Resolution Corrections for Au-Au 200GeV Elliptic Flow Analysis

Erik Johnson
04/17/06

The resolution correction is determined for the elliptic flow spectrometer analysis. The values of the resolution correction are determined for two sets of detectors that are outside the spectrometer acceptance. These sets are use for different vertex ranges. By default, Tile ring 1 should be used for negative vertices, while a combination of the flow rings, silicon ring 1, and the left large BB counters is used for the positive vertices.

Table of Contents:
Defining Event Properties.....page 2
Binning of the Data.....page 3
Procedure to Define and Correct Reaction Plane.....page 4
Determining Resolution Correction.....page 7
Mapping Where to Use the Correction in Vertex....page 22

Note: All of the code and ascii data files are located in cvs.
The sim values are located at brahms_app/ebj_app/Simulation/BFlowS/ResCor/
The data values are at brahms_app/ebj_app/Flow/ResCor/

Note: For both cases an executable script was made to process the data completely.

Defining the Event Properties:

The vertex and centrality have been saved in the Au-Au 200GeV dsts. Here is the procedure for selecting the best vertex.

```
Int_t vz_type = -1;  // 0 Spec Vtx, 1 BB Vtx, 2 ZDC LED Vtx
Float_t fVertex = -1111;
if ( TMath::Abs(G_fZdcVtxZ[0])      < 50 )
        { fVertex = G_fZdcVtxZ[0];      vz_type = 2; }   // ZDC
if ( TMath::Abs(G_fBbVtxZ[0])       < 50 )
        { fVertex = G_fBbVtxZ[0];       vz_type = 1; }   // BB
if ( G_fTrackProjVtxZ[0] < 50 && G_fTrackProjVtxZ[0] > -20
    && G_fNoTpm1Track[0] > 5  )
        { fVertex = G_fTrackProjVtxZ[0]; vz_type = 0; }   // Spec
if ( fVertex == -1111 ) { return kFALSE; }
```

Now for the centrality, the data and simulation are different.
For the data:

```
Float_t fcent = G_fCent[0];
if ( TMath::Abs(fVertex) > 36 ) { fcent = G_fTMACent[0]; }
```

For the simulation, the input centrality is used. Since this value is not stored in the dst, the inverse function used to convert the centrality to impact parameter is used to recover the input centrality. On the off chance, that this function returns an invalid value, the Npart value is used to give the centrality.

Here is the function to convert the impact parameter value to centrality.

```
TF1 *fbtocent = new TF1("fbtocent","pol11",0,30);
Double_t btcparams[12] = {-1.88902,3.36771,-1.02008,0.228512,
                          -0.00819414,-0.000853076,1.47656e-05,4.18602e-06,
                          1.41687e-07,-1.18726e-08,-1.28598e-09,5.98504e-11};
fbtocent->SetParameters(btcparams);
```

Here is the function to convert the Npart value to the centrality.

```
fnparttocent = new TF1("fnparttocent","pol11",0,400);
Double_t nparttcparams[12] = {98.1551,-2.7312,0.0740081,-0.00113816,
                              9.35896e-06,-3.86856e-08,4.95359e-11,1.56815e-13,
                              -3.99039e-16,-8.83981e-19,3.74664e-21,-3.36026e-24};
fnparttocent->SetParameters(nparttcparams);
```

The centrality is then.

```
Float_t fcent = fbtocent->Eval(GN_fb[0]);
if ( fcent < 0 ) { fcent = fcent_np; }
if ( fcent > 100 ) { fcent = 100; }
```

Binning of the data:

The data has been divided in to bins of centrality, eta, and vertex. These bins are common for all the steps in determining the resolution correction.

Centrality Bins

Bin	***0***	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***
Min	0%	10%	20%	35%	40%	50%	60%	70%	80%	90%
Max	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%

Eta Bins

Bin	***0***	***1***	***2***	***3***	***4***	***5***	***6***	***7***
Min	-4.0	-3.5	-3.0	-2.5	-2.0	-1.5	-1.0	-0.5
Max	-3.5	-3.0	-2.5	-2.0	-1.5	-1.0	-0.5	0.0

Bin	***8***	***9***	***10***	***11***	***12***	***13***	***14***	***15***
Min	0.0	0.5	1.0	1.5	2.0	2.5	3.0	3.5
Max	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0

Vertex Bins

Bin	***0***	***1***	***2***	***3***	***4***	***5***	***6***	***7***	***8***	***9***
Min	-50	-45	-40	-35	-30	-25	-20	-15	-10	-5
Max	-45	-40	-35	-30	-25	-20	-15	-10	-5	0

Bin	***10***	***11***	***12***	***13***	***14***	***15***	***16***	***17***	***18***	***19***
Min	0	5	10	15	20	25	30	35	40	45
Max	5	10	15	20	25	30	35	40	45	50

Procedure to Define and Correct the Reaction Plane:

The reported reaction plane in the Au-Au 200 GeV dsts is not fully corrected. There are two corrections that need to be done: 1) re-centering of the Q values and 2) the flattening of the distribution.

Step 1: Re-centering the Q values:

Definition of Q:

The Q values are the sum of cosine and sine terms over a ring element. Here we will define Q_x and Q_y as

$$Q_x = \sum_i w_i \cos(n\phi_i) \quad (1)$$

$$Q_y = \sum_i w_i \sin(n\phi_i) \quad (2)$$

This sum is done for each element in a ring. Silicon rings 1a,c-g are considered individual rings for this correction.

Step 1a: Generating Values:

The code used to generate these value is called

For data: Flat_Selector.C, MeasQ_Selector.h, Processor_MeasQ.C

For Sim: MeasQ_Sim_Selector.C, MeasQ_Sim_Selector.h, Processor_MeasQ_Sim.C

The raw values of Q have been saved into the dsts. The eta bin for each individual ring is determined as is the centrality bin for the event. Histograms are filled with the Q values. Here is an example for si ring 1a:

```
if ( si1a_etai >= 0 ) {
    hQ2x_Si1a[centi][si1a_etai]->Fill(FL_fSi1aSumCos[0][1]);
    hQ2y_Si1a[centi][si1a_etai]->Fill(FL_fSi1aSumSin[0][1]);
}
```

Each of these histograms are then saved to a root file. The code will then get the histograms from the files and save the mean of the histograms to an ascii file.

Step 1b: Applying the Corrections:

The Q values are loaded into arrays. Here is the example for si ring 1 a.

```
Int_t ci,ei;
Float_t Qx2,Qy2;

ifstream si08file("/direct/brahms+u/ebj/Analysis/ResCor/params/Si1aQCor.dat", ios::in);
while(si08file) {
  si08file>>ci>>ei>>Qx2>>Qy2;
  AvgQx2_Si[8][ci][ei] = Qx2;
  AvgQy2_Si[8][ci][ei] = Qy2;
```

```
}
si08file.close();
```
The index [8] is a convention used in the code where ring 8 corresponds to 1a, 9 to 1b, 10 to 1c and so on.

The values are then applied.

```
 Float_t Si1aX2 = FL_fSi1aSumCos[0][1];
 Float_t Si1aY2 = FL_fSi1aSumSin[0][1];

 if ( si1a_etai >= 0  ) {
   Si1aX2 -= AvgQx2_Si[8][centi][si1a_etai];
   Si1aY2 -= AvgQy2_Si[8][centi][si1a_etai];
 }
```
Here centi and si1a_etai are the centrality and eta bins for si ring 1a, respectively.

Step 1c: Redefining the Reaction Plane:

Since this correction is applied to the Q values of the reaction plane calculation, the value of the reaction plane needs to be calculated. The first step is to sum together the Q values for the combination planes. For the silicon rings, this is just summing together the Qx and Qy values for rings 1a,c-g. For the silicon combination plane (Si Ring 1a,c-g, Flow Ring 2, Flow Ring 3), the normalized sum is determined.

$$Q_x^{combo} = \sum_i \frac{Q_{xi}}{\sqrt{Q_{xi}^2 + Q_{yi}^2}} \qquad (3)$$

$$Q_y^{combo} = \sum_i \frac{Q_{yi}}{\sqrt{Q_{xi}^2 + Q_{yi}^2}} \qquad (4)$$

To ensure the same selection criteria is made for this new value of the reaction plane as what is stored in the dst, the reaction plane is only determined to events where the value of the plane in the dst has a value between $-\pi/2$ to $\pi/2$. As an example, here is the new value of the reaction plane for the silicon rings 1a, c-g.

```
if ( TMath::Abs(fPsi_2Def[0]) < TMath::PiOver2() ) {
        fPsi_2[0] = 0.5*TMath::ATan2(Si1Y2,Si1X2);
}
fPsi_2Def[0] = FL_fSiPsi[0][1]
```
Si1Y2 and Si1X2 are the sum of the Qy and Qx values for all the silicon rings.

Step 2: Flattening the Reaction Planes:

Definition of the Flattening Term:

The correction is a Fourier decomposition of the reaction plane, adding small components to the value giving it a flat distribution. The flattened reaction plane with correction terms is

$$\Psi^{flat} = \Psi + \sum_n \frac{2}{n+1}\{\langle \cos[(n+1)\Psi]\rangle \sin[(n+1)\Psi] - \langle \sin[(n+1)\Psi]\rangle \cos[(n+1)\Psi]\} \quad (5)$$

Here n is a harmonic of the Fourier decomposition and is taken out to the 20th term. The averages inside of the sum are averaged over all events within a centrality bin.

Step 2a: Determining the Flattening Coefficients:

The code used to generated these values are

For data: Flat_Selector.C, Flat_Selector.h, Processor_Flat.C

For Sim: Flat_Sim_Selector.C, Flat_Sim_Selector.h,Processor_Flat_Sim.C

The Q values are first corrected, and the new reaction planes are determined. The values of $\cos[(n+1)\Psi]$ and $\sin[(n+1)\Psi]$ fill a separate histogram for each reaction plane, centrality bin, and n term. The only issue is that the expansion is valid for the first harmonic, Ψ_1. This is circumvented by distributing Ψ_2 over 2π, by adding or subtracting π to the reaction plane from every other event.

The histograms are stored in a root file, and the means of the histograms are then stored in ascii files.

Step 2b: Applying the Flattening Coefficients:

To account for the fact that equation 5 and the saved flattening coefficients are valid for only the first harmonic, for each reaction plane less than zero and event number that is even, π is added to the reaction plane value. If the reaction plane value is greater than or equal to zero and the event number is even, then π is subtracted from the reaction plane value. The coefficients are then added to the reaction plane as described by equation 5. Then the reaction plane distribution is restricted back to $-\pi/2$ to $\pi/2$ by either adding or subtracting π accordingly.

Determining the Resolution Correction:

The first step in this process is to determine the resolution correction for several rings using the standard three ring method. This can be done for a limited range in h for Si Ring 1a,c-g, Flow Ring 2, Flow Ring 3, Tile Ring 1, the large left BB tubes, and a combination of the silicon and flow rings. The resolution correction is determined over eta for the individual rings, but over vertex for the silicon combination plane. The values determined from the data is compared to the true reaction plane resolution correction from the simulated data. For the centralities and rings where the values are in agreement, the true reaction plane correction from the simulated data is used for the parameterization of the analysis.

Step 1: Resolution Correction for Individual Rings

The Q values are first re-centered and the reaction planes are flattened.

To minimize autocorrelations, the eta ranges for the detectors used for each reaction plane needs to be determined. The eta range is smeared in vertex due to the vertex resolution, and a ban of 0.2 units in eta are appended to either side of the range for all vertices. The eta limits for each ring are loaded from three different files for each ring type.

SiEtaLimitParams_*.dat ----- Silicon Rings
TiEtaLimitParams_*.dat ----- Tile Rings
FlEtaLimitParams_*.dat ----- Flow Rings
BBLEtaLimitParams_*.dat ----- BBL Rings
The * can be
5mm for TPM1 vertices
3cm for BB vertices
4cm for ZDC vertices.

These files hold the parameters for 9 degree polynomials that represent that eta min and max of the detector as a function of vertex. A 0.2 units in eta ban needs to be added to these functions. When a vertex and vertex type are determined, the appropriate eta limits are then defined.

The resolution correction for ring a from three independent rings a, b and c is

$$ResCor_a = \sqrt{\frac{\langle \cos[2(\Psi_a - \Psi_b)] \rangle \langle \cos[2(\Psi_a - \Psi_c)] \rangle}{\langle \cos[2(\Psi_b - \Psi_c)] \rangle}} \qquad (6)$$

The averages are over all events.

Once the reaction planes are determined, histograms are filled that hold these averages. The histograms are only filled when both reaction planes are defined and when the detectors used to define the reaction planes are not overlapping in eta. A histogram is defined for each combination in the eta bin space for each detector ring and for each centrality bin. Note that $\langle\cos[2(\Psi_a-\Psi_b)]\rangle = \langle\cos[2(\Psi_b-\Psi_a)]\rangle$, so only one histogram is defined to determine the average. When looking for the resolution correction for the

silicon combination plane (flow rings and silicon rings), the histograms where divided into vertex bins.

When working with the simulated data, the true reaction plane resolution correction is determined directly from the true reaction plane. For ring a, this is defined as

$$ResCor_a = \langle \cos[2(\Psi_a - \Psi_R)] \rangle \quad (7)$$

The average is determined from filling a histogram for each ring for a number of eta (or vertex) and centrality bins.

After the histograms are filled and stored to a root file, the data is loaded into arrays to be further processed. The first step is to determine the resolution for each individual ring (Si Ring 1, Flow Ring 2, Flow Ring 3, Tile Ring 1, and Left Large BB Ring). The res correction for each ring was determined from 6 different combinations of rings. This is not to say each combination was used since some combinations used rings that overlapped in eta. The averages in equation 6 were determined from the mean of the histograms, and the error on the resolution correction was determined based on the rms of the histograms. Furthermore, only combinations where all three averages were determined from at least 10,000 events was used.

At this point, each ring has up to 6 resolution correction values for a eta and centrality bin. The correction values were then averaged to give a final correction. The average was weighted using the errors determined from the previous step. Only values that are greater than 0, less than 1, and have an error of less than 100%, are used in the final average. This final average is then stored in a ascii file. The file names are
SiAvgResCor.dat
TileAvgResCor.dat
Fl2AvgResCor.dat
Fl3AvgResCor.dat
BB4AvgResCor.dat
It was found out for the tile ring the combination between the flow ring 3 and the bb counters was the only one that gave a reasonable correction. Therefore, TileAvgResCor.dat only holds the res corrections from this combination.

The silicon combination reaction plane res correction is also determined in a similar manner, but the data is binned in vertex. There is only one combination of rings (BB and Tile) that can be used to determine the reaction plane.

The sim data files will have values for the corrections measured in the exactly same manner as it is done with the data and the true resolution.

Step 2: Comparing the Resolution Correction from Data and Simulation:

The following figures are the resolution corrections for various rings for different centrality bins. The figures are labeled well so there should be little confusion what is being plotted. A discussion of the results will follow the plots.

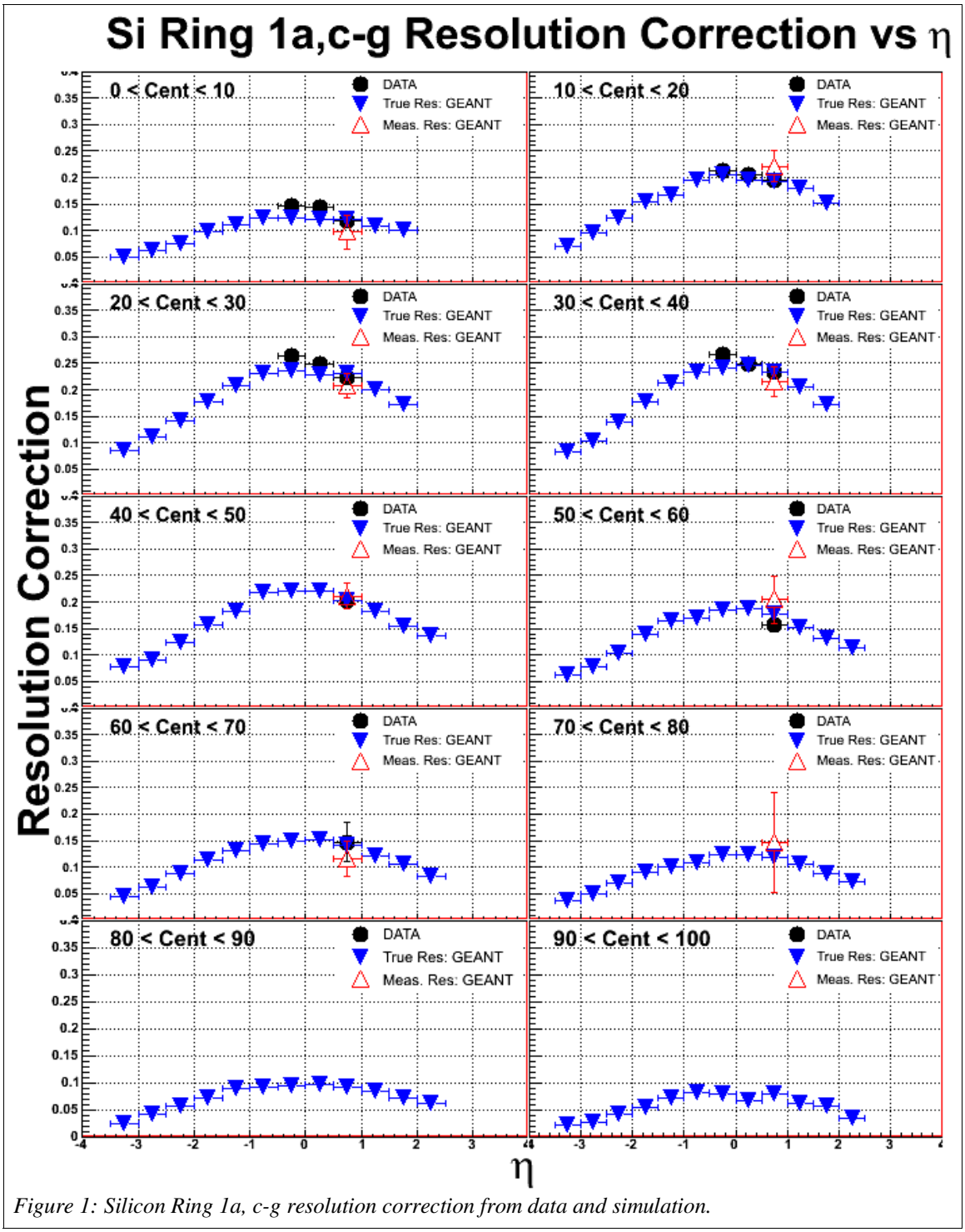

Figure 1: Silicon Ring 1a, c-g resolution correction from data and simulation.

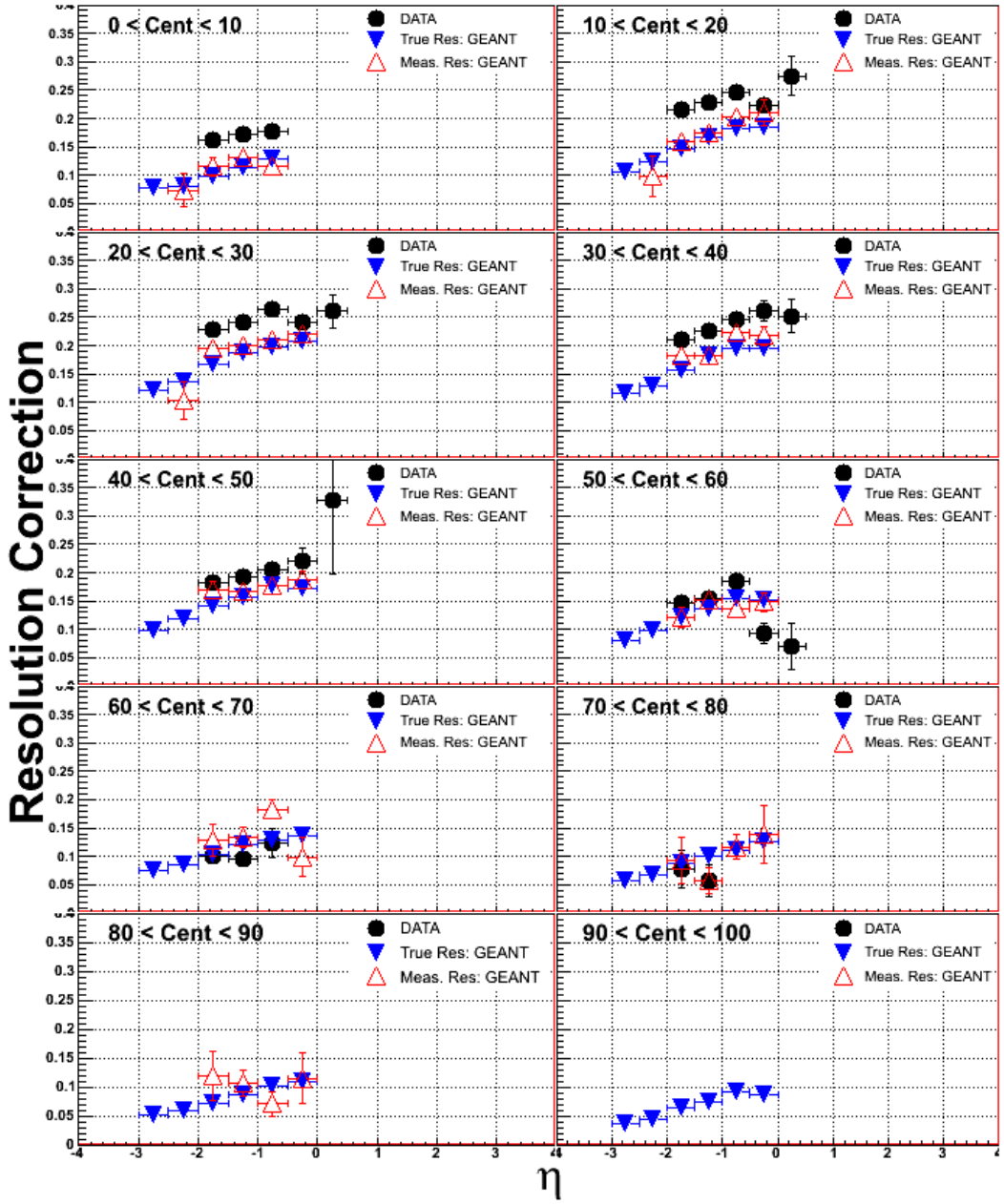

Figure 2: Tile Ring 1 resolution correction from data and simulation.

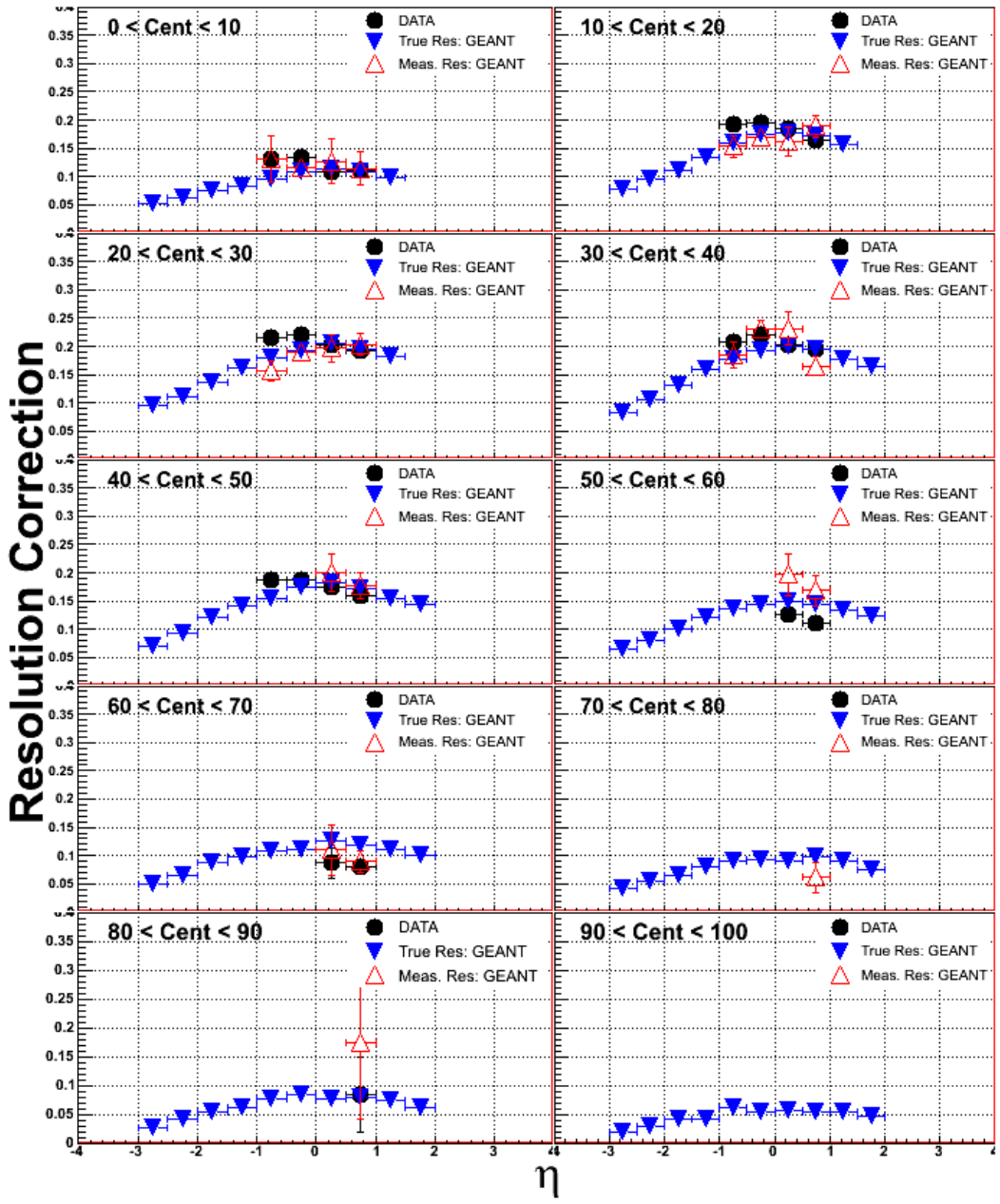

Figure 3: Flow Ring 2 resolution correction from data and simulation.

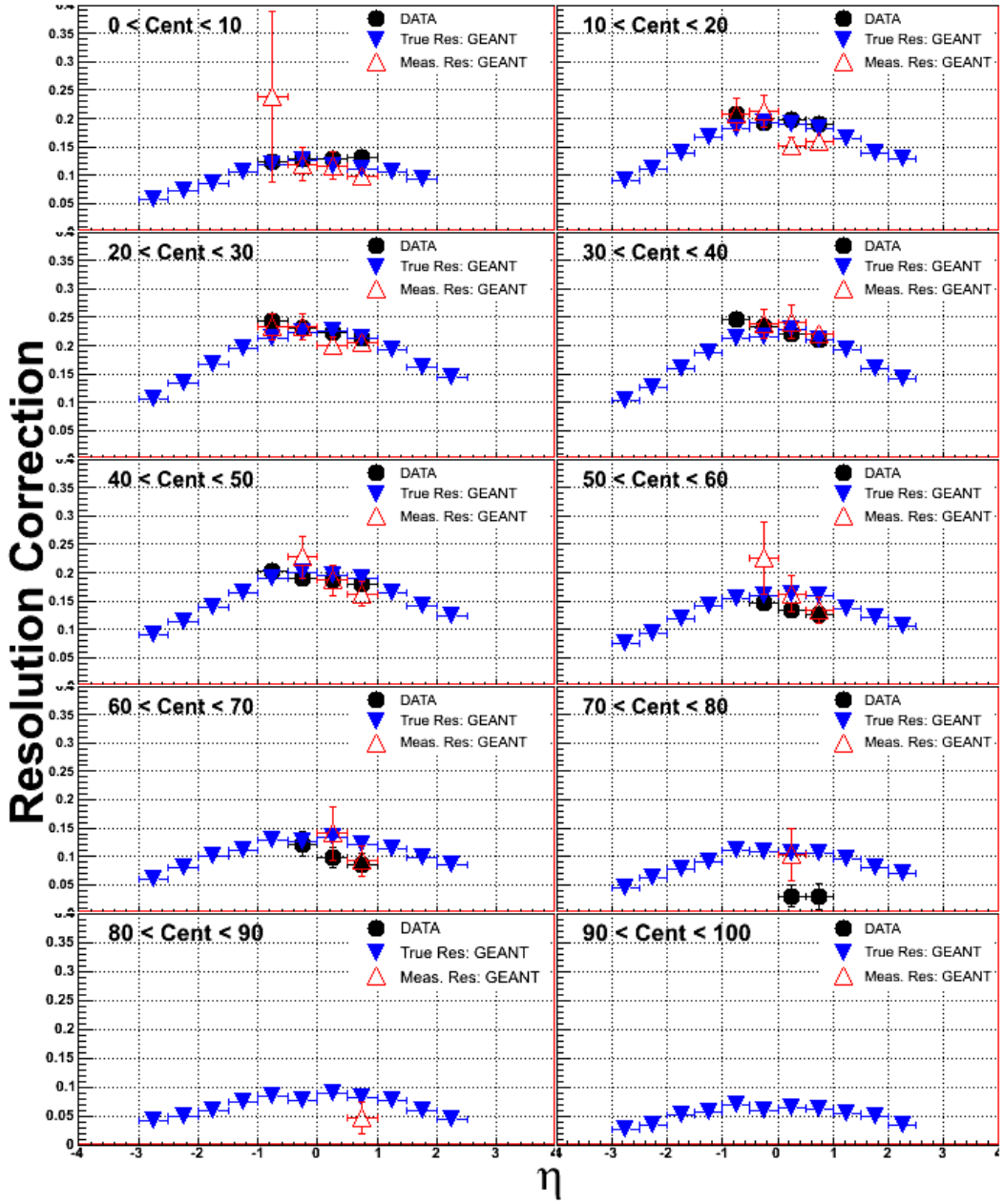

Figure 4: Flow Ring 3 resolution correction from data and simulation.

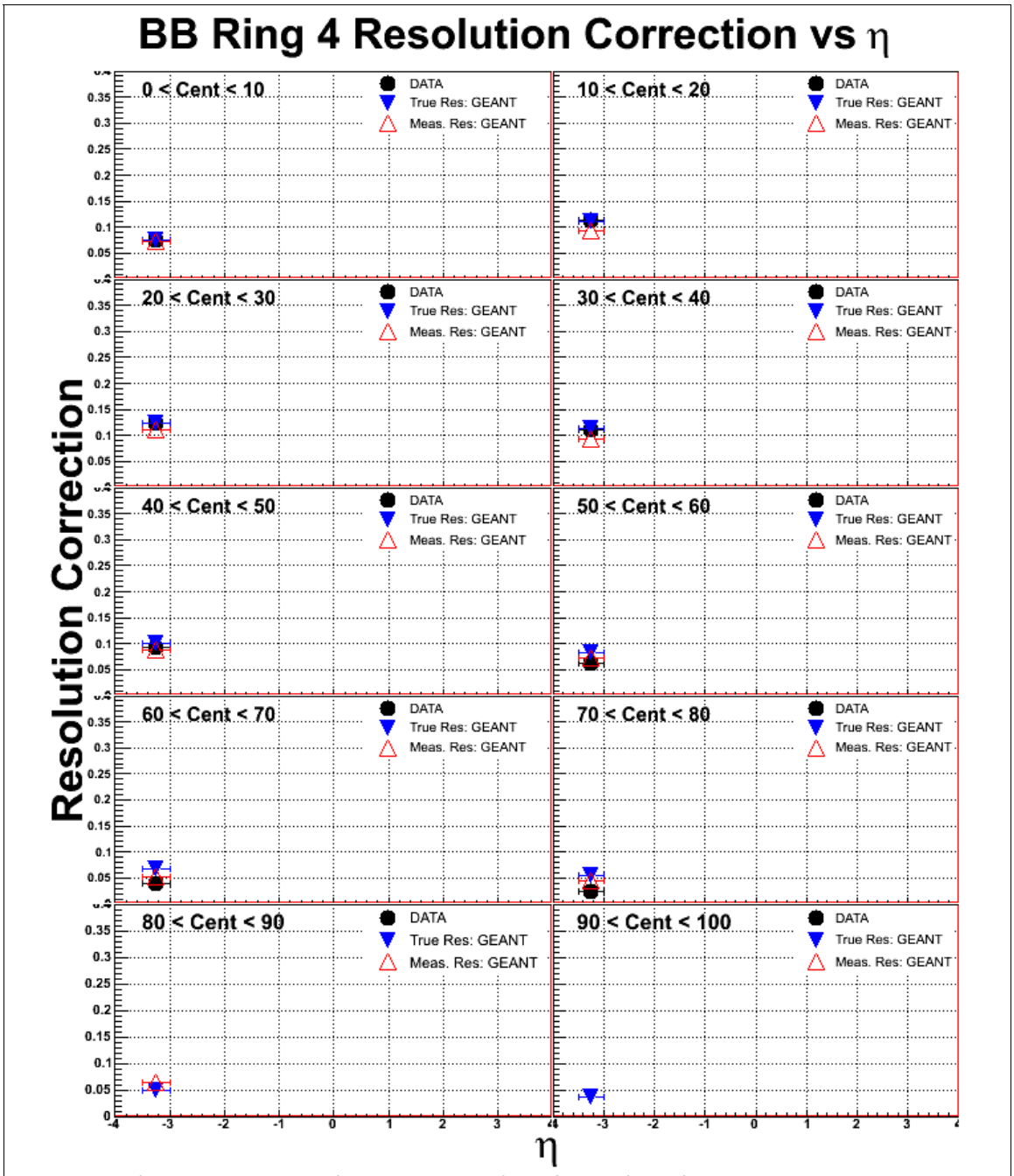

Figure 5: Left Large BB ring resolution correction from data and simulation.

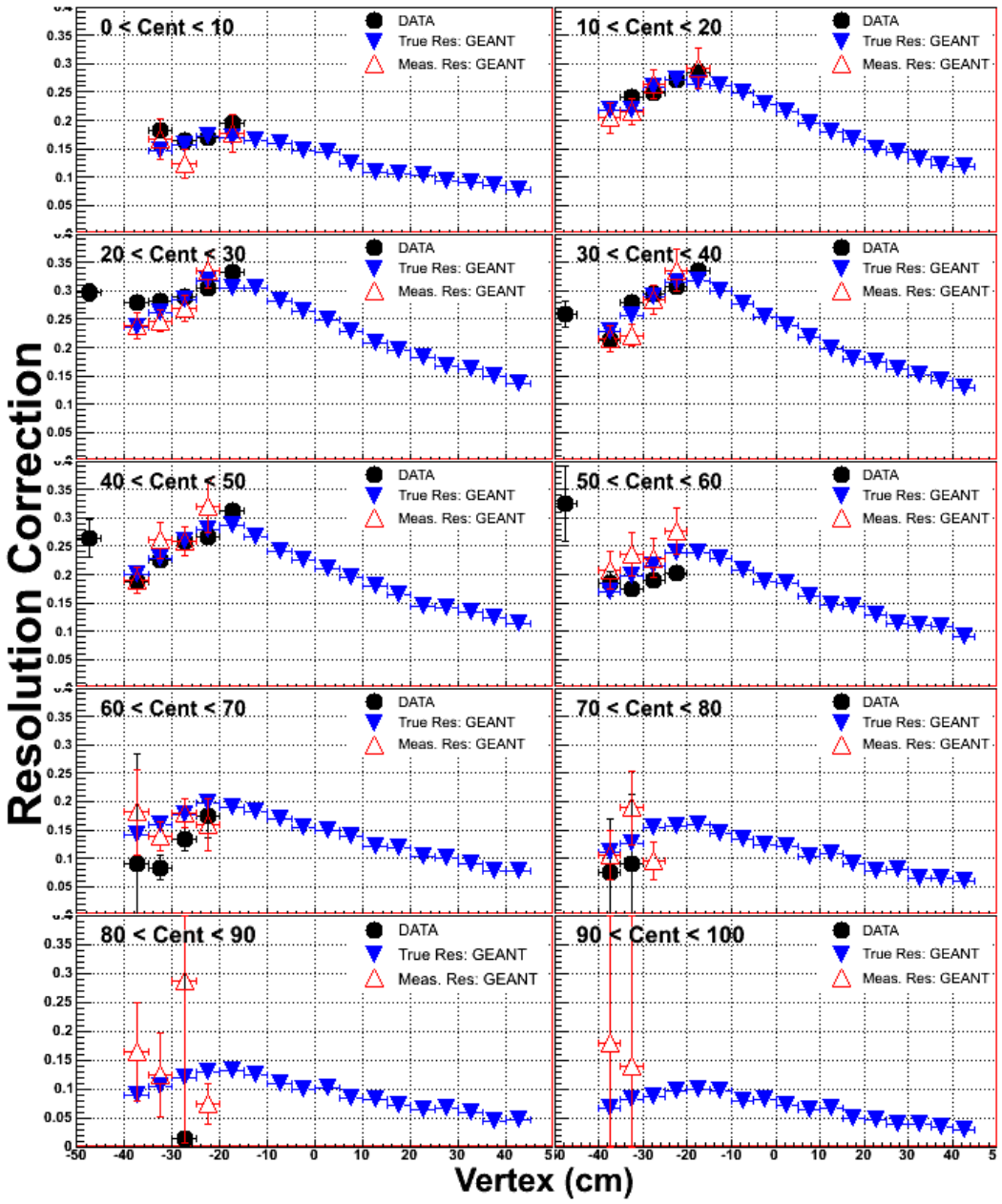

Figure 6: Silicon Combo Ring (Si Ring 1a, c-g, Flow Ring 2, Flow Ring 3) resolution correction from data and simulation.

The first thing is that the tile resolution correction is not in agreement with the true resolution from the simulation, and therefore, the true resolution correction for the tile ring from the simulation can not be used. The other detector systems are in excellent agreement for the top 50% central events. Deviations for 50% -100% central events are predicted to be due to the simulation's incorrect extrapolation of the multiplicity over these centralities. Therefore, the resolution correction will be limited to the top 50% central events.

Step 3: Parameterize the Resolution Correction for the Best Ring Combinations:

The two best ring combinations are Tile Ring 1 and a system with all the other rings (SiBB). The resolution from Tile Ring 1 is determined from the data, while the resolution correction from SiBB is determined from the simulation. The correction from Tile Ring 1 can be determined over the centrality range of 0 – 70% but is not done here.

The first step is to fit the data to some polynomial. These fits are shown in figures 7 and 8. A table of the functions and parameters used will be given at the end of this section. The parameters of these fits are then parameterized with separately as functions over centrality, and these are shown in figures 9 and 10. The tile data is presented as a function of eta, and to give the reader an idea over the vertex range the parameterization is valid, the corrections are calculated over vertex, as shown in figure 11.

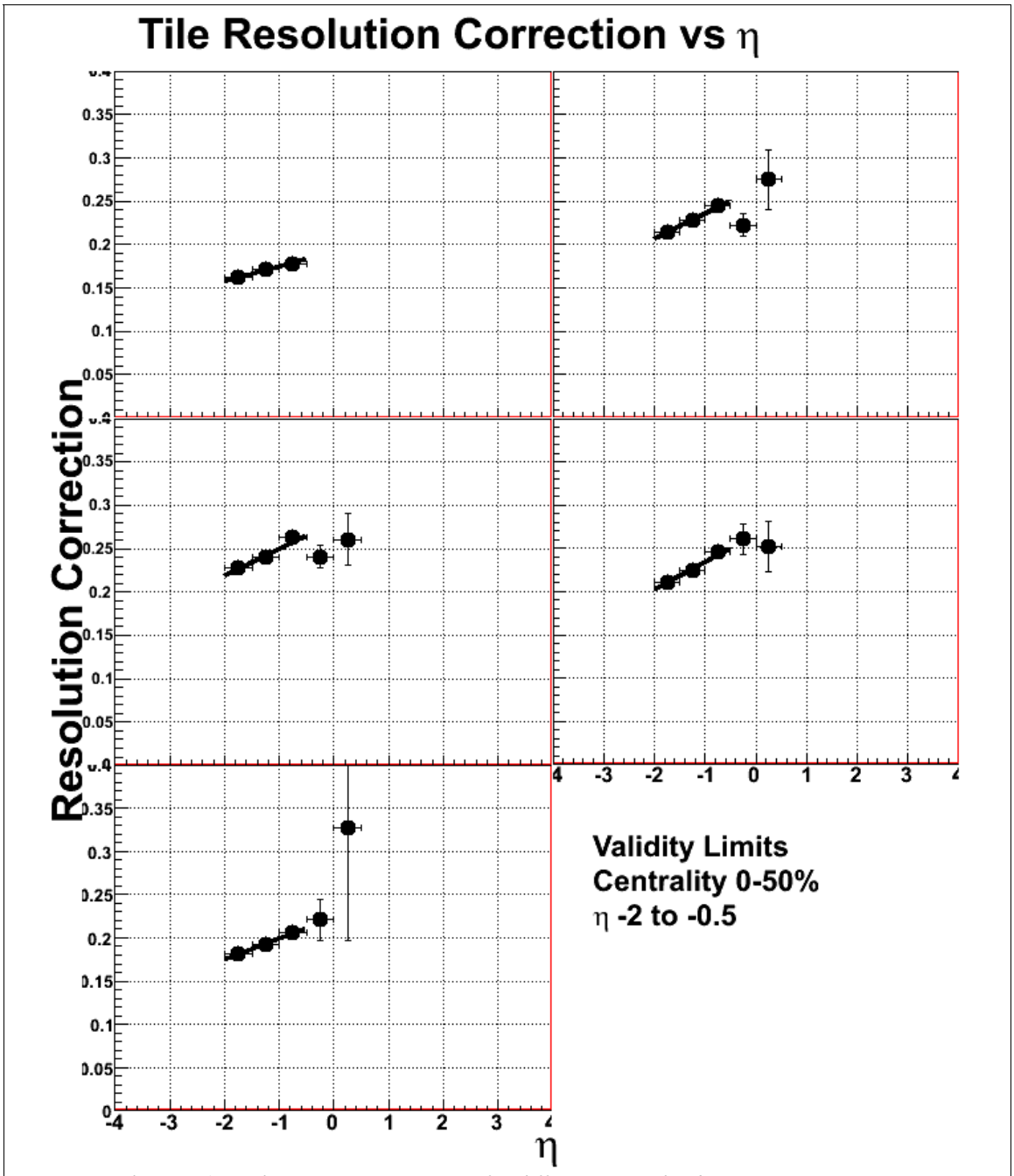

Figure 7: Tile Ring 1 resolution correction vs eta for different centrality bins.

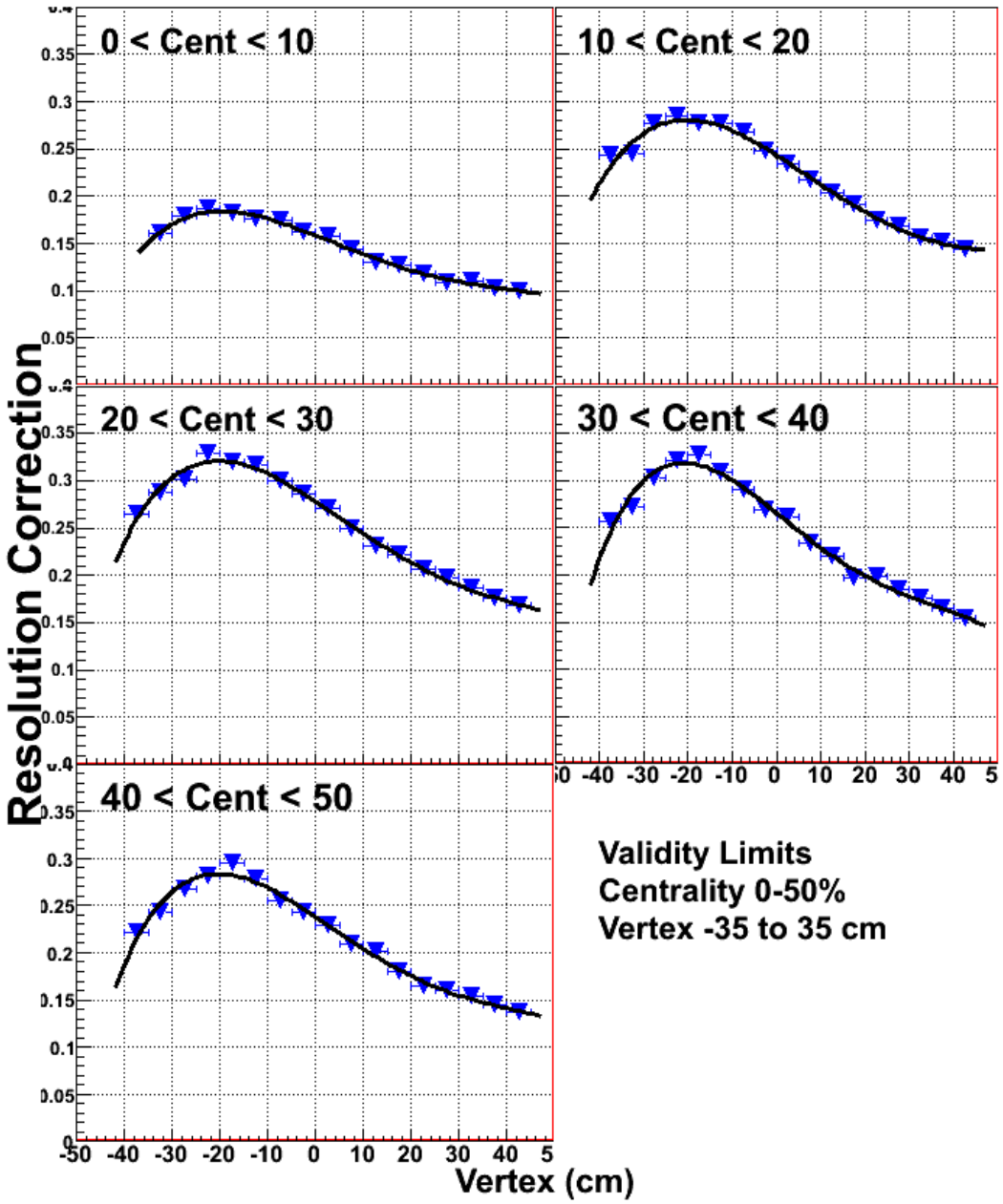

Figure 8: SiBB ring system res correction vs vertex for the top 5 centrality bins.

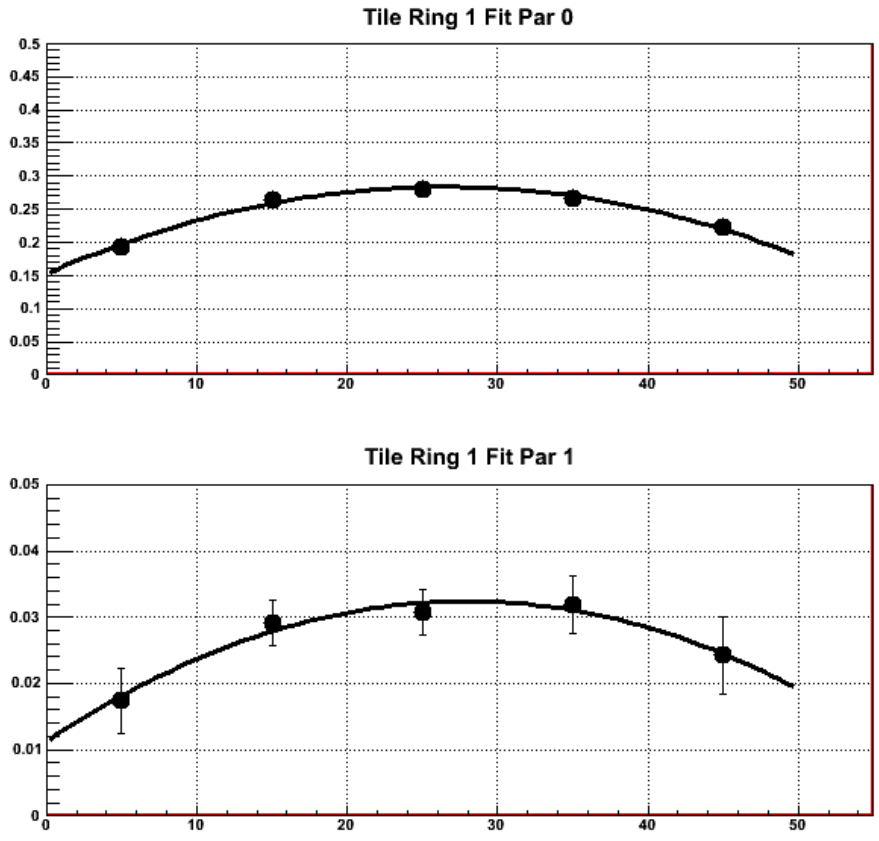

Figure 9: Tile Ring 1 fit parameters as a function of centrality.

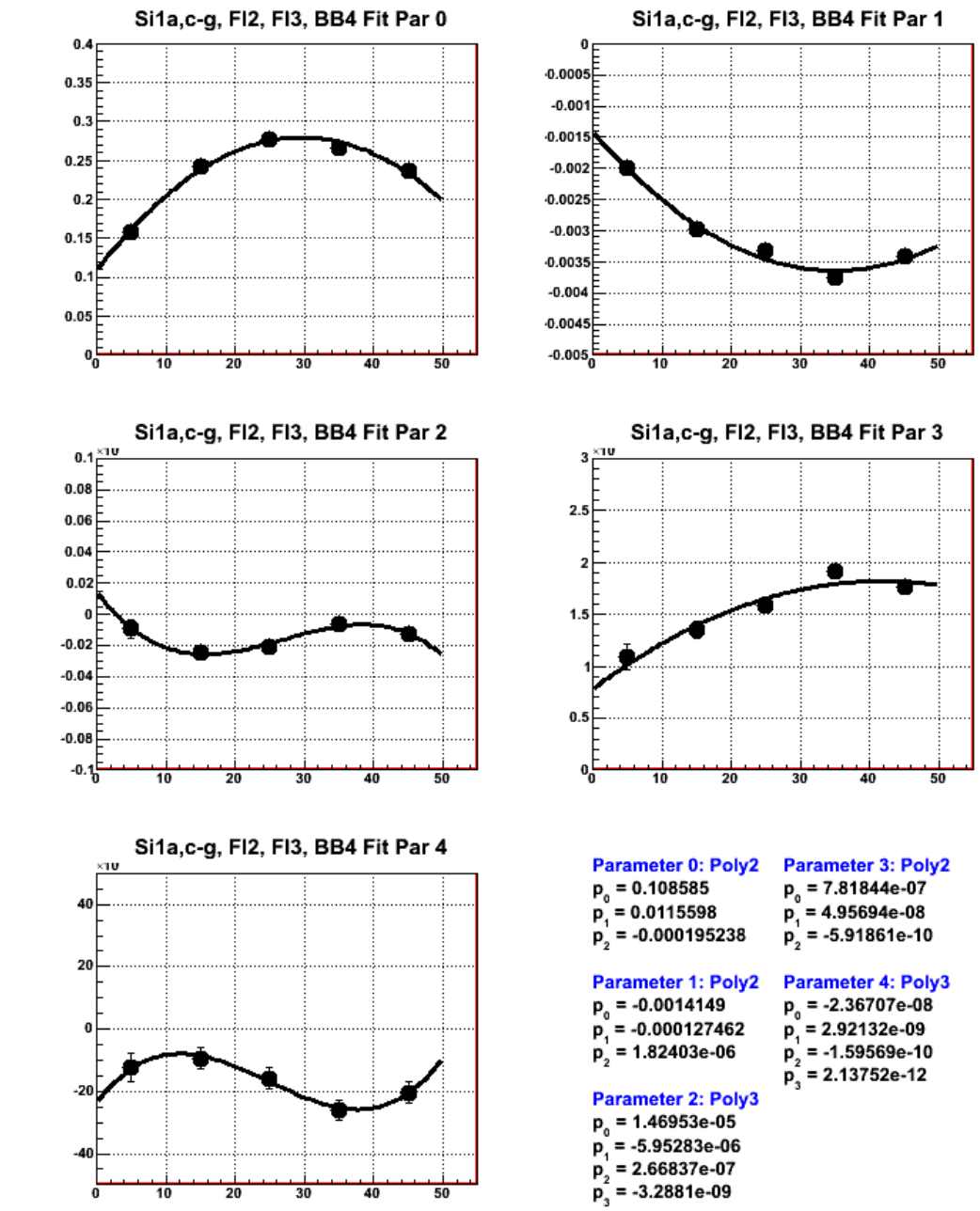

Figure 10: SiBB fit parameters as a function of centrality.

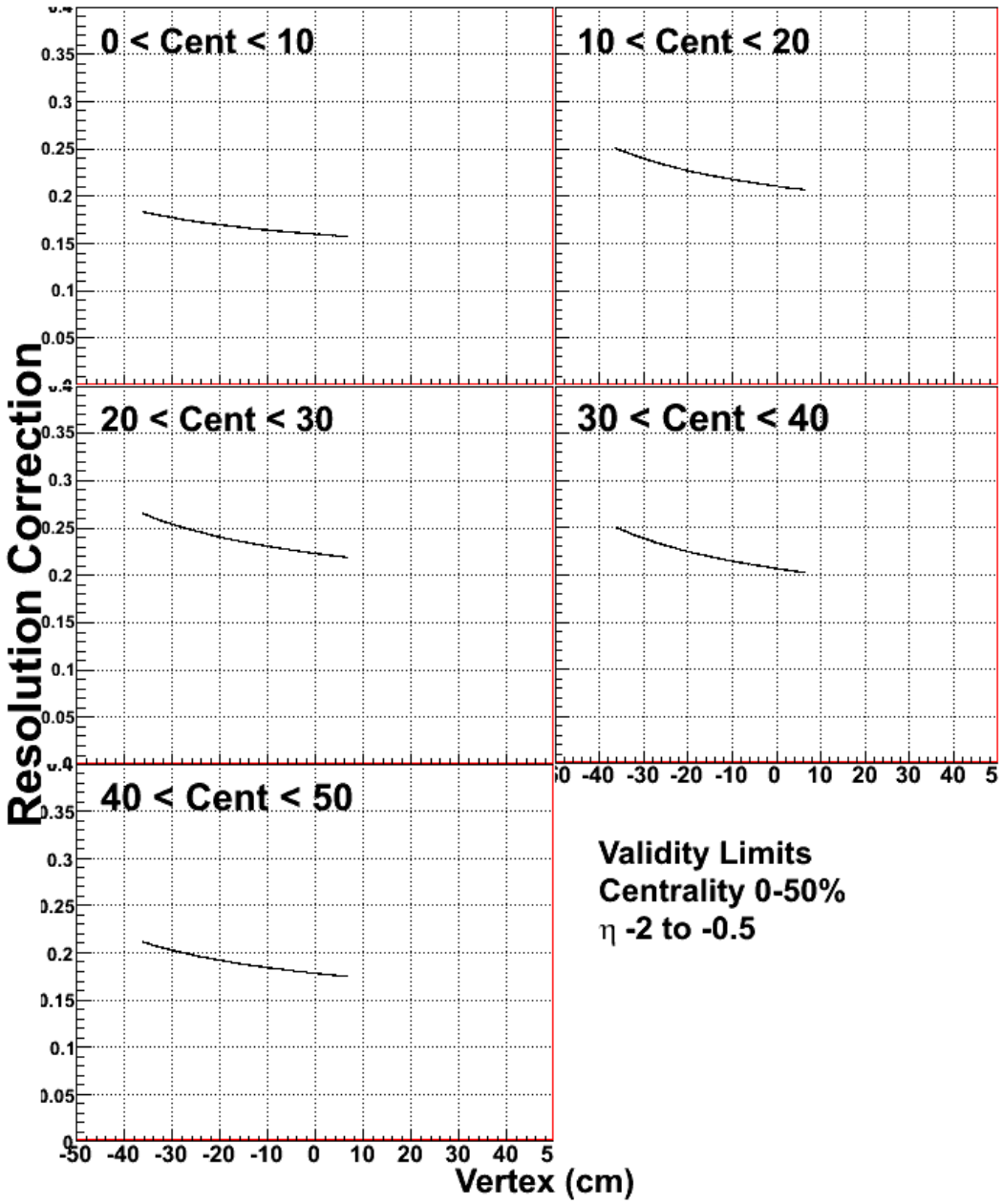

Figure 11: Tile Ring 1 res corrections vs vertex for 0-50% most central events.

Tile Ring 1 Functions and Parameters

These values are only valid for 0-50% central events and $-2.0 < \eta < -0.5$.

r = radius = 13.750 cm

z = nominal z position = -43.411 cm

$$\eta(v_z) = -\log\left(\tan\left(\frac{1}{2}\tan^{-1}\left(\frac{r}{z - v_z}\right)\right)\right) \quad (8)$$

$$ResCor(v_z, cent) = p_0(cent) + p_1(cent)\cdot\eta(v_z) \quad (9)$$

$$p_0(cent) = a_0 + a_1\cdot cent + a_2\cdot cent^2 \quad (10)$$

$$p_1(cent) = b_0 + b_1\cdot cent + b_2\cdot cent^2 \quad (11)$$

$a_0 = 0.152401$

$a_1 = 0.00992114$

$a_2 = -0.000187473$

$b_0 = 0.0112493$

$b_1 = 0.00151607$

$b_2 = -2.71579\text{x}10^{-5}$

SiBB Functions and Parameters

These values are only valid for 0-50% central events and -35 < vertex < 35 cm

$$ResCor(v_z, cent) = p_0(cent) + p_1(cent)\cdot v_z + p_2(cent)\cdot v_z^2 + p_3(cent)\cdot v_z^3 + p_4(cent)\cdot v_z^4 \quad (12)$$

$$p_0(cent) = a_0 + a_1\cdot cent + a_2\cdot cent^2 \quad (13)$$

$$p_1(cent) = b_0 + b_1\cdot cent + b_2\cdot cent^2 \quad (14)$$

$$p_2(cent) = c_0 + c_1\cdot cent + c_2\cdot cent^2 + c_3\cdot cent^3 \quad (15)$$

$$p_3(cent) = d_0 + d_1\cdot cent + d_2\cdot cent^2 \quad (16)$$

$$p_4(cent) = e_0 + e_1\cdot cent + e_2\cdot cent^2 + e_3\cdot cent^3 \quad (17)$$

$a_0 = 0.108585$

$a_1 = 0.0115598$

$a_2 = -0.000195238$

$b_0 = -0.0014149$

$b_1 = -0.000127462$

$b_2 = 1.82403\text{x}10^{-6}$

$c_0 = 1.46953\text{x}10^{-5}$

$c_1 = -5.95283\text{x}10^{-6}$

$c_2 = 2.66837\text{x}10^{-7}$

$c_3 = -3.2881\text{x}10^{-9}$

$d_0 = 7.81844\text{x}10^{-7}$

$d_1 = 4.95694\text{x}10^{-8}$

$d_2 = -5.91861\text{x}10^{-10}$

$e_0 = -2.36707\text{x}10^{-8}$

$e_1 = 2.92132\text{x}10^{-9}$

$e_2 = -1.59569\text{x}10^{-10}$

$e_3 = 2.13752\text{x}10^{-12}$

Mapping Where to Use the Correction in Vertex:

The last concern is that the eta acceptance of the spectrometers overlaps the acceptance of the detectors used to determine the reaction plane. The following plots are the accepted tracks for the spectrometers at different angles. From these plots, it is clear that the standard method using the combination SiBB reaction plane for vertices greater than zero and Tile Ring 1 for vertices less than zero will remove any autocorrelations. One can be more aggressive for the forward spectrometer, but the selection must be done carefully to avoid any autocorrelations.

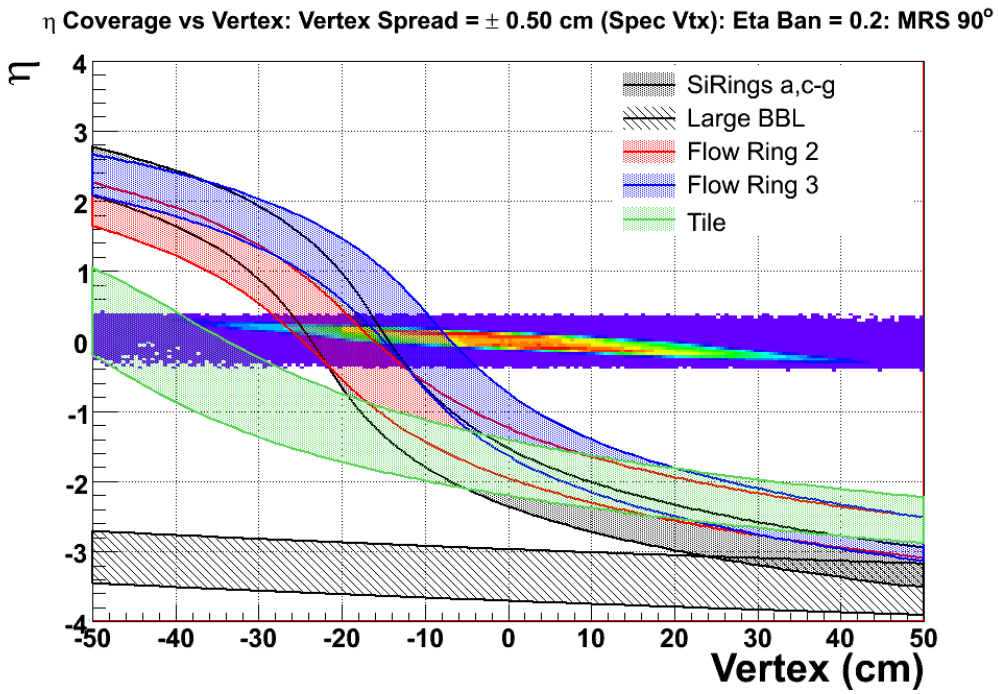

Figure 12: Eta Acceptance of tracks for the MRS at 90 degrees.

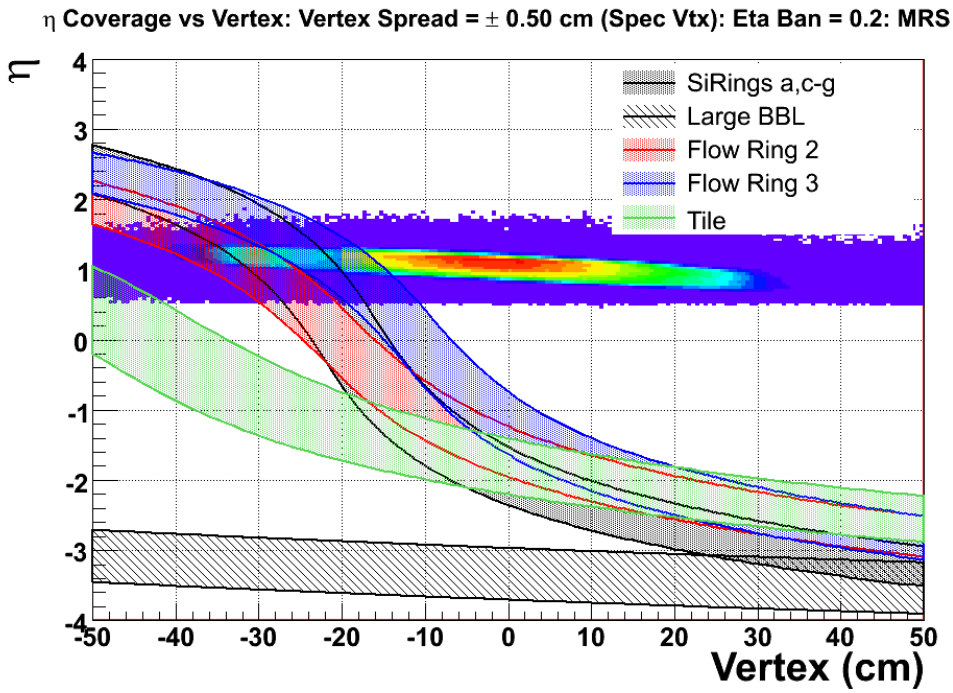

Figure 13: Eta Acceptance of tracks for the MRS at 40 degrees.

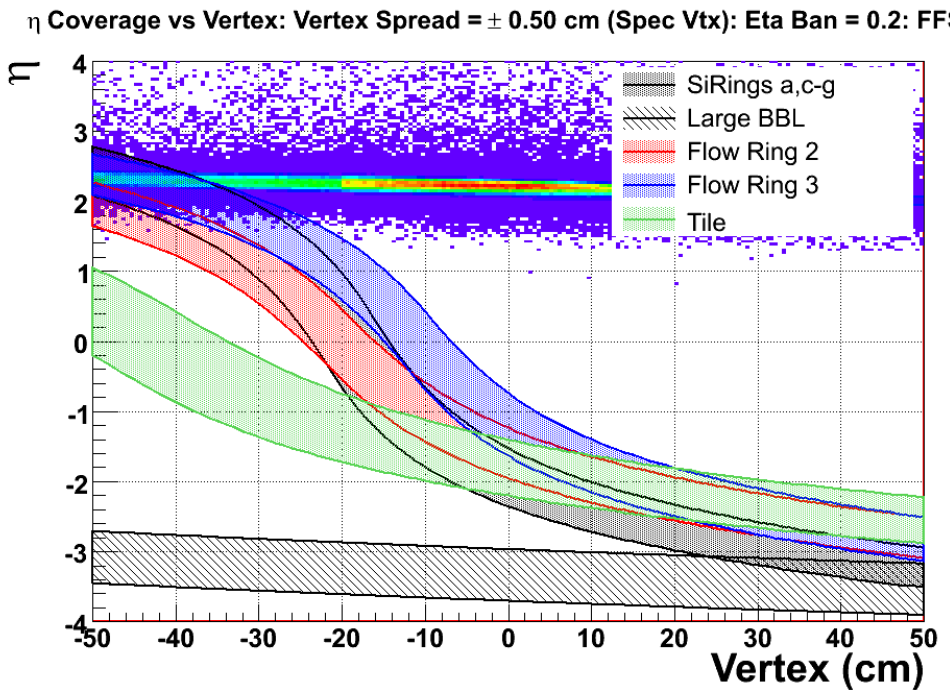

Figure 14: Eta Acceptance of tracks for the FSS at 12 degrees.

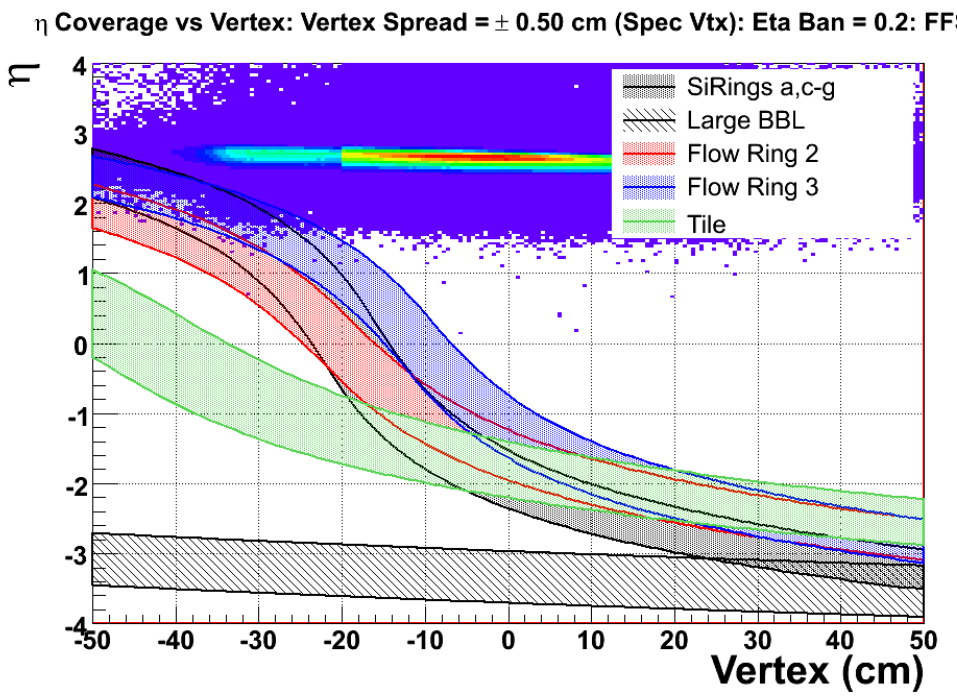

Figure 15: Eta Acceptance of tracks for the FFS at 8 degrees.

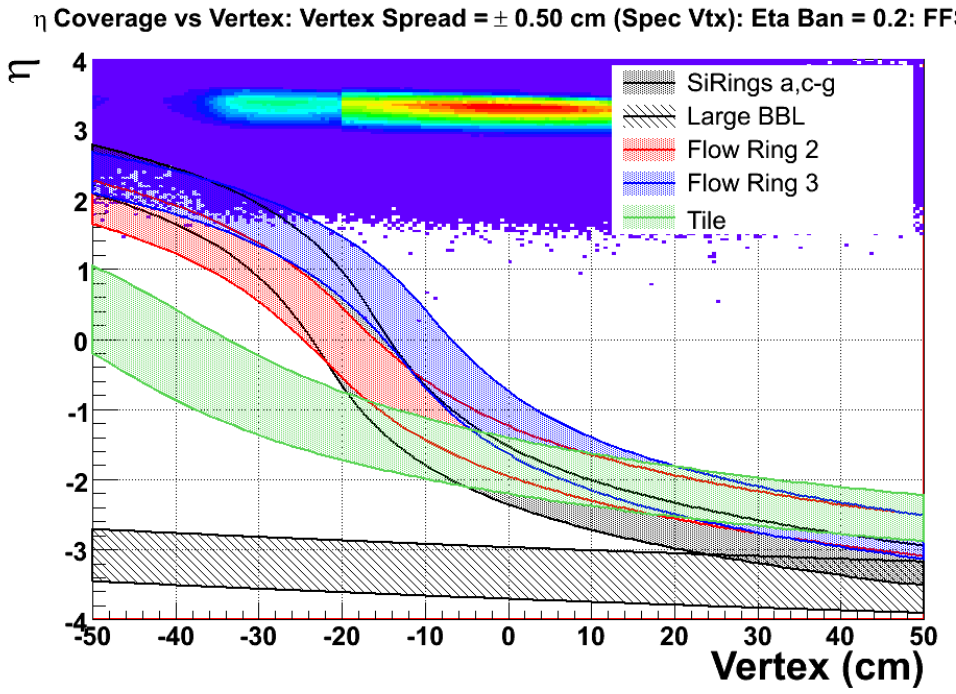

Figure 16: Eta Acceptance of tracks for the FFS at 4 degrees.