CDF Note 7928

Resolution and Simulation of the EMTiming System

Max Goncharov, David Toback, Peter Wagner, and Vyacheslav Krutelyov*

Texas A&M University

(Dated: May 8, 2006)

In this note we present a summary of the corrections used in the calibration of the EMTiming system and their impact on the resolution of the system. We find a timing resolution of 0.64 ± 0.01 ns and 0.68 ± 0.01 ns for high energy CEM and PEM electrons respectively after all corrections and assuming the correct vertex has been selected. The efficiency for an in-time requirement of 3σ for this case is 99.8%. In addition we compare the data to our Monte Carlo simulation of the system that can be found in the *CalorTimeMods* package. It handles all event and particle types as in real data and it is constructed to reproduce the time of arrival of prompt, isolated electrons from W's. It takes into account physics effects like vertex position and event t_0, and corrects for energy slewing effects as well as channel-by-channel energy thresholds. We note that this package can be used in the search of long-lived neutral particles that decay to photons.

* now at University of California, Santa Barbara

I. INTRODUCTION

The installation of the EMTiming system was completed in Fall 2004 [1, 2], and is now fully commissioned and calibrated using jet data as well as $W \to e\nu$ and $Z \to ee$ event samples. In this note we describe the resolution and simulation of the system and show that the corrected arrival time distribution in the EMTiming system, measured with a $W \to e\nu$ data sample with a correctly selected vertex, has a resolution of 0.64 ns (0.68 ns) for CEM (PEM) and provide detail on the corrections that produce it. We quantify each correction and its uncertainty and finally derive an intrinsic calorimeter timing resolution that is used in the simulation. Then we describe the functionality of the Monte Carlo simulation of the system in the order as it is implemented in the code. We show that it reproduces the time distributions from data such that it can be used in the search of long-lived neutral particles that decay to photons [3].

In order to understand the system we start with a description of the relevant hardware components. The EMTiming system uses a signal from the photo multipliers (PMTs) in the CEM and PEM subdetectors as input to a fixed height discriminator (ASD) [2]. The output of the discriminator triggers a TDC which writes the time of arrival into the data stream. There are a number of effects that contribute to the arrival time that are corrected for offline [4]. They are the:

- Start time of the TDC (offset, tower-by-tower)
- Intrinsic system resolution
- TDC readout in integer values
- Energy slewing from the ASD
- z_0 variation of the collision
- t_0 variation of the collision

The raw time as it comes out of the TDCs, $t_{\rm raw}$, is corrected for the energy slewing and the TDC start time, $t_{\rm slewing}$, for the z_0 variation of the vertex position (time of flight (TOF)-correction, $t_{\rm zcorr}$), and for the t_0 of the vertex, $t_{\rm event}$. The final corrected time is obtained by:

$$t_{\rm corr} = t_{\rm raw} - t_{\rm slewing} - t_{\rm event} + t_{\rm zcorr} \tag{1}$$

The corrections are applied such that ideally the corrected time for prompt particles is 0 ns. As we will see, after all these corrections the timing distribution of electrons from $W \to e\nu$ events in the CEM (PEM) is Gaussian with an RMS of 0.64 ns (0.68 ns) without tails if the vertex of the electron is correctly identified. For PEM electrons we see a second Gaussian with an RMS of 2.3 ns at a rate of approximately 10% due to vertex misidentification.

The uncertainties due to each effect are estimated using a Monte Carlo (MC) simulation. The corrections and the sample, vertex and event requirements are discussed in Section III, after describing the datasets in Section II. The contribution to the raw time of the RMS of each correction is listed in Table I and explained in Sections III A- III C. Finally in Section III D an intrinsic EMTiming resolution is derived using a $Z \to ee$ data sample and put into the MC "by hand" as this hardware effect is not simulated. This makes the argumentation for the correctness of the MC in this note circular in the sense that given the intrinsic resolution from data, the MC simulation reproduces the data results.

The EMTiming system is simulated using a Monte Carlo (MC) that is run, independently, after event generation and simulation with the *cdfSim* and production of *cdfsoft* release 6.1.2 [5]. Its purpose is mainly to reproduce the time of arrival of prompt, isolated particles, e.g. electrons from W's and Z's with a robustness that allows it to be used in the search of long-lived neutral particles that decay to photons [3]. Thus, our emphasis is not on a detailed reproduction of the hardware but on a correct reproduction of the data results found with $W \to e\nu$ samples generated with PYTHIA [6]. All the above effects are taken into account and their relative effects on the raw RMS and the final resolution are summarized together with the results from data in Tables I and II and visualized in Figs. 1, 2 and 3 for the CEM and PEM separately. In Section IV we give a brief overview of the MC simulation framework and the modules involved. Then in Section V we explain the effects and corrections in the order they are applied in the simulation and compare its results to the real data plots. In Section VI we show the efficiency for an in-time cut of 3σ on prompt particles and for an out-of-time cut on delayed particles as it will be applied in a search for long-lived particles decaying to photons [7], and the systematic uncertainties.

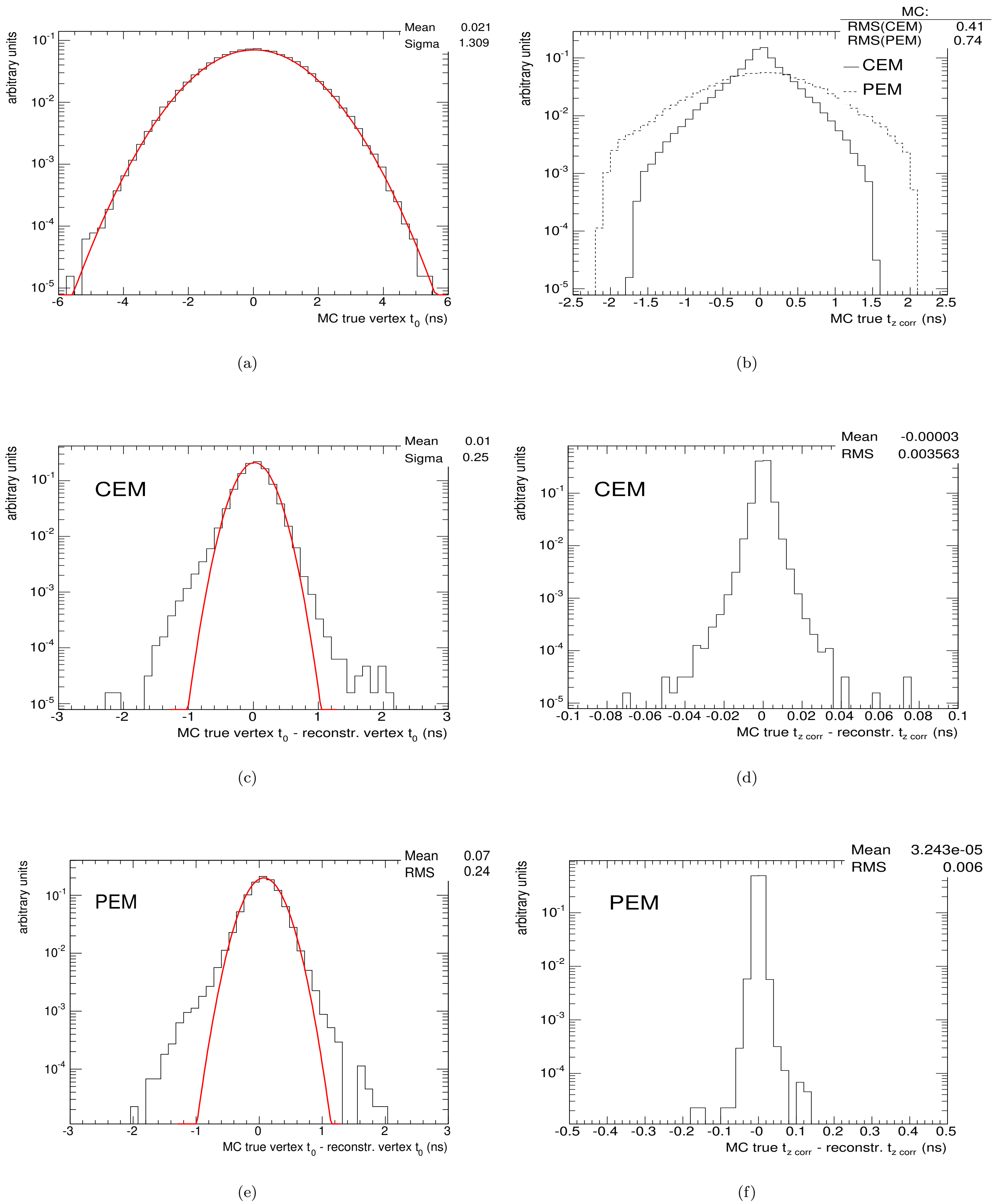


FIG. 1: A MC simulation of the vertex time and position effects that contribute to the RMS of the raw data and their impact on the final timing resolution using $W \to e\nu$ MC samples in CEM (Figs. (c) and (d)) and PEM (Figs. (e) and (f)). Figures (a) and (b) show the MC values for the collision time and the time distribution originating from the spread of the primary vertex in the z-direction, respectively. The contributions to the raw time RMS are 1.3 ns from the vertex t_0 and 0.41 ns (0.74 ns) from the TOF correction in the CEM (PEM). Similarly, Figures (c),(d) and (e),(f) show the difference between the reconstructed vertex information and the true values; a measure of the final resolution of these quantities for CEM and PEM respectively. The resolution of the vertex z measurement is negligible with 0.004 ns (0.006 ns), the resolution of the vertex t_0 is 0.25 ns (0.24 ns). The non-Gaussian tails are described in Section III B.

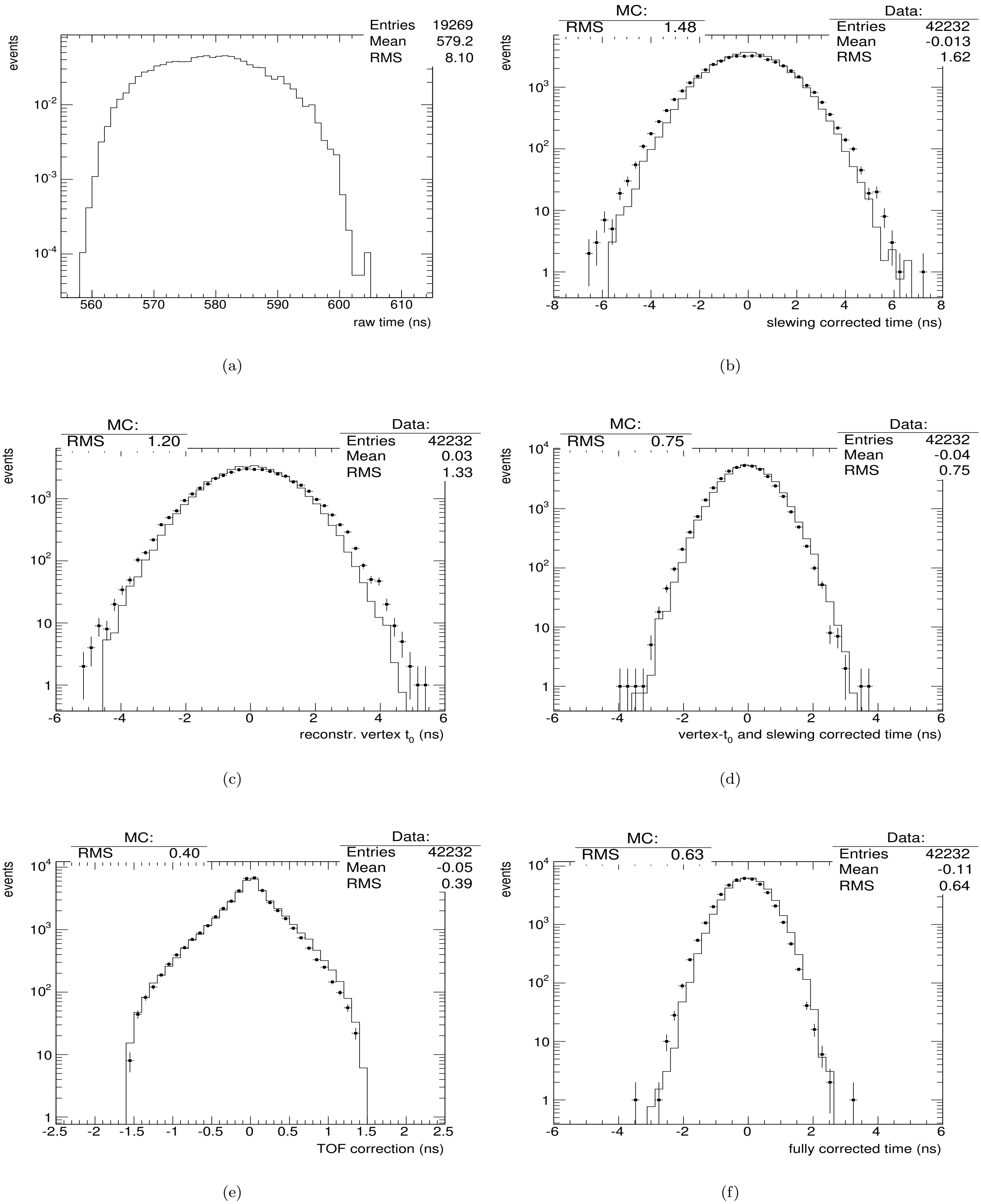


FIG. 2: A comparison between MC (solid) and data (points) for the timing information for CEM electrons from $W \to e\nu$ samples after cumulatively applying the various corrections (left side) and the corrections themselves (right side). Figure (a) shows the raw time as it comes out of the EMTD banks (MC only) [8]. Figure (b) shows the time after the slewing corrections, Figure (c) shows the vertex t_0 correction, Figure (d) the corrected time after the vertex t_0 correction, Figure (e) the TOF correction and Figure (f) the TOF corrected, i.e. the fully corrected time. In all cases the distributions are well centered around 0 and the resolutions of data and MC fit well with a fully corrected RMS of 0.64 ns. The mean of the fully corrected data of -0.11 ns is likely due to small correlations between the variables and can be corrected e.g. by improved run-by-run corrections. The two outliers with corrected times beyond 2 ns have been studied and are most probably cosmics.

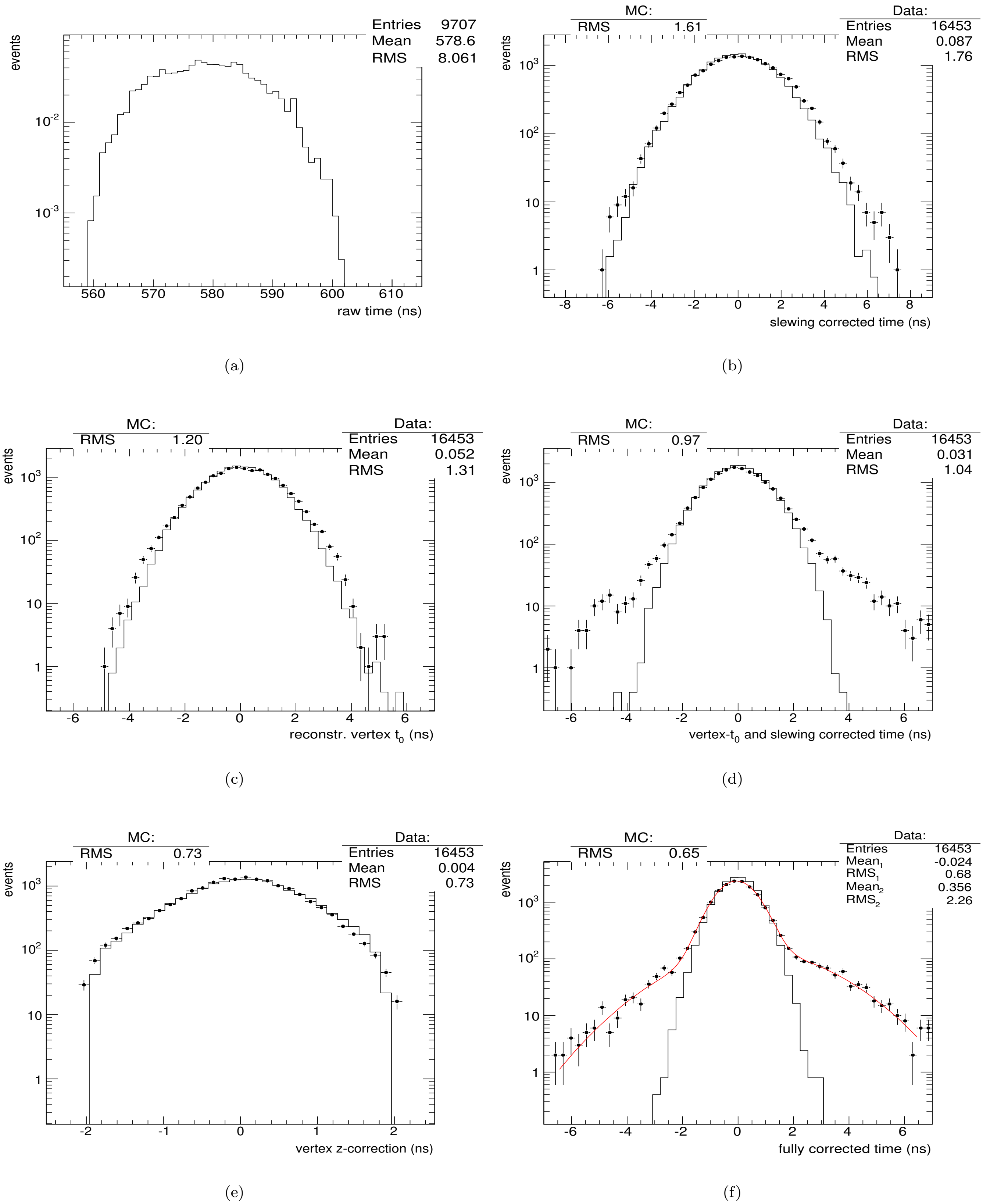


FIG. 3: The same as the previous figure but using PEM electrons. In all cases the distributions are well centered around 0 and well described by the sum of two Gaussians. The primary Gaussian of data and MC agree well, the secondary Gaussian in the data distribution of (d) and (f) comes from picking a vertex that is not associated with the electron (not simulated in MC). The shift between the primary and the secondary Gaussian tail in (f) is suspected to come from the difference in vertex z-distributions between the actual W events and the events that cause the second vertex, as discussed in [9].

Correction	CEM		PEM	
	MC (ns)	Data (ns)	MC (ns)	Data (ns)
Before corrections (raw)	8.10		8.06	
Slewing (E-dependent)	1.48	1.62	1.61	1.76
Vertex t_0	0.75	0.75	0.97	1.04
Vertex Position	0.63	0.64	0.65	0.68 (2.26)

TABLE I: The RMS of the corrected time distribution after cumulatively correcting for the effects shown in the first column for CEM and PEM electrons taken from data and MC using $W \to e\nu$ samples (see Figs. 2 and 3) [8]. The last row shows the fully corrected RMS. The number in parentheses is the RMS from the mismeasured vertex.

Effect	CEM		PEM	
	Contribution to Raw RMS (ns)	Uncertainty of Correction (ns)	Contribution to Raw RMS (ns)	Uncertainty of Correction (ns)
Slewing	7.9*	0.59*	7.9*	0.59*
Vertex t_0	1.20	0.25	1.20	0.24
Vertex Position	0.40	< 0.1	0.73	<0.1
TDC integer conv.	0.29	0.29	0.29	0.29
Total	8.10	0.63	8.06	0.65

TABLE II: The contributions of physics and hardware effects in the CEM and PEM, and the uncertainty of their corrections, to the RMS of the EMTiming system taken from $W \to e\nu$ MC samples (see Figs. 1 and the MC histograms in 2 and 3). The asterisk indicates that the slewing resolution is energy dependent and is estimated in Section III D.

II. DATASETS

We select central and plug electrons using standard ID cuts [10] from post-shutdown data with run numbers 191208-196902. This corresponds to 145.7 pb^{-1} of data. For the central we require one central EM cluster with $E_T > 20$ GeV from the W_NOTRACK trigger. For the plug we require one plug EM cluster with $E_T > 20$ GeV from the MET_PEM trigger. The final requirements for CEM and PEM electrons are listed in Table III.

Requirement
CEM Electrons
$E_T > 20$ GeV
$E_{\mathrm{Had}}/E_{\mathrm{Em}} < 0.055 + 0.00045 \cdot E$
$\chi^2_{\mathrm{Strip}} < 10$
Fiducial: $\lvert X_{\mathrm{CES}}\rvert < 21$ cm && 9 cm $< \lvert Z_{\mathrm{CES}}\rvert < 230$ cm
$Lshr < 0.2$
$p_T > 10$ GeV
$E^{\mathrm{Iso}} > 0.01 \cdot E_T$
$-3 < \Delta x \cdot q < 1.5$ && $\lvert\Delta z\rvert < 3$
$\lvert z_0\rvert < 60$
$p_T > 50$ GeV or $0.5 < E/p < 2.0$
Track quality: 3x3SLx5 hits
PEM Electrons
$E_T > 15$ GeV
$E_{\mathrm{Had}}/E_{\mathrm{Em}} < 0.05$
$\chi^2_{3\times3} < 10$
$E^{\mathrm{Iso}} < 0.1 \cdot E_{\mathrm{Leak}}$
$1.2 < \lvert\eta_{\mathrm{PES}}\rvert < 2.8$
$E^{\mathrm{u}}_{\mathrm{PES}}(5\times9) > 0.65$ && $E^{\mathrm{v}}_{\mathrm{PES}}(5\times9) > 0.65$
PEM 3x3 Fit Tower
0 cm$< \Delta R_{\mathrm{trk}}(\mathrm{PES}) < 3$ cm
Global Event Cuts
$\not{E}_T$ >25 GeV
good vertex: $\Sigma p_T^{\mathrm{trk}} > 2$ GeV, $N_{\mathrm{trks}} > 2$ and $\lvert z_{\mathrm{vtx}}\rvert < 60$ cm
Electron track-vertex matching:
$\lvert z_{trk} - z_{vtx}\rvert < 2$ cm and $\lvert t_{\mathrm{trk}} - t_{\mathrm{vtx}}\rvert < 1.3$ ns (CEM)
$\lvert z_{\mathrm{trk}} - z_{\mathrm{vtx}}\rvert < 5$ cm (PEM)

TABLE III: CEM and PEM electron ID requirements and global event cuts for the data and MC samples. "q" is the charge of the electron. The track requirements for the vertex reconstruction are listed in Table IV.

Requirement
$p_T > 0.3$ GeV
$p_T > 1.4$ GeV $\vert\vert$ $\frac{\mathrm{d}E}{\mathrm{d}x} < 20$
$\lvert z_0\rvert < 120$ cm
$\mathrm{Err}(z_0) < 1$ cm
$\lvert\eta\rvert < 1.6$
NCotStSeg(5)$\geq$2
NCotAxSeg(5)$\geq$2
$\lvert t_0\rvert < 40$ ns
$0.05 < \mathrm{Err}(t_0) < 0.8$ ns

TABLE IV: Track requirements for the vertex reconstruction.

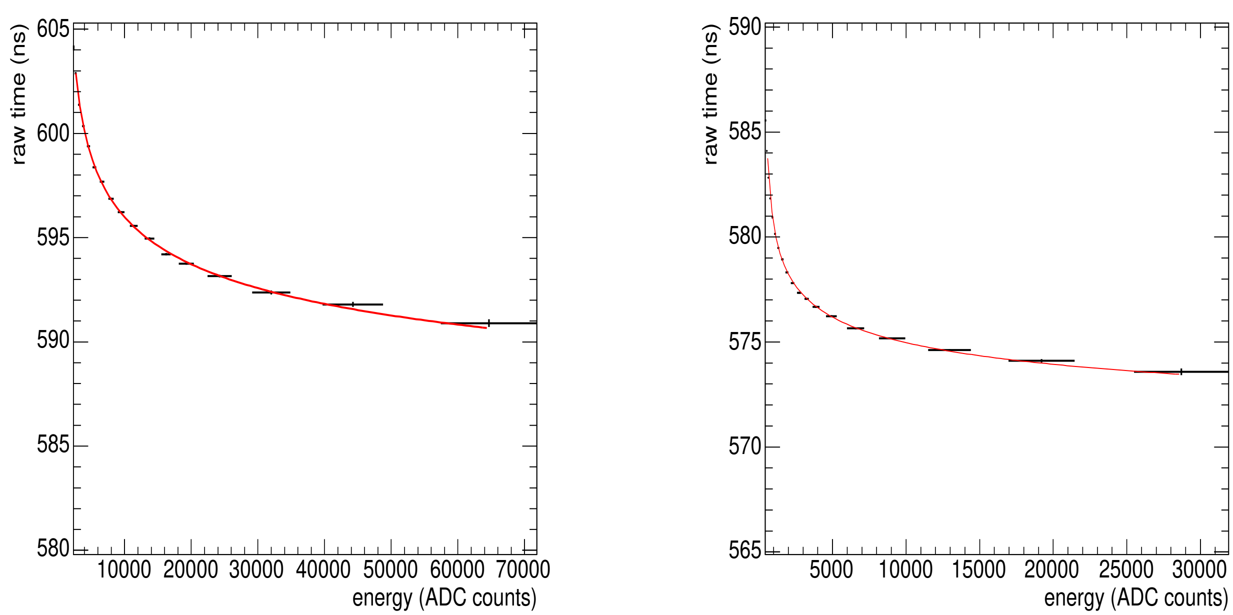
III. THE RESOLUTION OF THE EMTIMING SYSTEM AND THE CORRECTIONS

In this section we describe the corrections that lead to the fully corrected time distributions and their resolution. The raw time as it comes out of the TDC banks has to be corrected for the effects listed in Eq. 1; each is explained in one of the following subsections [11]. The resolutions of the corrections are estimated using the MC simulation as explained in Sections IV and V. Figures 2 and 3 summarize the timing distributions for central and plug electrons.

A. Slewing and Channel-by-Channel Corrections

The raw time has a channel-by-channel offset due to the cable length, and varies tower-by-tower due to TOF differences, PMT-energy asymmetry and an energy dependent slewing, all of which are corrected for using slewing calibration tables. The tables are produced from b-stream data runs taken after the full system installation [1]. Figure 4 shows the slewing effect for an example channel in the CEM and the PEM. The raw integer time values that are filled into the *StorableBank* are energy dependent and lie between 560 ns and 590 ns for a prompt particle.

As the actual mean vertex position per event has not been taken into account when producing the slewing calibration tables this has to be corrected separately offline. In practice this effect is only noticeable for the PEM ("ring-corrections"), as shown in Appendix A. This is also only true for data, because in MC the effect compensates as the slewing is simulated by applying the exact same table reversely. As shown in Table I and visualized in Figs. 2(b) and 3(b) after this set of corrections the RMS is reduced from ∼8 ns to 1.62 ns (1.76 ns) for CEM (PEM), centered around 0 for the MC [8].



(a) (b)

FIG. 4: The functional form of the slewing correction from calibration tables as a function of the energy sum of the PMTs (in ADC counts [12]) of an example tower in the CEM and in the PEM with the data for an example run. Note that the raw times are energy dependent due to the fixed height discrimination slewing of the ASD and lie between 560 ns and 590 ns for a prompt particle.

B. Correction for the Variation of the Vertex t_0

Figure 1(a) shows the typical distribution of the primary vertex t_0, i.e. t_{event}, for a MC $W \to e\nu$ sample with the default $\sigma = 1.3$ ns. The vertex reconstruction is done with a separate module using the t_0 of the tracks (from the *Track_CotT0* class) of the underlying event [9]. The vertex clustering requirements are listed in Tables III and IV. In particular, as tracks that are identified as slow protons are rejected, the simulation of the dE/dx of the tracks is critical for the MC quality. The dE/dx as it comes out of Stntuple (TStnTrack::DeDxCT()) has to be corrected for pedestals and the dip angle according to:

$$\frac{\mathrm{d}E}{\mathrm{d}x}_{\text{corr}} = (\text{TStnTrack :: DeDxCT()} - 12) \cdot (1 - 1.35 \log(\sin(\theta)))^{-1} \tag{2}$$

where θ is the dip angle of the track. Further, the track t_0 shows in both data and MC a charge-dependent correlation with the track p_T and η at low p_T, which have to be corrected. Appendix B gives details on the uncorrected and corrected distributions for MC as well as the correction function and the non-Gaussian tails in Figs. 1(c) and 1(e).

Figure 1(c) shows a comparison of the vertex reconstructed time vs. the true generated collision time. The RMS is a measure of the t_{event} resolution of the MC W sample. We measure it to 0.25 ns (CEM) and 0.24 ns (PEM). This makes up the largest non-EMTiming contribution to the final resolution. The reconstructed t_{event} distribution is shown in Figs. 2(c) and 3(c). The time distributions from the $W \to e\nu$ sample after this and the slewing correction are shown in Figs. 2(d) (3(d)) with a data RMS of 0.75 ns (1.04 ns) for the CEM (PEM) (see Table I). The tail in the data part in Figs. 3(d) and 3(f) is explained in the next section.

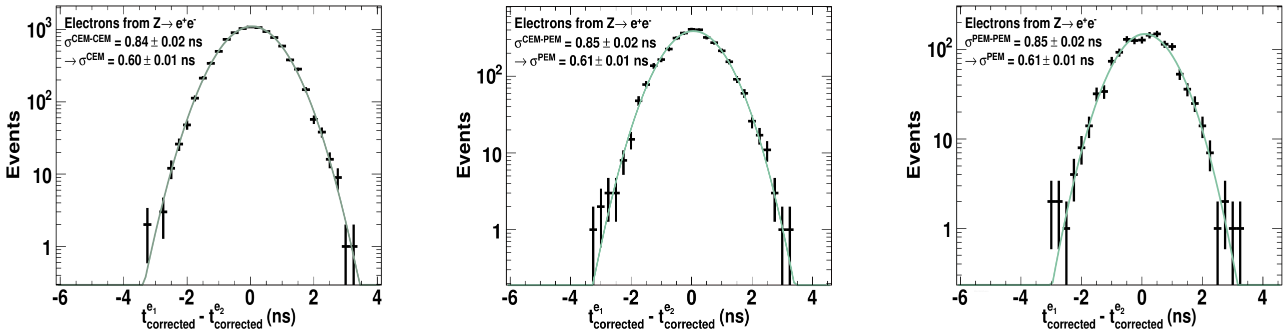
C. Correction for the Variation of the Time of Flight and the Fully Corrected Distribution

As not all collisions occur at $z = 0$, a correction factor for the time of flight (TOF) is introduced. Figure 1(b) shows the TOF correction needed to compensate for the collision position being different from $z = 0$ if the spread of the vertex position is $\sigma_z = 28$ cm, from a MC $W \to e\nu$ sample. The mean corrected TOF is centered at 0 with an RMS of 0.41 ns (0.74) for CEM (PEM) electrons. Figures 1(d) and 1(f) show the time difference of the TOF between the true and measured vertex z-position for a MC W sample. The small RMS shows that the resolution of the TOF correction is negligible at 0.003 ns (0.006 ns) for the CEM (PEM) assuming that the vertex is correctly picked. The time distribution from the reconstructed vertex has an RMS of 0.39 ns (0.73 ns) as shown in Figs. 2(e) and 3(e) for the CEM (PEM). After this correction the RMS of the fully corrected timing distribution is 0.64 ns (0.68 ns) for the CEM (PEM) as shown in Table I and visualized in Figs. 2(f) and 3(f). The mean of the fully corrected data distribution of -0.11 ns in Fig. 2(f) is most probably due to small corrections between the variables and can be corrected e.g. by improved run-by-run corrections. We also see two outliers with a corrected time beyond 3 ns. Event display plots show that they are most probably cosmics; note that we have no requirement on the effective transverse mass.

The data distributions in Figs. 3(d) and 3(f) show a fraction of events for which the highest Sum-p_T vertex is not associated with the electron. To understand this we note that tracks for PEM electrons are silicon only and thus cannot contribute to the vertex reconstruction as it requires both a z and t measurement. The RMS is, assuming no correlations between the vertex t_0 and z, the square-sum of the RMS's of the slewing corrected time and the reconstructed vertex t_0: $1.76 \oplus 1.31 \approx 2.19$ ns, which agrees well within the fit errors. We believe that the difference of $\sim$0.4 ns between the means of the two Gaussian fits comes from the different vertex topologies between the W vertices and the ones that cause the second vertex, which also become apparent in the ring-corrections (see Appendix A).

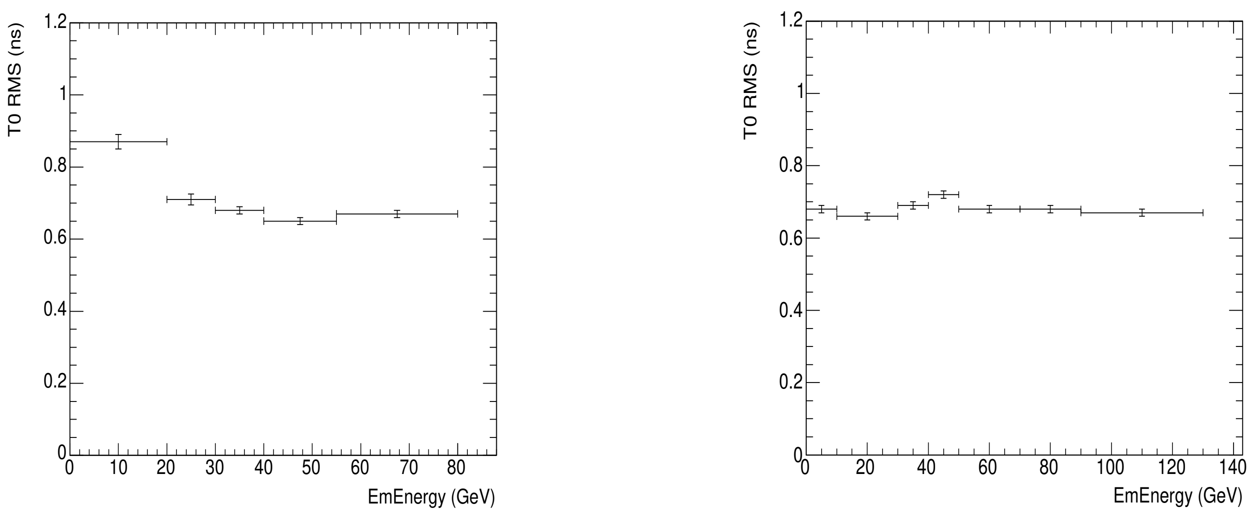
D. Intrinsic EMTiming System Resolution

The intrinsic EMTiming resolution can be estimated using a sample of $Z \to ee$ events. Figure 5 shows the arrival time difference of the two decay electrons in a data $Z \to ee$ sample at high energy. The RMS is $\sim$0.84 ns after correcting for the TOF (see Section III C), independent of whether the electrons hit the CEM or PEM. From this one gets a resolution of $\sigma_{\text{EMTiming}} = 0.84 \text{ ns}/\sqrt{2} = 0.59$ ns. After subtracting the RMS in quadrature that comes from rounding the time to integers in the *StorableBank* ($\frac{1}{\sqrt{12}} \approx 0.29$ ns) and subtracting the resolution for the TOF correction (see Fig. 1(d), negligible) one gets an "intrinsic" EMTiming system resolution of 0.5 ns. If one puts this value into the MC simulation and varies it according to its energy-dependence derived from data (see Fig. 6, where we look at EM clusters with low energy) one can reproduce the data plots after each correction as shown in Figs. 2 and 3 and as explained in Section V which is a confirmation of its correctness.



(a) (b) (c)

FIG. 5: The difference in the arrival times of the two decay electrons from a Z using data, corrected for the TOF. The figures show the case when the electrons are CEM-CEM (a), CEM-PEM (b) and PEM-PEM (c). The distributions are centered at 0 ns with an RMS of $\sim$0.84 ns which indicates an intrinsic EMTiming resolution of 0.84 ns/$\sqrt{2} \approx 0.59$ ns.



(a) (b)

FIG. 6: The RMS of the fully corrected arrival time (see Section III B) versus the electron EM energy from (a) a CEM and (b) a PEM electron data sample produced with the requirements in Table III without the E_T cut. This effect is simulated by varying the intrinsic EMTiming resolution as a function of the EM energy of the incident particle with an RMS of 0.5 ns at an EM energy of 25 GeV. We note that the intrinsic resolution as a function of energy is found by folding the value in the figure with the TDC resolution and the tracking resolution (negligible).

IV. MONTE CARLO FRAMEWORK AND MODULES

The EMTiming MC modules are located in the *CalorTimeMods* package. The executable part is *test/McTiming.cc* and the library part is *src/TDCBankSimModule.cc* and *CalorTimeMods/TDCBankSimModule.hh*. The binary is run after MC files are generated independently with *cdfSim* [6]. As with real data the EMTiming information is filled in the *EMTD_StorableBank*. Both reconstructed particle information from *CalData* and true information from *OBSV_StorableBank* and *OBSP_StorableBank* are used as input. We quote results using PYTHIA with the default settings in the *cdfSim* framework and the default MC production settings of cdfsoft release 6.1.2. In the next section we describe the generation of the arrival time in the Monte Carlo simulation using the intrinsic resolution and the corrections that have been explained in Section III.

V. MONTE CARLO GENERATION AND RECONSTRUCTION

In this section we describe the process of generating the true time of arrival, simulating the effects at the calorimeter, correcting for the TOF and vertex time and storing the time value in the *EMTD_StorableBank*. We summarize the steps and then describe each in detail:

1. Calculate the true arrival time for a particle
2. Check to see if it hits an EMTiming-instrumented part of the detector
3. Smear the corrected arrival time for the intrinsic resolution (see Section III D)
4. Check to see if it has enough energy to create a hit (see Section III A)
5. Convert the arrival to a raw time using the slewing curve (see Fig. 4)
6. Truncate the raw time to an integer to simulate the TDC
7. Put the raw time into the *EMTD_StorableBank*

Steps (1)-(3) yield the "generated arrival time" which is explained in Section V A, steps (4)-(7) are explained in Section V B.

A. Generated Arrival Time

As the EMTiming module is not part of the calorimeter simulation, the final simulated arrival time of a particle at the calorimeter, taking into account the vertex time and TOF, t_{arrival}, is obtained using information generally available in the production output, i.e. true vertex information from the *OBSV_StorableBank* and true particle information from the *OBSP_StorableBank*. It can be expressed as follows:

$$t_{\text{arrival}} = t_{\text{vert}} + \frac{|\vec{x_f} - \vec{x}_{\text{vert}}|}{|\vec{v}_{\text{part}}|} \tag{3}$$

where $\vec{v}_{\text{part}}$ is the velocity of the particle, $\vec{x}_{\text{vert}}$ and t_{vert} the position and time of its production vertex and $\vec{x}_f$ its final position at the central (CES) and plug (PES) shower maximum subdetectors (r = 184.15 cm and |z| = 185.4 cm respectively). As the energy threshold of the timing system is on the order of 2-3 GeV [2] (see Section III A) one can calculate t_{arrival} by extrapolating the particle trajectory as a straight line from $\vec{x}_{\text{vert}}$ (either as part of a decay or directly), to the position $\vec{x}_f$. The value of t_{arrival} is then given by the vertex time plus the TOF of the particle. The vertex time information in the *OBSV_StorableBank*, t_{vert}, takes into account the simulation of the primary vertex position and time, t_{event} (see Fig. 1(a)), and, if it is a decay vertex, the actual time the decay parent needed to propagate through the detector volume. As the slewing calibration corrections assume that particles come from the center of the detector, $\vec{x}_{event} = \vec{0}$, at $t_{\text{event}} = 0$ ns, t_{arrival} is corrected for the TOF of the particle that it would have if it came from $\vec{0}$. The corrected arrival time, $t^{\text{corr}}_{\text{arrival}}$, is then:

$$t^{\text{corr}}_{\text{arrival}} = t_{\text{arrival}} - \frac{|\vec{x}_f|}{c} \tag{4}$$

For a prompt particle this is equivalent of $t_{\text{event}} + t_{\text{zcorr}}$, however for decay particles it correctly takes into account the travel time of the decay parent.

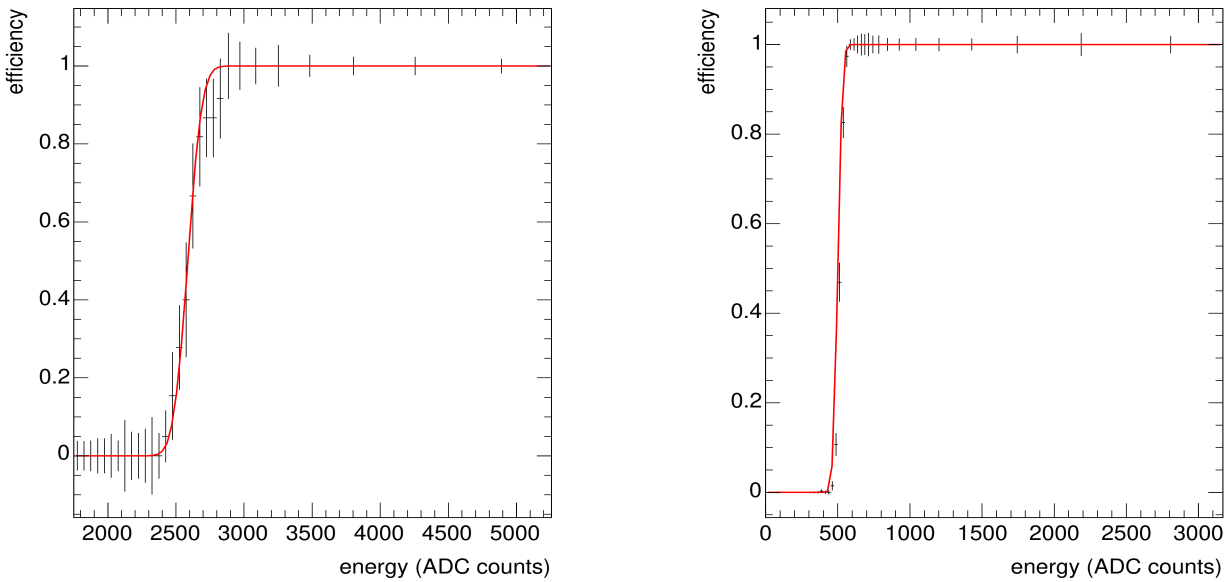
We next check to see if the particle hits the detector. While this is typically straightforward there are few pathological cases dominated by when the energy deposit happened in the neighboring tower rather than the tower calculated in $\vec{x}_f$. In this case we assign the same timing information to the neighboring tower. Particles that do not deposit energy in the calorimeter or that have a decay vertex outside the detector volume are filtered out. We note that in the case of multiple particles arriving at the same tower, only the first hit in time is used. Considering the resolution of the system this is a good approximation for jets. In this case there is no handle available to decide which particle actually triggered the timing system as the fraction of energy left in the EM calorimeter is not known. However this is only critical if the energies of the particles that hit the same tower are around threshold (see Sec. V B).

If the particle passes the above requirements the intrinsic EMTiming resolution (see Section III D) is implemented as a Gaussian smearing of $t^{\rm corr}_{\rm arrival}$ and is varied according to its energy-dependence derived from jet data (see Fig. 6).

B. Calibration Tables Implementation and Raw Time

Once we have a set of towers with potential timing hits, we use the calibration tables to test in the simulation as to whether the deposited energy, taken from *CalData*, is sufficient for triggering a timing hit. The hit probability is determined using the energy turn-on curves determined from jet data. Figure 7 shows the hit efficiency dependence on the energy sum of the PMTs of an example tower for both the CEM and the PEM [12]. The threshold for all towers lies consistently at $\sim$3.5 GeV and $\sim$2 GeV deposited energy for the CEM and PEM respectively with widths of $\sim$1 GeV and $\sim$0.5 GeV [4]. The run dependence of the threshold curves is negligible unless a channel is broken (broken energy measurement or broken time measurement). In this case the channel is flagged bad in the calibration tables and no timing hits are registered.

If a tower is determined to have a hit, then the generated time is converted to a "raw time" using the inverted slewing calibration tables (see section III A) and is truncated to integer to simulate the TDC. This is put into an *EMTD_StorableBank* so that the time format resembles the output of the TDCs for real data, and can be easily processed by the standard data correction tools. Note that the same conversion table is used later when applying the slewing corrections to reconstruct $t^{\rm corr}_{\rm arrival}$. Figures 2(a) and 3(a) show, using MC $W \rightarrow e\nu$ samples in CEM and PEM respectively that the values range from 560 to 590 ns for a prompt particles with an RMS of $\sim$8 ns. The raw time mimics the data such that it can be processed with the standard data correction tools.



(a)

(b)

FIG. 7: The functional form of the TDC efficiency as a function of energy sum of the PMTs of an example tower in the CEM (left) and the PEM (right) in ADC counts [12] from data calibration tables with the data for an example run. In this case the threshold is around 7.0 GeV (1.6 GeV) for the energy sum and the width around 1.2 GeV (0.5 GeV) for the CEM (PEM). Note that in the CEM the energy is here given by the energy sum of the PMTs instead of the usual geometric average as this is the quantity that the fixed height discriminator triggers on.

Timing requirement	$t < -t_0$	$-t_0 < t < +t_0$	$t > +t_0$
$t_0 = 2$ ns	60 (54)	42129 (42136)	43 (42)
$t_0 = 3$ ns	0 (1)	42232 (42230)	0 (1)
$t_0 = 5$ ns	0 (0)	42232 (42232)	0 (0)

TABLE V: The number of CEM electrons from $W \rightarrow e\nu$ events taken from Fig. 2(f) that pass the various cuts on the final corrected time at 2, 3 and 5 ns, as indicated in the first row. The data, to which the MC is normalized, is shown in parentheses, with a total of 42232 events without a timing cut. For data, t_0 is shifted by the mean offset of -0.11 ns to correctly compare it to the MC.

VI. RESULTS AND DISCUSSION

With the full simulation in hand we now study the efficiencies for high energy photons or electrons that pass various cuts on the final corrected time. We consider two different analysis: (1) a "prompt analysis" where the photon arrives promptly in time [13] and (2) a "delayed analysis" e.g. with photons from long-lived neutralinos where the photon arrives with a time-delay with respect to the collision time [3]. Table V shows the number of events in the MC sample of prompt, central electrons (normalized to data) that pass timing cuts of 2, 3 and 5 ns for a correctly chosen vertex. Note that for a prompt analysis the interesting quantity is the efficiency of events within the timing window which is 99.76% (99.77%), 100% (99.99%) and 100% (100%) for cuts at $t_0 = 2$, 3 and 5 ns respectively for MC (data).

We estimate the efficiency and the systematic uncertainty of the signal in a delayed photon analysis using photons from long-lived neutralinos with a lifetime of 10 ns and a mass of 110 GeV from a PYTHIA GMSB MC sample. Figure 8 shows the fully corrected time of arrival of photons decayed from long-lived neutralinos from MC, compared with the prompt distribution. Clearly visible is the exponential tail. The fraction of events with $t > t_0 = 2$ ns is 53.3%. We get 58.07% (49.09%) of the events if the mean shifts by $+(-)0.2$ ns ($\sim$ 8% systematic uncertainty) and 53.46% (53.68%) of the events if the intrinsic resolution is shifted from 0.5 ns to 0.48 (0.52) ns ($\sim$1% systematic uncertainty).

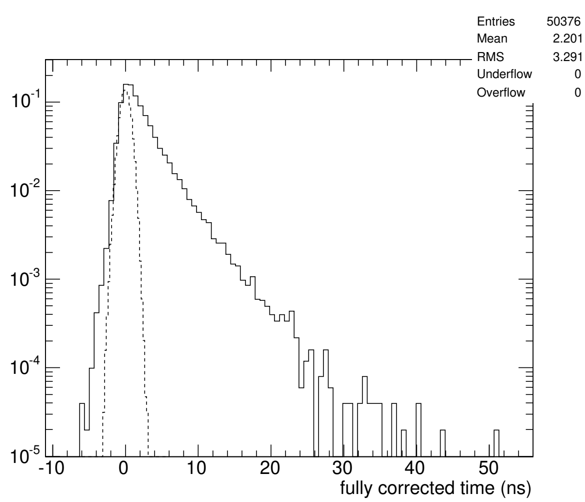


FIG. 8: The time of arrival of photons from neutralinos in a GMSB model with a neutralino lifetime of 10 ns and a mass of 110 GeV after all corrections applied to the MC. The dashed histogram shows the prompt distribution of CEM electrons from Fig. 2(f).

VII. CONCLUSION

We have measured the resolution of the EMTiming system and used a MC simulation to estimate and crosscheck the various contributions that lead to the fully corrected time of arrival of isolated particles such as electrons in a $W \rightarrow e\nu$ sample. The corrections to the raw time include slewing effects, channel-by-channel offsets and corrections for vertex position and time. The raw time distribution from the MC, as well as the time distributions after each of the subsequent offline corrections, match the corresponding data distributions well. To summarize, the resolution at high energy is 0.64 ns (0.68 ns) for CEM (PEM) where the main contributions to the uncertainty in the raw time distribution come from

- the intrinsic resolution of the system (0.5 ns) at high energy
- the precision of the TDC output (0.29 ns)
- the vertex t_0 resolution (0.25 ns)

The dominant effect however comes from a sample-dependent fraction of events for which the wrong vertex is picked, producing a corrected time distribution with an RMS of 2.3 ns. The efficiency for an in-time cut at $t_0 = 2$ ns (3σ) is 99.8%, assuming the correct vertex.

Acknowledgments

The authors would like to thank S. Yun and the Simulation group for helpful suggestions on *cdfSim*. We would also like to thank Fermilab and the FNAL CAF system administrators for providing computing power.

---

[1] http://hepr8.physics.tamu.edu/hep/emtiming/

[2] H Frisch, H. Sanders, M. Cordelli, F. Happacher, S. Miscetti, R. Wagner, M. Goncharov, T. Kamon, V. Khotilovich, V. Krutelyov, S.W. Lee, D. Toback and P. Wagner, "The Timing System for the CDF Electromagnetic Calorimeters", CDF Note 7918. Submitted to NIM, FERMILAB-PUB-05-543-3, physics/0502171.

[3] P. Wagner and D. A. Toback, Phys. Rev. D **70**, 114032.

[4] M. Goncharov, V. Khotilovich, V. Krutelyov, S.W. Lee, D. Toback and P. Wagner, "Performance of the EMTiming System as Seen by the ObjectMon Online Monitoring System," CDF Note 7515.

[5] This release has been chosen mainly for the good tracking simulation. The time out of the timing system has to be corrected for the vertex time which critically depends on the dE/dx of the tracks (see the Section about vertex correction).

[6] PYTHIA: T. Sjöstrand, L. Lönnblad and S. Mrenna, arXiv:hep-ph/0108264; we generate 140000 $W \to e\nu$+jets events using process number MSEL=14 and decay channel MDME(206)=1
cdfSim: G.S. Sganos, Tutorial on cdfSim.

[7] M. Goncharov, V. Krutelyov, D. Toback and P. Wagner, "Search for Neutral, Long-Lived Particles that Decay to Photons," CDF Note 8016.

[8] For technical reasons we have decided not to show distributions of the raw time from the TDC banks.

[9] M. Goncharov, V. Krutelyov, D. Toback and P. Wagner, "Space-Time Vertex Reconstruction Using COT Tracks," CDF Note 8015.

[10] C. Issever, B. Wagner, A. Robson and C. Mills, "Plug Electron Baseline Cuts as defined in Summer 2003 and their Efficiencies," CDF Note 6789.

[11] On Stntuple level run-by-run corrections are applied which take into account the relative clock shifts between the calorimeter detectors. To make the data handling consistent, these corrections are applied reversely during the MC simulation.

[12] The conversion from ADC counts to energy is roughly: 1000 ADC counts $\simeq$ 3 GeV.

[13] P. Onyisi, "Run 2 Photon+$\not\!\!E_T$ Search," CDF Note 6381

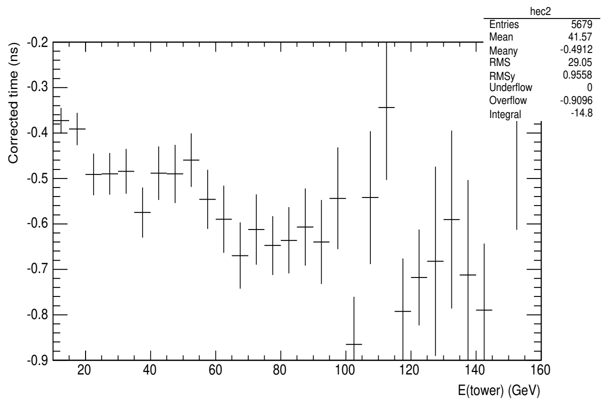
[14] Note that the aspect angle correction and the wire correction have to be disabled when generating Stntuples from MC production files. (Not committed as of now). Figure 11 has been produced without these data-only corrections.

[15] T. Philipps, "Event T0 from the COT and the CLC," CDF Note 6277.

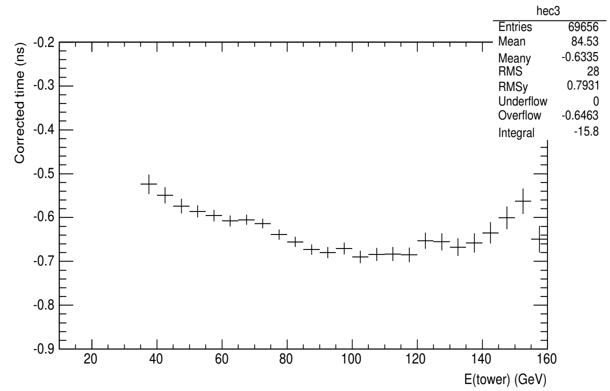
APPENDIX A: CORRECTIONS TO THE SLEWING CALIBRATION

As discussed in Section III A, the tower-by-tower slewing calibrations were made without taking into account the vertex position event-by-event in the sample from which it was made. For PEM electrons the correlation of the vertex position from the EM tower position and the EM energy turns out to be non-negligible as there are variations in the mean of the fully corrected time as a function of vertex position. This data-only effect is compensated offline with ring- and energy-dependent corrections ("ring-corrections"). Figures 9 and 10 show the fully corrected time as a function of energy, tower and vertex position before and after applying the ring-corrections for the MET_PEM data sample we use and for a subset (run 195571) of the b-stream data sample from which the slewing corrections were originally made. The energy and ring-dependent offset f is calculated from MET_PEM and for the energy dependency (E) a fit to b-stream data is used with a ring-dependent slope parameter g: $f + g/E$. This is the correction as it is done in the code:

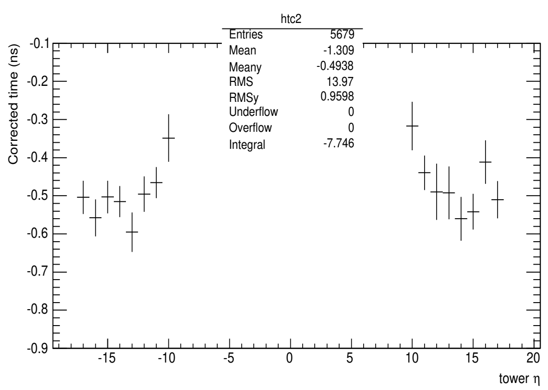
```
float _towerPemTave[16] = {-0.562098,
  -0.6631,
  -0.713025,
  -0.688011,
  -0.692245,
  -0.621683,
  -0.54,
  -0.53,
  -0.47,
  -0.65,
  -0.661043,
  -0.7726,
  -0.779405,
  -0.819555,
  -0.734077,
  -0.653045};

  float ringcorrt0=0;
  float energy = TStnElectron::E();

  if(tower>= 10 && tower< 18) ringcorrt0 = towerPemTave[index];
  if(tower>= 10 && tower< 18) ringcorrt0 += 4.13895/energy;

  if(( tower== 15 || tower==16 || tower==13) && side==1) ringcorrt0+=1.8/energy;
  if(( tower== 11 || tower==17) && side==-1) ringcorrt0-=1.0/energy;
```



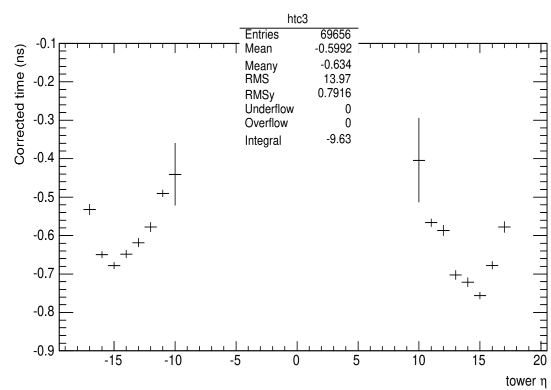
(a)



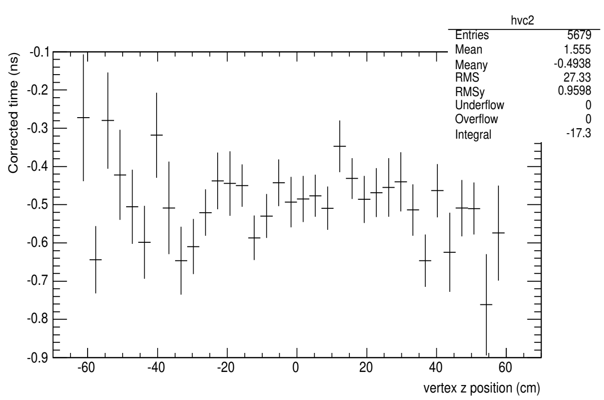
(b)



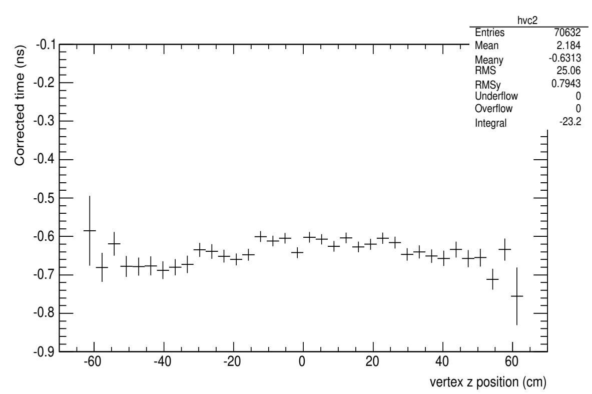
(c)



(d)

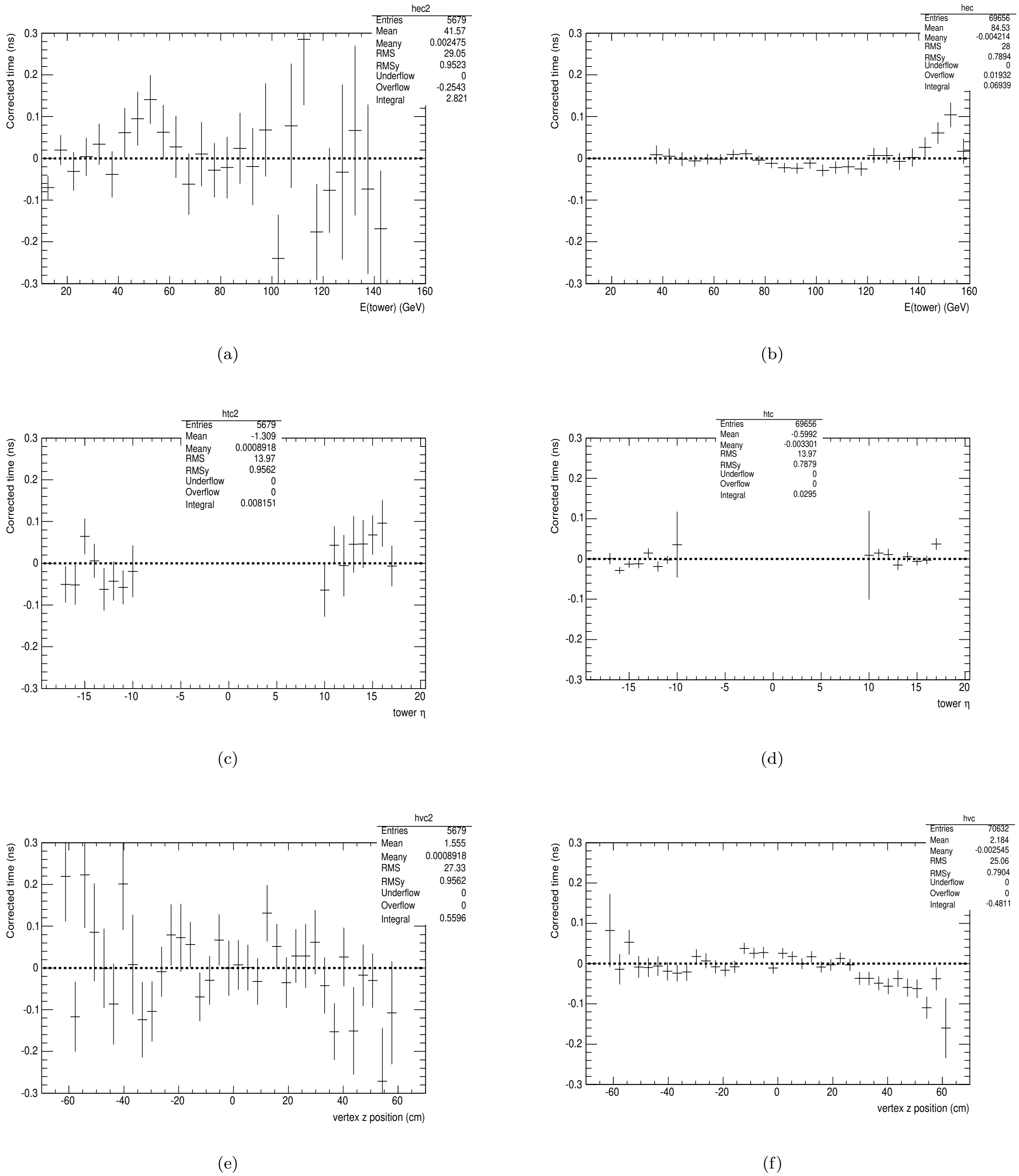


(e)



(f)

FIG. 9: The fully corrected time before applying the ring-corrections as a function of tower energy, tower number (integrated over ϕ) and vertex position from b-stream (only run 195571) on the left and the MET_PEM trigger (runs 191208-196902) on the right side. Both samples show the same tendency as a function of the above variables: the corrected time falls with energy and shows a u-shape as a function of tower number. It is flat as a function of vertex position.



(a) (b)

(c) (d)

(e) (f)

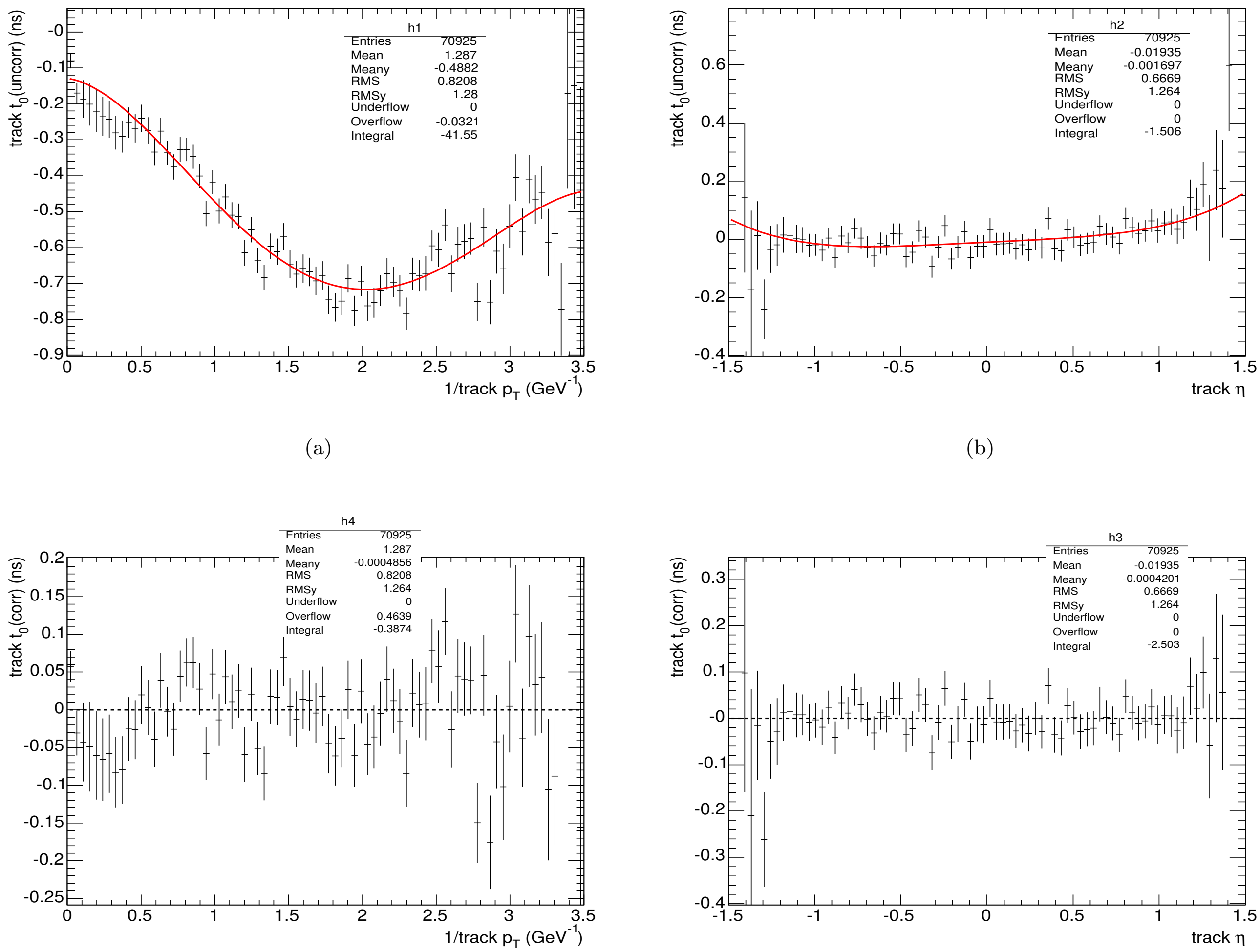
FIG. 10: The fully corrected time after applying the ring-corrections as a function of tower energy, tower number (integrated over Φ) and vertex position from b-stream (only run 195571) on the left and the MET_PEM trigger (runs 191208-196902) on the right side. All dependencies are resolved as the mean corrected time is well centered and all variations are within 0.2 ns.

APPENDIX B: CORRECTIONS TO THE VERTEX t_0

The track t_0 in Stntuple is produced with the *Track_CotT0.hh* module in the *CalorTime* package and is accessible as *TStnTrack::T0Cot()*. Figures 11 and 12 show its distribution as a function of p_T and η for positively and negatively charged tracks separately [9, 14]. Since it is not flat this effect is corrected in MC using the following formulae:

```
float xPt = TStnTrack::Pt();
float xEt = TStnTrack::Eta();
if(TStnTrack::Charge()==-1) {
  TStnTrack::T0Cot() = TStnTrack::T0Cot() - 0.55*sqrt(1.5/xPt)+0.1;
  TStnTrack::T0Cot() = TStnTrack::T0Cot() - 0.01*(9-10*pow(xEt,4));
}
if(TStnTrack::Charge()==1 && xPt<10) {
  TStnTrack::T0Cot() = TStnTrack::T0Cot() - ((2.56521-1.63752e1*xPt)/(1+exp(-(1/xPt-2)))
                                          + (1.92549+2.21641*xPt)/(1+exp((1/xPt-2))));
  TStnTrack::T0Cot() = TStnTrack::T0Cot() - 0.01*(-2.64164+3.32608e-01*xEt
                                          + 4.15586*pow(xEt,2)+3.64391*pow(xEt,3)
  + 2.28618*pow(xEt,4));
}
```

Figure 13 shows the difference between the reconstructed and the true vertex t_0. The non-Gaussian part of Figs. 1(c) and 1(e) is due to the track t_0 being biased towards a vertex t_0 of 0 ns (see Fig. 13). This effect is present in MC and data due to tracking hits in the COT being associated to a track with the assumption that the vertex t_0 is 0 ns. With these biased hits the track t_0 is calculated [15]. This effect is not corrected for as the fraction of the events on the non-Gaussian tail is low ($\sim 1\%$) with only a minor systematic error on the order of 0.3 ns at a vertex t_0 of 3 ns.



(a) (b)

(c) (d)

FIG. 11: The mean t_0 of the negatively charged tracks as it comes out of *TStnTrack::T0Cot()* as a function of the $1/p_T$ and η of the tracks. Figure (a) shows the p_T before any corrections, (b) the η after the corrections for p_T, and (c) and (d) both distributions after both corrections are applied.

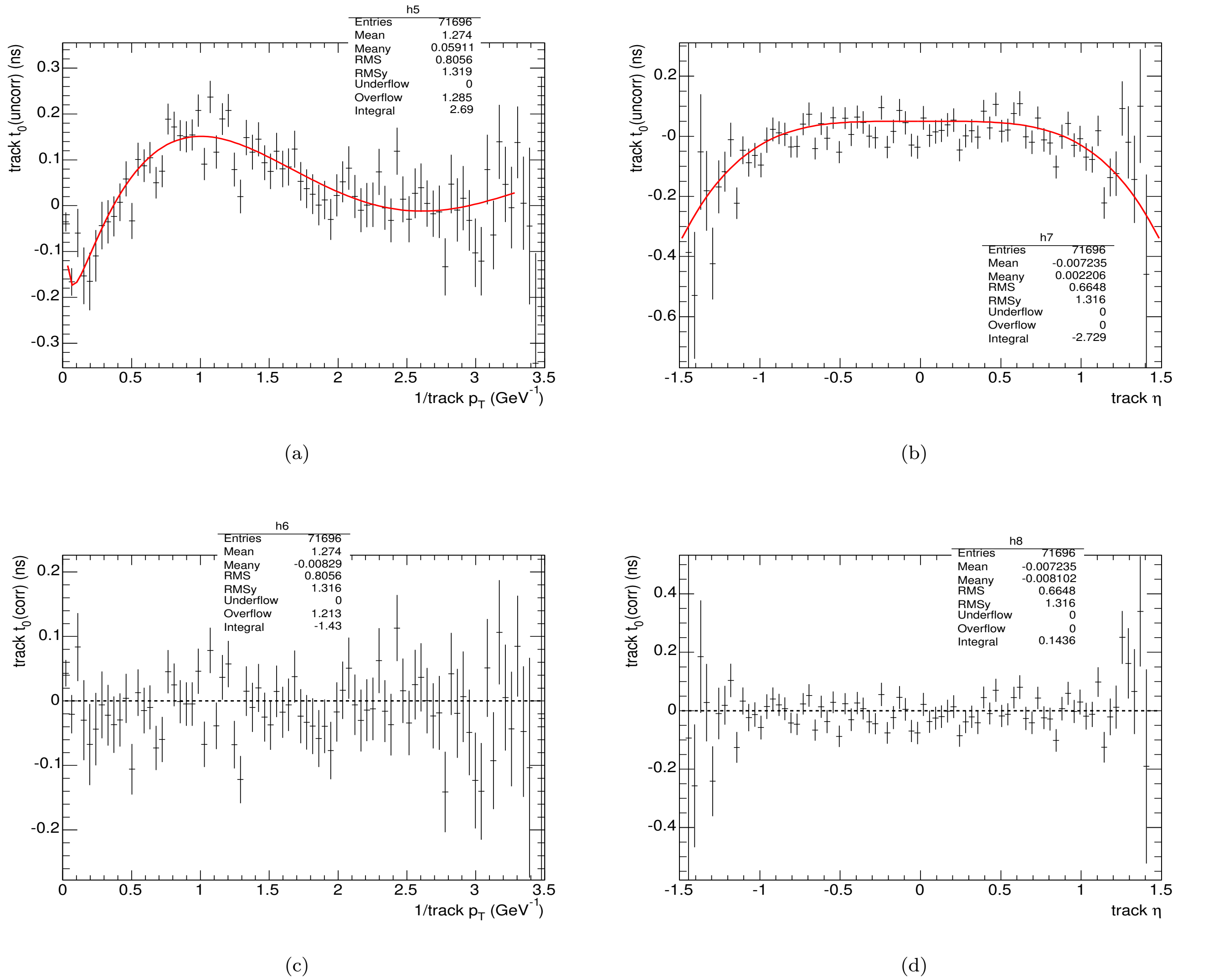


FIG. 12: The t_0 of the positively charged tracks as it comes out of *TStnTrack::T0Cot()* as a function of the p_T and η of the tracks. Figure (a) shows the p_T before any corrections, (b) the η after the corrections for p_T, and (c) and (d) both distributions after both corrections are applied.

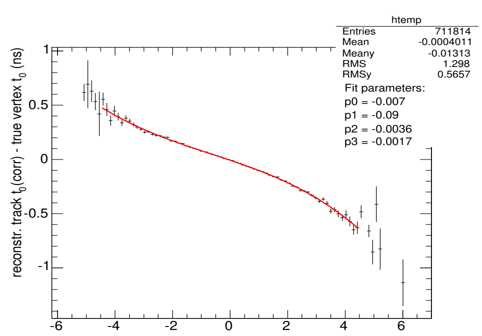


FIG. 13: The difference between the reconstructed track t_0 and the true vertex t_0 as a function of the true vertex t_0 after the corrections derived from Fig. 11 for all tracks that pass the basic requirements in Table IV. The fit is made with a 3rd order polynomial. This effect is due to the biased hit selection towards 0 of the track reconstruction algorithm.