Online Beam Width Measurement at CDF

Chris Neu

CDF/D0/AD Luminosity Meeting
13 July 2004

Outline:

- **Motivation**
- **Brief intro to the technique**
- **Offline testing**
- **Online testing: FAKEHITS and beam tests**
- **Results for store 3576**
- **Open Issues**

Motivation

- **Accelerator Division has requested feedback from the experiments on the beam width near the IP:**
  - Widths are an ingredient to measurement of β^*, a figure-of-merit in determining beam loss
  - AD prefers real-time feedback for quick check of Tev-studies tunes
  - Understanding losses helps promote general Tevatron health

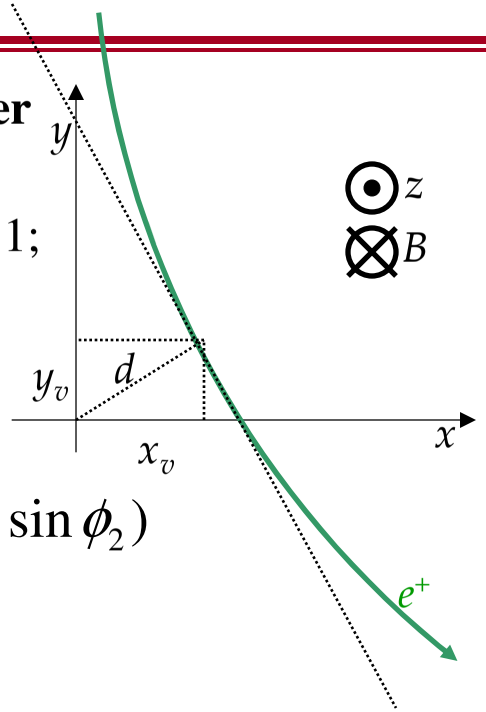
- **There is a clear benefit for CDF:**
  - The p and pbar beams are focused into the nominal center of CDF
  - The *xz* and *yz* "waists" of the colliding beams however do not coincide at the same point in *z*
    - NB: Things appear to have improved over the past 6 months
    - Improvements can still be made
  - With coincident beam waists, ***the luminosity goes up***

- **CDF measures the beam width and β^* in 2 ways:**
  - Offline studies using the locations of primary vertices
  - Online feedback using tracks recorded in our silicon detector

Technique

- Calculation uses **track pairs from the Silicon Vertex Tracker (SVT)**, a component of the **CDF trigger**
- First exposure in CDF with this technique was studied in Run 1; See CDF 4189 and forthcoming Run 2 note.
- Technique:
  - Recall: $d = y_v \cos\phi - x_v \sin\phi \quad \Rightarrow$

$$d_1 d_2 = (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2)$$



  - Now collect a large number of **track pairs** and take the average:

$$\langle d_1 d_2 \rangle = \langle (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2) \rangle \Rightarrow$$

$$\langle d_1 d_2 \rangle = \frac{1}{2}(\sigma_x^2 + \sigma_y^2)\cos(\Delta\Phi) + \frac{1}{2}(\sigma_x^2 - \sigma_y^2)\cos(2\Phi)$$

where

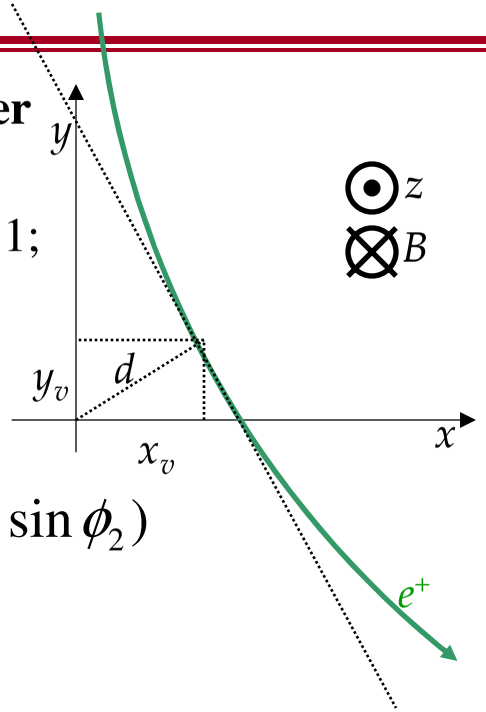
$$\langle x_v^2 \rangle \equiv \sigma_x^2 \quad \langle y_v^2 \rangle \equiv \sigma_y^2 \quad \langle x_v y_v \rangle = 0 \quad \Delta\Phi \equiv (\phi_1 - \phi_2) \;\; and \;\; 2\Phi \equiv \phi_1 + \phi_2$$

NB: this all assumes <d1>=<d2>=0, which is not indeed the case in practice. The fitter handles the real situation properly.

Technique

- Calculation uses **track pairs from the Silicon Vertex Tracker (SVT),** a component of the **CDF trigger**
- First exposure in CDF with this technique was studied in Run 1; See CDF 4189 and forthcoming Run 2 note.
- Technique:
  - Recall: $d = y_v \cos\phi - x_v \sin\phi \quad \Rightarrow$

$$d_1 d_2 = (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2)$$



  - Now collect a large number of **track pairs** and take the average:

$$\langle d_1 d_2 \rangle = \langle (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2) \rangle \quad \Rightarrow$$

Important!

$$\langle d_1 d_2 \rangle = \frac{1}{2}(\sigma_x^2 + \sigma_y^2)\cos(\Delta\Phi) + \frac{1}{2}(\sigma_x^2 - \sigma_y^2)\cos(2\Phi)$$

We know this

where

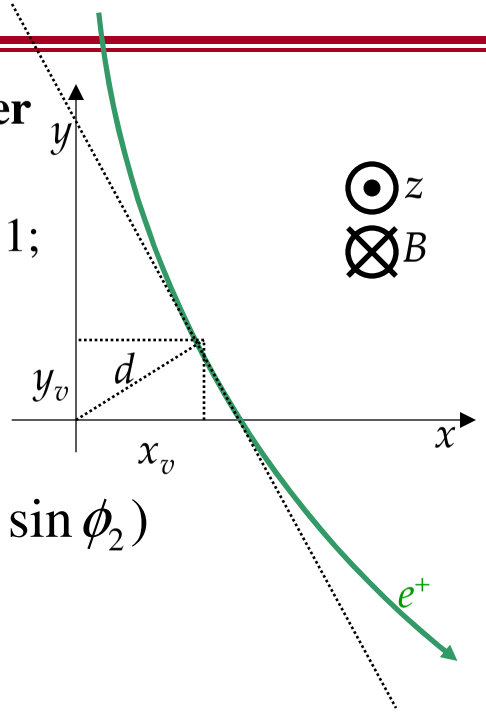
$$\langle x_v^2 \rangle \equiv \sigma_x^2 \qquad \langle y_v^2 \rangle \equiv \sigma_y^2 \qquad \langle x_v y_v \rangle = 0 \qquad \Delta\Phi \equiv (\phi_1 - \phi_2) \;\; and \;\; 2\Phi \equiv \phi_1 + \phi_2$$

NB: this all assumes <d1>=<d2>=0, which is not indeed the case in practice. The fitter handles the real situation properly.

Technique

- Calculation uses **track pairs from the Silicon Vertex Tracker (SVT),** a component of the **CDF trigger**
- First exposure in CDF with this technique was studied in Run 1; See CDF 4189 and forthcoming Run 2 note.
- Technique:
  - Recall: $d = y_v \cos\phi - x_v \sin\phi \quad \Rightarrow$

$$d_1 d_2 = (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2)$$



  - Now collect a large number of **track pairs** and take the average:

$$\langle d_1 d_2 \rangle = \langle (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2) \rangle \Rightarrow$$

Important!

$$\langle d_1 d_2 \rangle = \frac{1}{2}(\sigma_x^2 + \sigma_y^2)\cos(\Delta\Phi) + \frac{1}{2}(\sigma_x^2 - \sigma_y^2)\cos(2\Phi)$$

We know this

And we know these guys

where

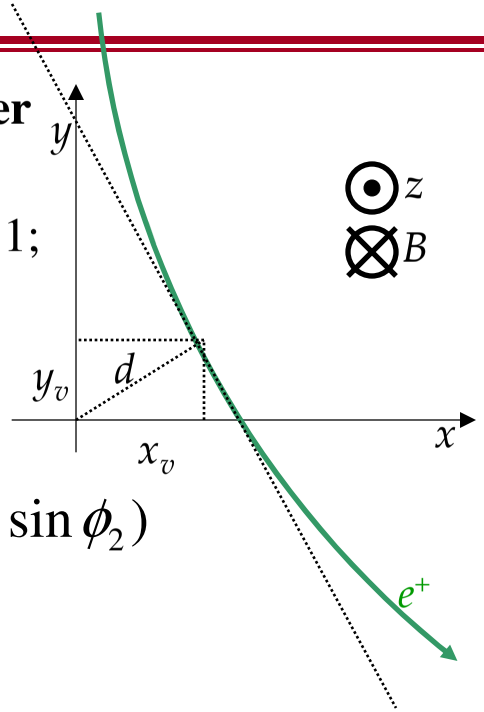
$$\langle x_v^2 \rangle \equiv \sigma_x^2 \quad \langle y_v^2 \rangle \equiv \sigma_y^2 \quad \langle x_v y_v \rangle = 0 \quad \Delta\Phi \equiv (\phi_1 - \phi_2) \;\; and \;\; 2\Phi \equiv \phi_1 + \phi_2$$

NB: this all assumes <d1>=<d2>=0, which is not indeed the case in practice. The fitter handles the real situation properly.

Technique

- Calculation uses **track pairs from the Silicon Vertex Tracker (SVT),** a component of the **CDF trigger**
- First exposure in CDF with this technique was studied in Run 1; See CDF 4189 and forthcoming Run 2 note.
- Technique:
  - Recall: $d = y_v \cos\phi - x_v \sin\phi \quad \Rightarrow$

$$d_1 d_2 = (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2)$$



  - Now collect a large number of **track pairs** and take the average:

And we want to know these

$$\langle d_1 d_2 \rangle = \langle (y_{v1}\cos\phi_1 - x_{v1}\sin\phi_1)(y_{v2}\cos\phi_2 - x_{v2}\sin\phi_2) \rangle \quad \Rightarrow$$

Important!

$$\langle d_1 d_2 \rangle = \frac{1}{2}(\sigma_x^2 + \sigma_y^2)\cos(\Delta\Phi) + \frac{1}{2}(\sigma_x^2 - \sigma_y^2)\cos(2\Phi)$$

We measure this

And we know these guys

where

$$\langle x_v^2 \rangle \equiv \sigma_x^2 \qquad \langle y_v^2 \rangle \equiv \sigma_y^2 \qquad \langle x_v y_v \rangle = 0 \qquad \Delta\Phi \equiv (\phi_1 - \phi_2) \;\; and \;\; 2\Phi \equiv \phi_1 + \phi_2$$

NB: this all assumes <d1>=<d2>=0, which is not indeed the case in practice. The fitter handles the real situation properly.

Technique

- So we gather a bunch of track pairs (restricted to the same crossing/event)
- Construct the following χ^2:

$$\chi^2 = \sum_{i=1}^{12}\sum_{j=1}^{12}\left[\frac{a\cos(\Delta\phi)_i - b\sin(2\phi)_j - \frac{1}{N_{Pairs\ ij}}\sum_{p=1}^{Npairs_{ij}}(d_1 d_2)_p}{\sigma_{\langle d_1 d_2\rangle_{ij}}}\right]^2$$

where

$$a \equiv \frac{1}{2}(\sigma_x^2 + \sigma_y^2) \quad and \quad b \equiv \frac{1}{2}(\sigma_x^2 - \sigma_y^2)$$

$$and \quad \sigma_{\langle d_1 d_2\rangle_{ij}} = \frac{1}{N_{Pairs\ ij}}\sqrt{\frac{N_{Pairs\ ij}\sum_{p=1}^{Npairs_{ij}}(d_1 d_2)_{ij}^2 - \left(\sum_{p=1}^{Npairs_{ij}}(d_1 d_2)_p\right)^2}{N_{Pairs\ ij} - 1}}$$

- And **minimize wrt *a,b***, thus giving us the ***x* and *y* widths**.
- Errors on *d* do not affect the correlation if we assume
  - the error on track j is uncorrelated to the impact parameter of track i
  - the average track impact parameter is unbiased and therefore average error is zero

Event/Track/Pair Selection

- We only consider tracks of a certain type:
  - $pT > 2$
  - $d < 0.02$ cm
  - $X^2 < 15$
  - Duplicate tracks rejected
  - $z_{in} = z_{out}$
  - $Wedge_{in} = Wedge_{out}$
- We only consider events that have fewer than 5 SVT tracks
  - Avoid high track multiplicity jets – they can skew measurement
    - Imagine a 10-track jet – typically confined to one wedge – it contribute to fit 10-choose-2 times!
  - This is a real effect…did some beam tests with 10-track limit and fitter was completely unstable
- We further reduce statistics by demanding track pairs have $\Delta\varphi > 0.02$

Offline Testing

- **Offline toy fitter** developed on fcdfsgi2
  - Runs using **ASCII input** made from SVT Ntuples in Paw
  - Uses same framework as online system wherever possible
- **Limitations:**
  - Cannot process any given run
    - SVT Ntuples made automatically for **only a certain subset** of collision events
    - This sample is **physics-rich**, meaning **lots of long-lived particles**
    - ruins width measurement
  - Would prefer of course other data with events that just have SVT tracks at L2
  - Fixable problem - manpower/time required to fix things not available
  - Used four special SVT runs from several years back
    - Results looked good in both circular-beam approximation (CBA) and x- and y-independent fits

Status: Online Testing

- Online version 1 of the code is complete

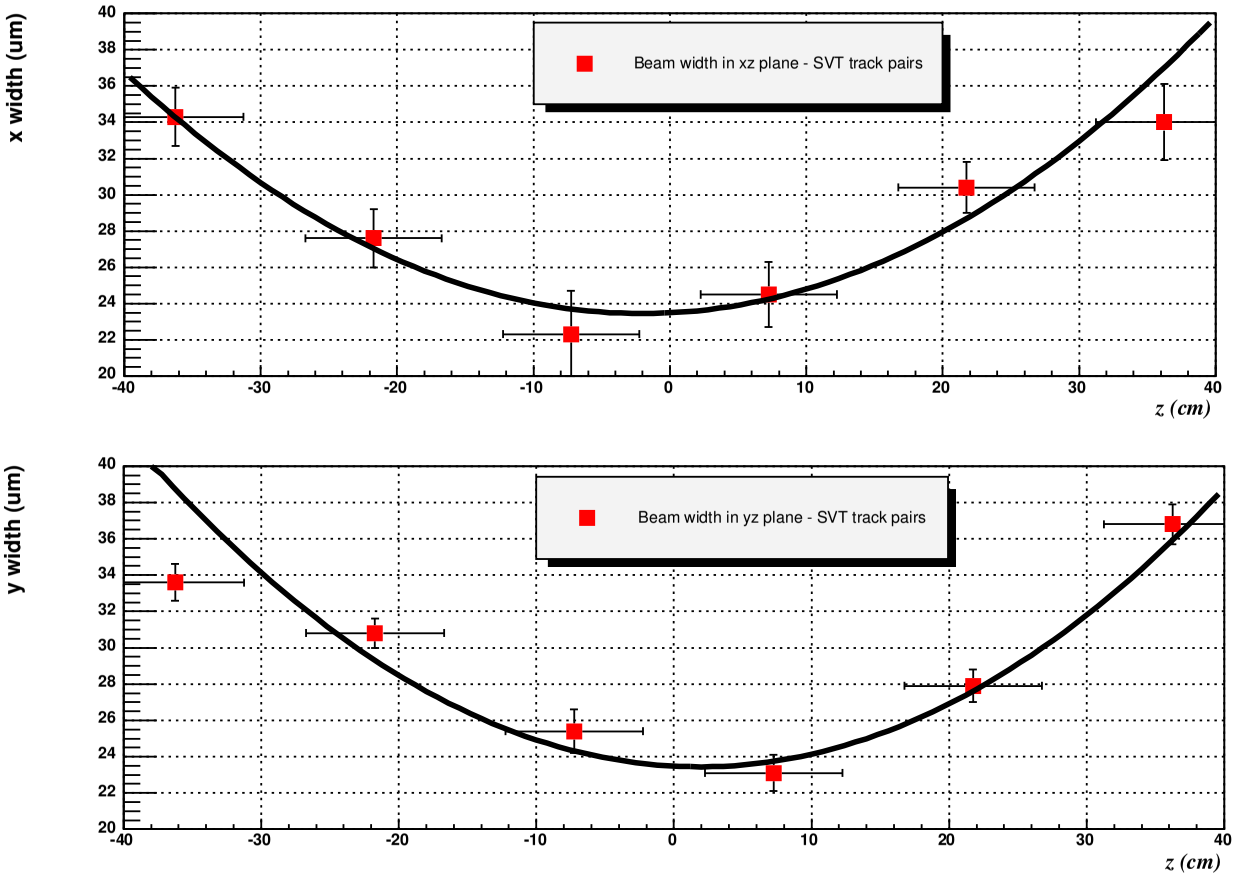
- Code runs within **SVTMON**
  - Online monitoring suite that spies on all the SVT tracks and track triggers
  - This monitor provides the online beam position from CDF
  - Runs in a VME crate CPU running VxWorks OS

- **Online environment** nothing like the offline toy fitter development
  - VxWorks is **not an easy platform** in which to do development
  - **Hard to debug problems**, like memory corruption/exhaustion
  - Have to deal with **pesky CDF/AD schedules**
  - The experiment is running well, **reluctant to break it**

- Other differences that caused head-scratching:
  - **Memory corruption**
  - **Duplicate track removal**
  - Need to **avoid high track multiplicity events**
  - Frequency of purging fit stats
  - Communication to Acnet only test-able online

Online Testing

- **Tested in beam** over the past 2.5 months
  - Non-negligible amount of learning and trial-and-error involved in getting fitter up and running
  - **Fitter now performing well in the online system**
  - No significant interference with beam position calculation



Store 3582 (6/17/2004) x and y widths as a function of z.

Vertical errors are statistical; horizontal errors were **put in by hand to acknowledge missing z information (did not have time to retrieve).**

Really seek much smaller statistical errors…

No validation with offline

Online Testing: Store 3576

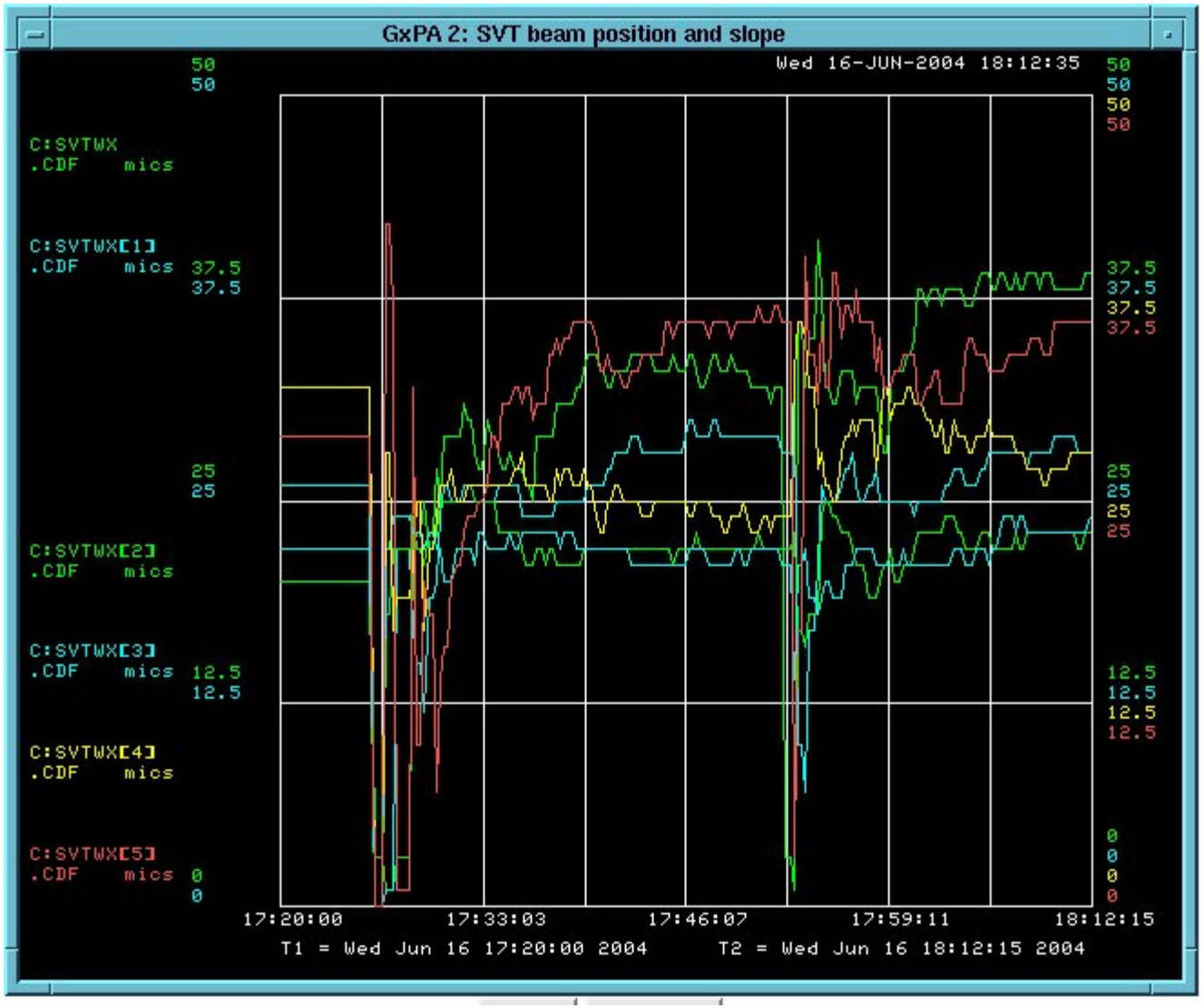
Communication to Acnet is working.

There are 24 devices filled. They can be found on C82, subpage 16 of Acnet monitor.

C:SVTWX[0-5]
C:SVTWXE[0-5]
C:SVTWY[0-5]
C:SVTWYE[0-5]

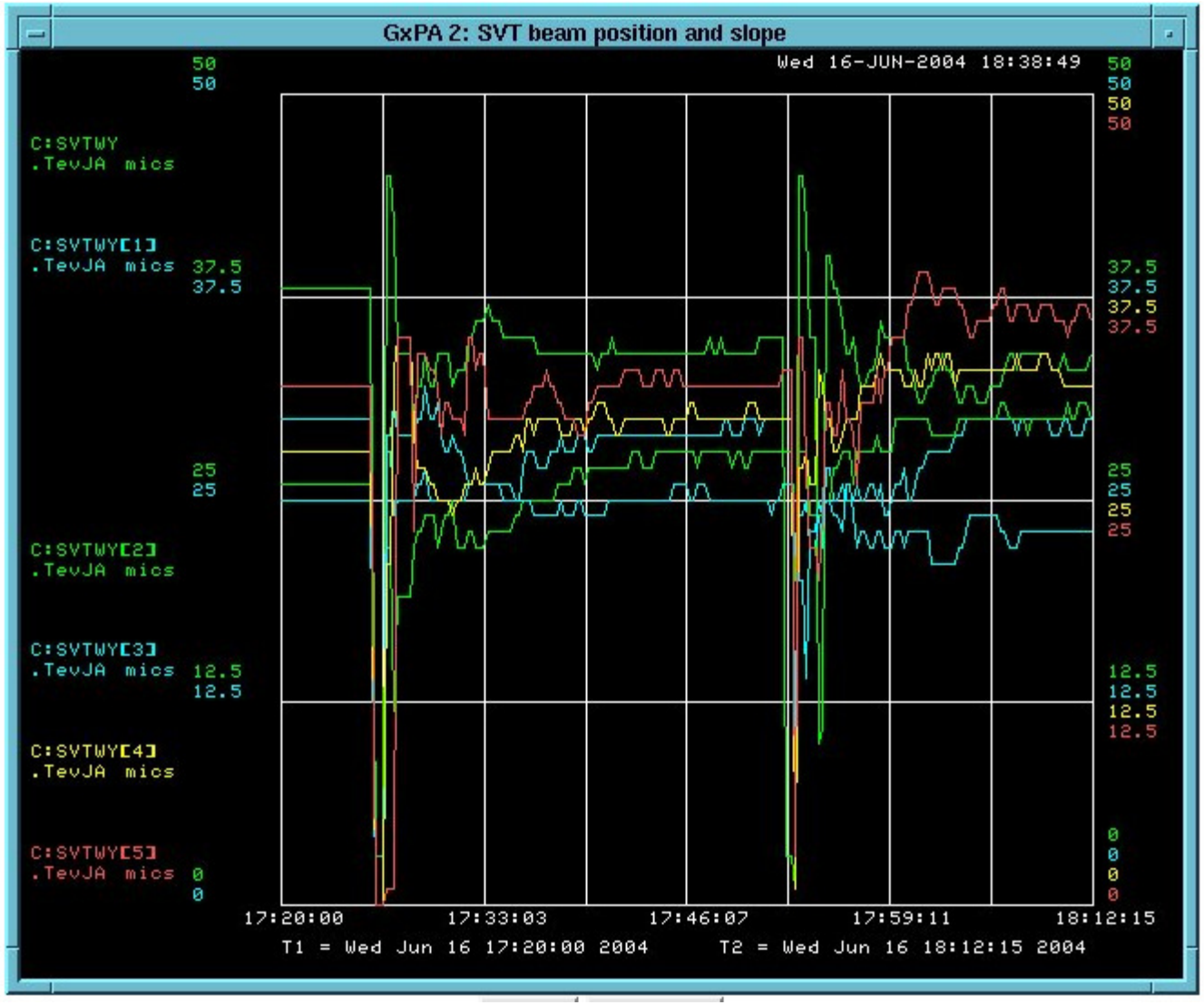
NB: while the widths are recorded in the CDF logger, **the errors on the widths cannot be found in any logger**. This needs to be fixed.

X-widths



Online Testing: Store 3576

Y-widths



Periodic deviation is due to **purging of statistics accumulators. A flush of stats was demanded every 50k track pairs (30min of running at 22E30)**

One item on the to-do list is to **improve how this clean-up is done**.

One idea is to calculate a running average in the widths using a **ring-type of calculation** instead of simply starting from scratch every few 100k track pairs.

Outstanding Issues

- Code is ready to reside permanently in CDF online environment – needs to be adopted and made default
  - Was attempted 2 weeks ago by SVT experts
  - They ran into some problems
  - These will be fixed this week and code will be running at all times
- Improvements to fit calculation envisioned:
  - We want time evolution without interruption
    - Use ring-type calculation
  - Need to report real z information to get more accurate β^*
  - Width errors logged properly in Acnet
- Offline validation needs to be completed not only for this store, but in an automatic way for all stores
  - Some sort of automated mechanism for validation would be nice
  - But certainly online/offline comparisons should be done for stores before and after any significant Tevatron events, like optics changes, shutdowns, catastrophes, etc.

Summary

- At CDF we use SVT tracks to determine the beam width in x and y with crude z dependence
- Offline testing is limited
- Online testing is not easy, but beam tests have been successful

- First version ready to run permanently in online system

- Need to resolve a few open issues

- Need to coordinate online/offline validation and online monitoring