CLAS-NOTE-2003-008
May 30, 2003

Raster Corrections for EG1b

P. Bosted
University of Massachusetts, Amherst, MA 01003

S. Kuhn
Old Dominion University, Norfolk, Virginia 23529

Y. Prok
University of Virginia, Charlottesville VA 22901

Abstract

The conversion of ADC counts to x and y position on the EG1b target is fit to data and examined as a function of run number. These were then used to provide better z vertex information, which can be used to make tighter cuts. A correction formula for the ϕ angle of charged particles based on raster position is given. Raster information is used to look for cases where the beam was mis-steered and hit the edges of the target cups.

1 Introduction

In EG1b, the beam was rastered using two magnets up-beam of the target. ADC's recorded the current going to these magnets, and the values are stored in the DST's for each trigger. To make the ADC values useful, a procedure was developed to translate ADC counts into x and y relative to the CLAS beam line. This could then be used to make corrections to the tracking (which assumes x and y are zero) which allows better z vertex reconstruction. This allows better rejection of events from up-beam and down-beam windows (especially for particles at small angles), and could also be used to reduce accidental coincidences in multi-particle final states (or to look for offset decays such as from the Λ). Knowing x and y allows a correction to the ϕ angle of the particles to be made, improving missing mass resolution for multi-particle final states. Finally, plotting the number of events as a function of raster information is useful in looking for mis-steered beam that hit the edges of the target cups.

2 Conversion of ADC counts to cm

Assuming the raster magnets have a linear relation to position, we fit the x and y raster ADC values using the form

$$x = (X - X_0) * c_x$$

$$y = (Y - Y_0) * c_y$$

where x and y are the raster positions in cm, X and Y are the ADC values. For the fitting, we selected events with an electron and positron in coincidence, mainly because we happened to have a reasonable sample of these events for each run for another purpose. Using Minuit, we then minimized χ^2, defined as

$$\chi^2 = \sum_1^N (z_c - z_0)^2 \tag{1}$$

where z_0 is a fit parameter that defines the center of the target, and the corrected vertex position z_c is given by

$$z_c = z_{nom} + x'/\tan(\theta)$$

where z_{nom} is the vertex z found by the tracking code assuming $x = y = 0$, θ is the particle angle relative to the beam line, and

$$x' = [x\cos(\phi_s) + y\sin(\phi_s)]/\cos(\phi - \phi_s)$$

is a measure of the distance in cm along the track length that was not taken into account in the tracking, and where ϕ_s is the sector angle given in degrees by $\phi_s = (S - 1) * 60.$), S is the sector number from 1 to 6, and ϕ is the azimuthal angle of the particle in the same coordinate system, defined as $\phi = \mathrm{atan2}(p_y, p_x)$, where p_x and p_y are the RECSIS momentum components in the x and y directions.

In some cases, the raw ADC values for x and y are almost constant and have values around 500 to 1000. These appear to be pedestal values, corresponding to no voltage from the raster magnet system. This seems to have generally been the case before run 25700. During run 25700, the first part looks like pedestals, then the raster ADC values started spanning the normal range, which is about 2000 to 6000 counts. We therefore put a cut in the analysis that both raster ADC values had to be greater than 1500.

The results for the four primary fit variables are shown in Fig. 1 as a function of run number for all runs that have so far been "cooked" and have raster information. The first set of runs is at 1.6 GeV, while the rest are all at 5.6 or 5.7 GeV. Basically, the results are quite stable to the tolerance that we care about them, with the notable exception of Y_0. This is because before Christmas, the beam we deliberately steered high to match the target cups. After Christmas, the target stick was changed so that the targets could be better centered on the beam. In practice, the runs were divided into three groups, with the average parameters shown for these groups as the lines on each plot. The results for the early run numbers are all at 1.6 GeV, where the raster pattern was smaller than at 5.x GeV, and the fits are more unstable.

We note that c_x and c_y seem to change between 1.6 and 5.x GeV energies, but are very stable within a given energy, and also agree pretty closely with each other, which would be expected if both magnets had the same strength and the controllers were run similarly to get s round raster pattern.

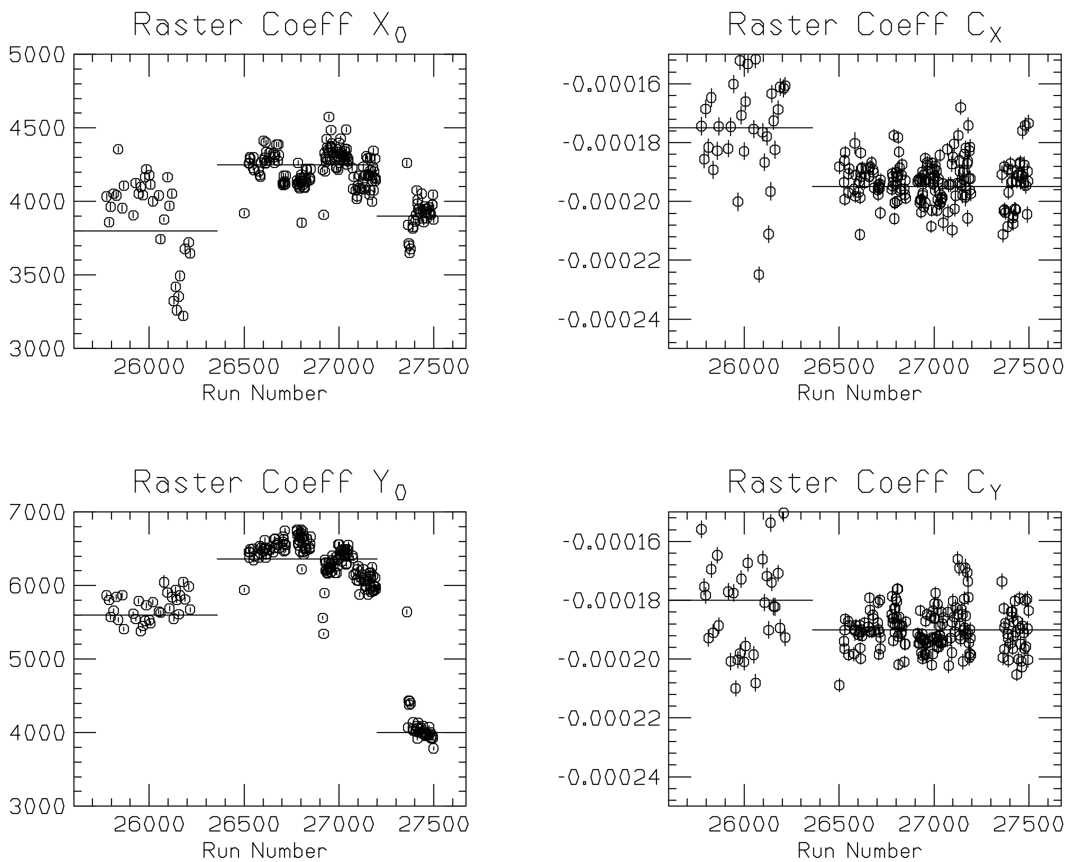

Figure 1: Fits to the four primary raster coefficients as a function of EG1b run number. The lines indicate the average values used in three time periods by the correction subroutine.

Figure 2 shows the fitted value of z_0 as a function of run number. It is quite stable with time, although shows a slight shift after Christmas, when the target was moved. The middle plot shows the reconstructed z position for electrons with (solid) and without (dashed) correction. Clearly the raster correction makes a big improvement. For small angle outbending particles, where θ can be as small as 6 degrees, the correction can be as much as 5 cm. With correction, a much tighter cut on z can be placed to reject background from windows. The right plot shows the difference in z for electron/positron events. Again, the improvement is quite dramatic, although the resolution is not good enough to identify where in the 1 cm long cell the interaction took place.

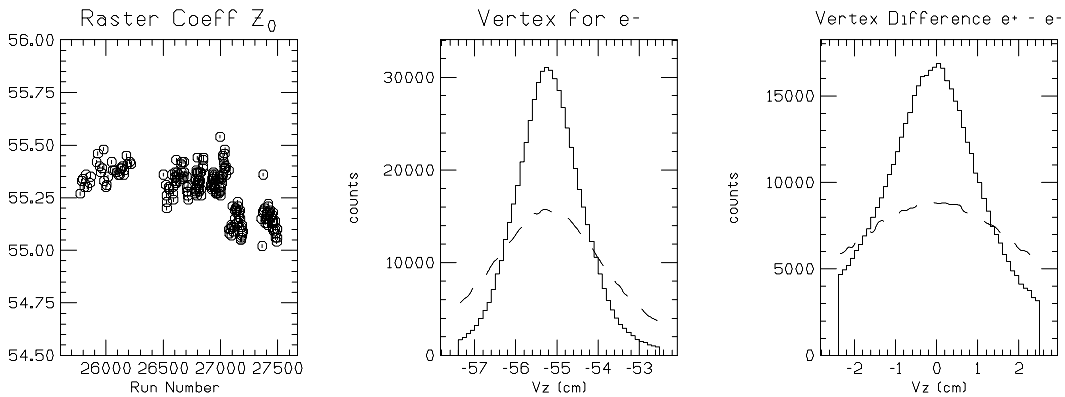

Figure 2: Fit values of z_0 versus run number are shown on the left plot. The middle plot shows reconstructed z with/without raster corrections (solid/dashed curves). The right plot shows the improvement in z difference between the electron and the positron in an e^+e^- event.

3 ϕ correction

The raster correction implies that there is a different track length for a particle traveling through the 50 kG magnetic field of the target than the tracking assumed, which means that the ϕ rotation is incorrectly calculated. This is corrected using:

$$\phi_c = \phi_0 - (q)(50)(x'/100.)/33.356/p_t$$

where q is the particle charge (± 1, the factor of 50 is the field in kG, the factor of 100 is to convert cm to m, the factor of 33.356 is the inverse speed of light in the appropriate units, and p_t is the transverse momentum of the particle in GeV, where $p_t = P\sin(\theta)$.

As it happens, both q and x' reverse sign for ep elastic events, so both particles are rotated the same amount, which doesn't matter in data analysis. However, for exclusive reactions with more than two particles, there is no such cancelation, and missing mass quantities can be improved by applying this correction. An example is shown in Fig. 3, where the solid line shows the missing mass squared for the $ep\pi^+\pi^-$ reaction (when all particles detected) with the correction, while the dashed curve is without the correction.

4 Target imaging

Using the raster information, it is informative to plot the number of events as a function of x and y. An example is shown in Fig. 4. Making these plots revealed that the target cups were often not completely full, although generally they were at least 90% full. Also revealed were runs where the beam was mis-steered and was significantly hitting the target cups (as in the lower right of the figure). Some NH3, ND3, and C runs were eliminated from the dilution factor analysis because of this problem.

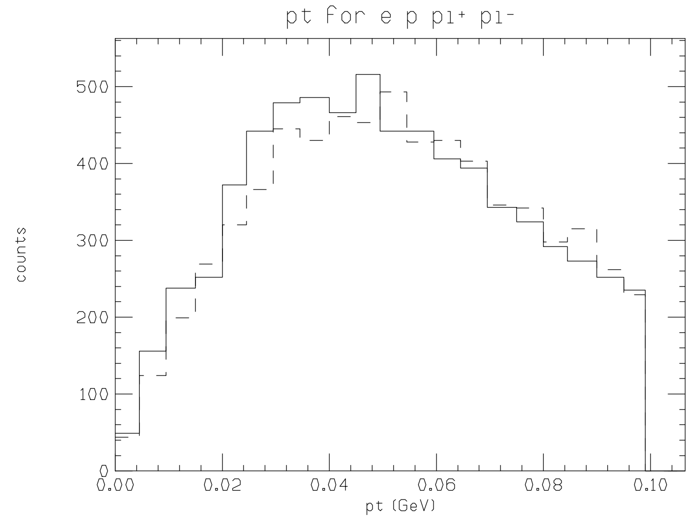

Figure 3: Missing transverse momentum for exclusive $p(e, e'p\pi^+\pi^-)$ events with/without the raster ϕ correction (solid/dashed histograms).

5 Correction Code

The code to find x and y, and correct p_x and p_y, given a pair of raster ADC values and a run number, can be found at
/site/www/html/Hall-B/secure/eg1/EG2000/Bosted/raster_corr.F

The code is:

```
c-----------------------------------------------------------------------
c    Raster Corrections for EG1b
c    P. Bosted May 30, 2003 version 2
c    Function: correct z vertex and x,y components of track
c              momentum for raster position. Transverse
c              momentum (sqrt(px**2+py**2) is preserved.
c    Arguements:
c        run       run number (integer) (input only)
c        raster_x  ADC value (integer)(input only)
c        raster_y  ADC value (integer)(input only)
c        sector    secotr of track (integer) (input only)
c        q         charge of track (integer) (input only)
c        trl1_theta angle of track (real) (input only)
c        p_x       p_x of track (real) (input and output both)
c        p_y       p_y of track (real) (input and output both)
c        v_z       p_z of track (real) (input and output both)
c        x_rast    raster x in cm (real) (output only)
c        y_rast    raster y in cm (real) (output only)
c   Example of usage in FORTRAN dst-reading code:
```

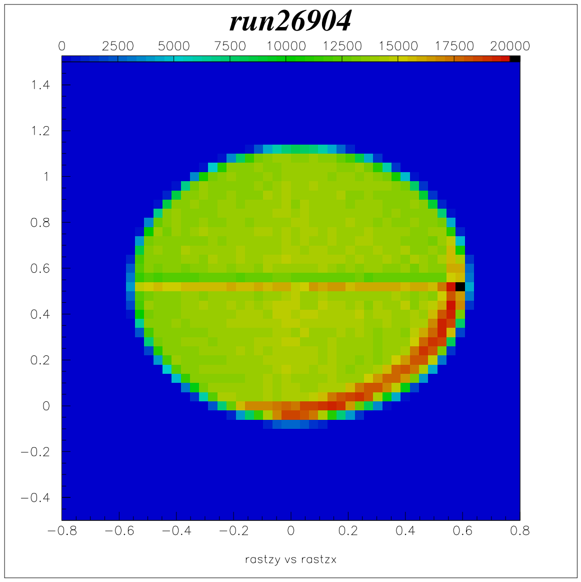

Figure 4: A plot of events as a function of raster x and y. The red zone on bottom right is where beam was hitting the target walls.

```
c      real x_rast,y_rast
c      do i=1,n_part
c        call raster_corr(run,raster_x,raster_y,sector(i),
c    >     q(i),trl1_theta(i),p_x(i),p_y(i),v_z(i),x_rast,y_rast)
c      enddo
c
c-----------------------------------------------------------------------
       subroutine raster_corr(run,raster_x,raster_y,sector,q,
     >      trl1_theta,p_x,p_y,v_z,x_rast,y_rast)
       implicit none
       integer run,raster_x,raster_y,sector,q
       real p_x,p_y,v_z,x_rast,y_rast,trl1_theta
       real phir,phid,pt,xp

c check data is valid
       if(q.eq.0) return
       if(raster_x.le.0.or.raster_x.ge.10000) return
       if(raster_y.le.0.or.raster_y.ge.10000) return
       if(raster_x.lt.1500.and.raster_y.lt.1500) return
       pt = sqrt(p_x**2 + p_y**2)
       if(pt .eq.0.) return
       if(trl1_theta.eq.0.) return

c convert ADC readings to cm. Offsets divided into three
```

```
c run groups because target center was moved over Christmas
c and centering of beam changed between 1.6 and 5.7 GeV
c in the eraly part of the run
c For 1.6 GeV
      if(run.le.26360) then
        x_rast = (raster_x-3800.) * -0.000175
        y_rast = (raster_y-5600.) * -0.000180
      endif
c for 5.7 GeV up to Xmas
      if(run.gt.26360.and.run.le.27200) then
        x_rast = (raster_x-4250.) * -0.000195
        y_rast = (raster_y-6360.) * -0.000190
      endif
c after Xmas
      if(run.gt.27200) then
        x_rast = (raster_x-3900.) * -0.000195
        y_rast = (raster_y-4000.) * -0.000190
      endif

! Find sector phi, and phi of track
      phir=(sector-1) * 60. * 3.1415928/180.
      phid=atan2(p_y,p_x)

! Find displacement along direction of track
      xp = (x_rast * cos(phir) + y_rast * sin(phir))/
     >          cos(phid-phir)

! Correct vertex z by tracing back
      v_z = v_z + xp / tan(trl1_theta * 3.1415928/180.)

! Find true track phi based on 5T field, and get new p_x, p_y
      phid = phid - float(q) * 50.0 * (xp/100.) / 33.356 / pt
      p_x = pt * cos(phid)
      p_y = pt * sin(phid)
      return
      end
```