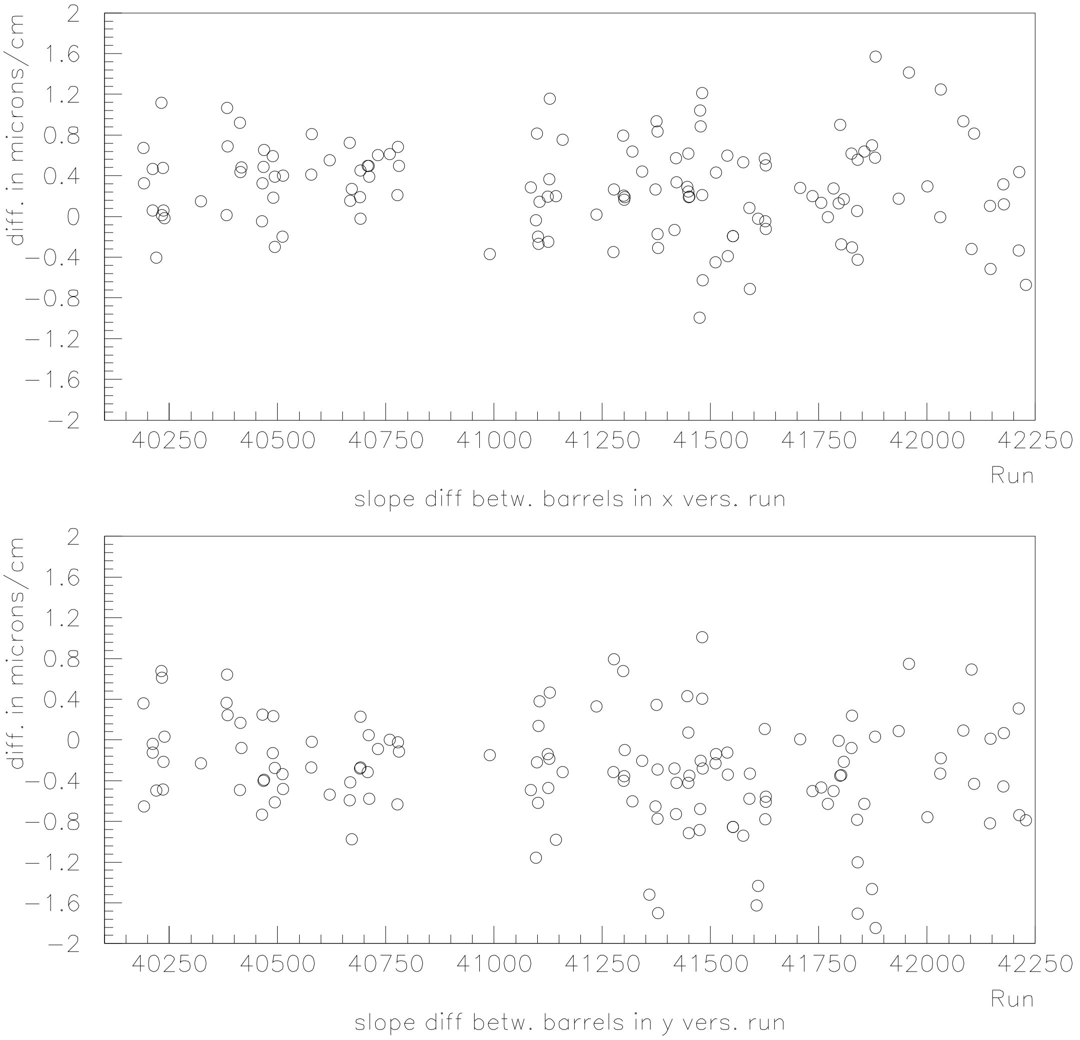

Figure 4: Diff. in x/y slope between the two SVX barrels vers. run

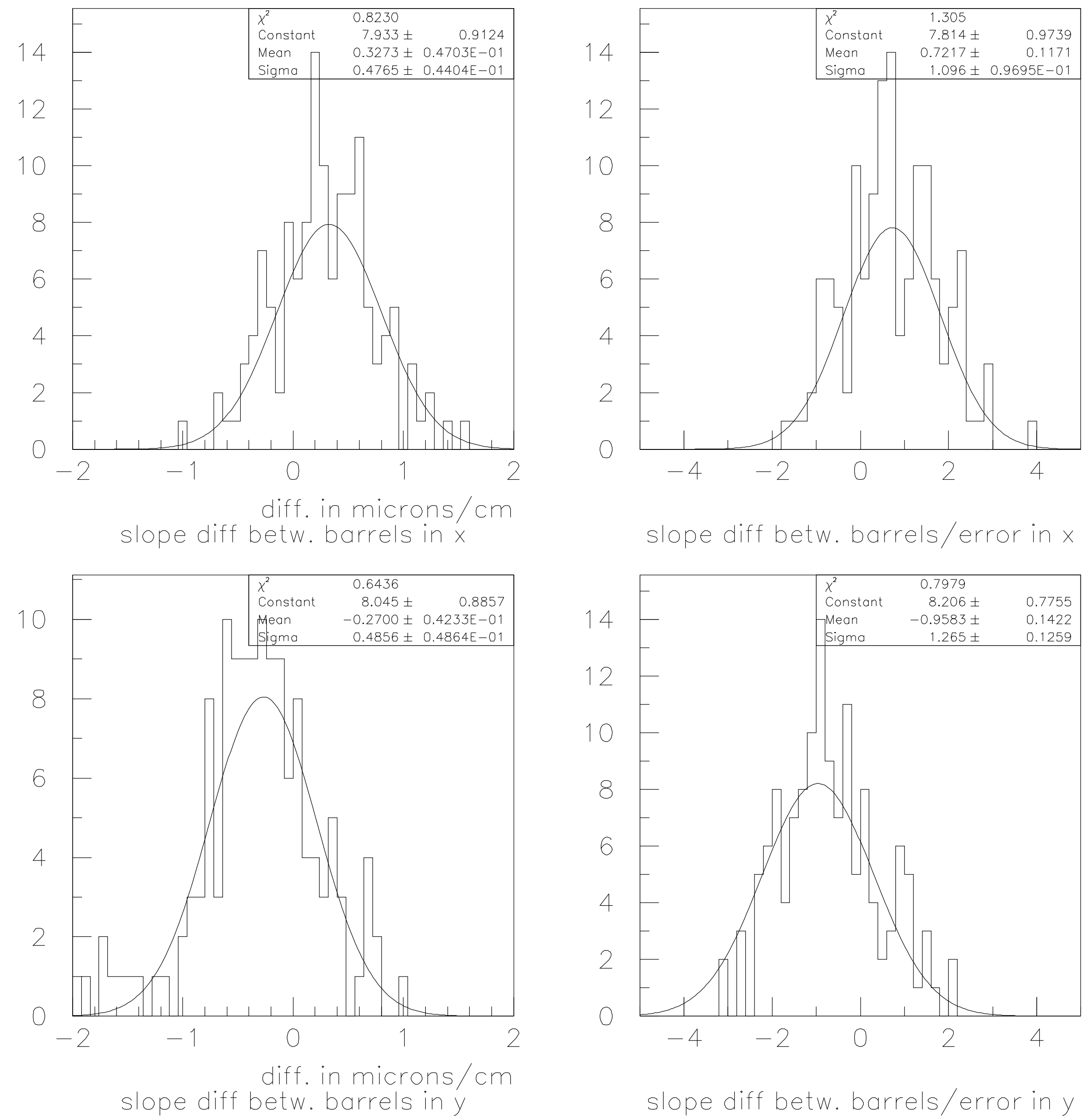

Figure 5: Difference and normalized difference in x/y slopes

4 Position of the beam in the first months of data taking

Figure 6 and 7 show the position of the beam in x and y over the first months of data taking. The position in x changed by up to 300μm while in y we observe one big jump of about 900 μm.

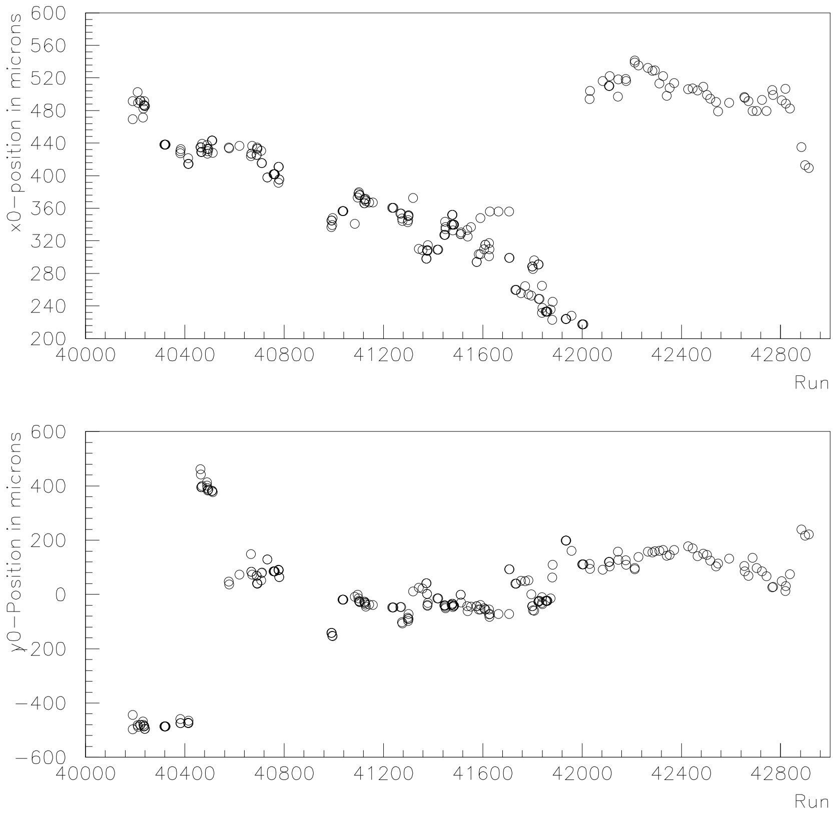

Figure 6: Beam position x0 and y0 vers. Run number

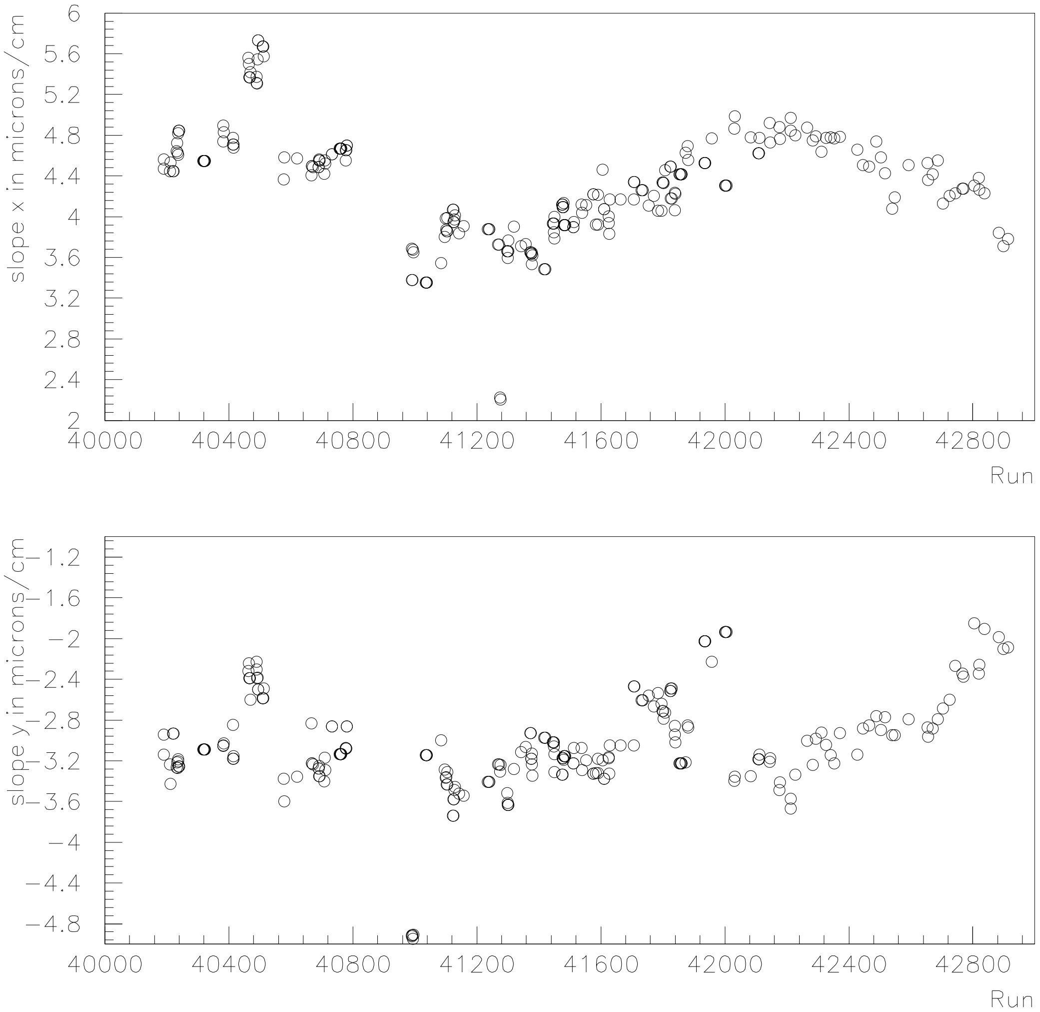

Figure 7: Slope in x and y vers. Run number

5 How stable is the beam position?

One important question is if the beam is stable in time or if it is moving. In case it is moving one wants to know how much. It is expected that the beam position is fairly stable during a store. Since the fit was done on a run by run basis it is possible to compare the results of different runs for the same store. Since the runs are getting longer it might be interesting to do the fit every couple of hundred events.

Another interesting possibility would be to use a feed back system which adjusts the accelerator magnets so that the beam is kept at the same position all the time.

Figure 8 shows the absolute values of the difference between two subsequent runs within the same store for x0 and y0. The mean value in x is about 5 microns and in y it is about 11 microns. In y one store (3962) shows a big variation (more than 40 microns) between the runs within the same store.

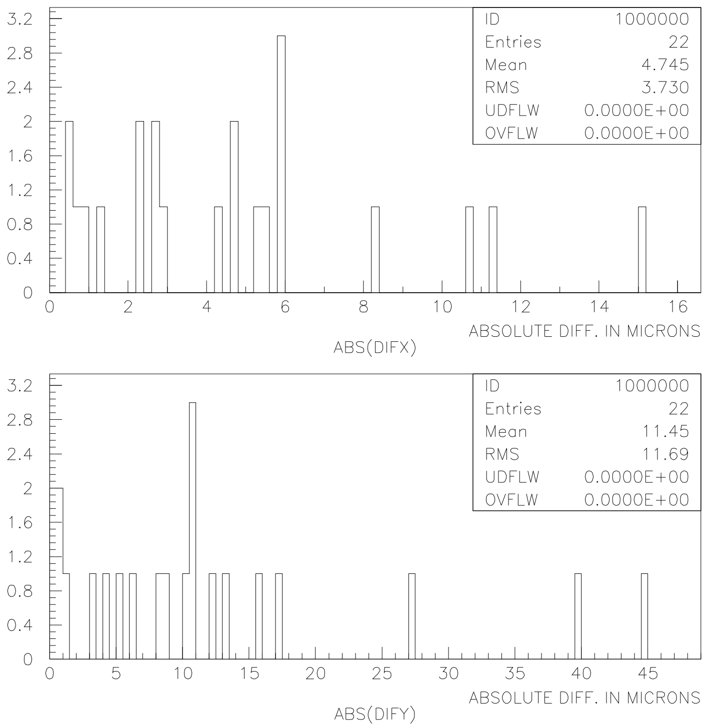

Figure 8:

Appendix

```
C
C          AUTHOR: HANS WENZEL FERMILAB
C
C       BEAM Position constants as calculated by the SVX detector
C
        INTEGER REQRUN,RUNTAK,ISTORE
        REAL XSVBEA,YSVBEA,SVSLOX,SVSLOY, SVSIGX,SVSIGY,SVCOXY
        REAL SVCOMA(4,4)
        LOGICAL SVFOBM
C
        COMMON/SVXBPO/REQRUN,RUNTAK,ISTORE,
      1 XSVBEA,YSVBEA,SVSLOX,SVSLOY,SVCOMA,
      2 SVSIGX,SVSIGY,SVCOXY,SVFOBM
C
C       REQRUN: RUN WHICH HAS BEEN REQUESTED BY LAST CALL OF
C       SVGEBP
C       RUNTAK: RUN WHICH HAS BEEN USED TO GET THE BEAM POSITION
C          (e.g. IN CASE THE REQUESTED RUN DIDN'T HAVE ENOUGH EVENTS
C          TO DO THE FIT THE NEAREST RUN IN THE SAME STORE IS TAKEN)
C       ISTORE: STORE NUMBER TO WHICH THE REQUESTED RUN BELONGS
C       XSVBEA: X-POSITION OF BEAM AT Z = 0
C       YSVBEA: Y-POSITION OF BEAM AT Z = 0
C       SVSLOX: SLOPE OF BEAMLINE IN X
C       SVSLOY: SLOPE OF BEAMLINE IN Y
C       SVSIGX: SIGMA OF BEAMSPOT IN X
C       SVSIGY: SIGMA OF BEAMSPOT IN Y
C       SVCOMA(4,4): COVARIANCE MATRIX OF THE VALUES ABOVE
C       SVCOXY: CORROLATION TERM (POSITION OF AXIS)
C       SVFOBM: LOGICAL INDICATING THAT DATABASE ENTRY FOR THIS RUN
C                  HAS BEEN FOUND
```

Figure 9: printout of C$INC:SVXBPO.INC