ACD Recon Upgrade -- Pass 8 Optimization

Introduction

After a first look at the AcdV2 cuts applied to Pass8 data, it seems like some cut optimization is in order. Additionally, from discussions with Bill it became obvious that the data sets and prefilters used in that analysis were not optimal. The goal here is to create a framework for designing the AcdV2 selection. Since it is likely that some of the Pass8 foundation will change along the way, the emphasis here is to develop tools and a methodology for creating this type of analysis. Additionally, we would like to show that we can reproduce the ACD cut efficiency found in Pass7

Background

- At Eric's suggestion, I looked into some of the work Markus did to develop the Pass7 event classes. Specifically, the plot on slide 3 seems like a very useful way to approach the ACD selection at hand.
- To examine the selection efficiency as a function of energy, we must have an event energy estimate. Carmelo has done some work to create simple energy selection for Pass8. Additionally, Carmelo has shown that for right now, CTBBestEnergy should serve adequately for this purpose.

Code Base

I've started to collect some of the code I've been developing in CVS:

```
users/kadrlica/eventSelect/python/
```

However, the code is rough and should be used at your own risk!

Prefilter Definition

Prefilters are defined in more detail here.

Pass 7 Events

To compare with Bill's Pass7 results, I collected the root files that he was using. I believe that they correspond to the following data sets in the data catalog:

```

#All Gamma
root://glast-rdr//glast/mc/ServiceChallenge/allGamma-GR-v17r31p14-OVL/merit/allGamma-*.root
#Background
root://glast-rdr//glast/mc/ServiceChallenge/background-GR-v17r31p14-OVL/merit/background-*.root
```

```
Pass7_AcdAliasDict = {
    # Pass7 Acd Variables
    "AcdCornerDocaENorm" : "AcdCornerDoca*(min(1000, max(30, CTBBestEnergy)))^.5/10." ,
    "AcdTkr1RibbonActDistENorm":"AcdTkr1RibbonActDist * sqrt(min(3000., max(10, CTBBestEnergy)))/10.",
    "AcdTkr1ActiveDistENorm":"AcdTkr1ActiveDist * sqrt(min(3000., max(10, CTBBestEnergy)))/10." ,
    "AcdTileEventEnergyRatio":"100*AcdTkr1ActDistTileEnergy/max(10., CTBBestEnergy)" ,
    "AcdTotalTileEventEnergyRatio":"100.*AcdTotalEnergy/max(10., CTBBestEnergy)" ,
    "AcdTkrVActiveDistENorm":"AcdActiveDist3D  * sqrt(min(3000., max(10, CTBBestEnergy)))/10." ,
    # Why this 1* is necessary, we may never know...
    "AcdTkr1RibbonActDistMaxTileEnergy":"1*max(AcdTkr1RibbonActEnergyPmtA, AcdTkr1RibbonActEnergyPmtB)",

    # Pass7 Acd Cuts
    "RibbonCut_p7":"(AcdTkr1RibbonActDistENorm > -40  && Tkr1SSDVeto < 3 &&  AcdTkr1RibbonActDistMaxTileEnergy
> .04)",
    "CornerCut_p7":"((Tkr1LATEdge/1.5)^2 + (AcdCornerDocaENorm - 10)^2 < 6400  && Tkr1SSDVeto < 3) ||
(Tkr1LATEdge < 300 && abs(AcdCornerDocaENorm-2) < 4)",
    "BasicTileCut_p7":"Tkr1SSDVeto< 5 && AcdTkr1ActDistTileEnergy > .7 && AcdTkr1ActiveDistENorm> -350",
    "TotalTileEnergyCut_p7":"AcdTotalTileEventEnergyRatio > .8 || AcdTkr1ActiveDistENorm > -200 &&
AcdTotalTileEventEnergyRatio > max(.005, .1 -  .0001*AcdTkr1ActiveDistENorm) * max(1., CTBBestLogEnergy/2.5)",
    "VetoTileCut_p7":"(AcdTkrVActiveDistENorm > -60 && AcdActDistTileEnergy /sqrt(max(1., CTBBestLogEnergy-
3.5)) > .9 +.15* TkrVSSDVeto)",
    "TileEdgeCut_p7":"Tkr1SSDVeto == 0 & abs(AcdTkr1ActiveDistENorm) < 10 & AcdTkr1ActDistTileEnergy > .025",
    "BasicAcdFilter_p7":"!(RibbonCut) && !(CornerCut) && !(BasicTileCut) && !(TotalTileEnergyCut) && !
(VetoTileCut) && !(TileEdgeCut)",
```

The first step is to compare against the event selection that Bill does. The results seem very comparable. One possible difference is that IM truncates it's floating point precision.

BILL PASS7	SIG	BKG
RibbonCut_p7	99.66% (99.66%)	88.38% (88.38%)
CornerCut_p7	98.25% (98.59%)	83.70% (94.70%)
BasicTileCut_p7	90.19% (91.79%)	16.04% (19.16%)
TotalTileEnergyCut_p7	85.42% (94.71%)	6.269% (39.09%)
VetoTileCut_p7	84.37% (98.77%)	4.384% (69.94%)
TileEdgeCut_p7	84.29% (99.91%)	4.312% (98.35%)

PASS7	SIG	BKG
RibbonCut_p7	99.6% (99.6%) – 121159	89.25% (89.25%) – 159950
CornerCut_p7	98.29% (98.68%) – 119564	85.81% (96.15%) – 153790
BasicTileCut_p7	92.64% (94.25%) – 112691	13.95% (16.26%) – 25009
TotalTileEnergyCut_p7	86.39% (93.25%) – 105089	7.007% (50.21%) – 12558
VetoTileCut_p7	84.8% (98.16%) – 103159	4.951% (70.66%) – 8873
TileEdgeCut_p7	84.74% (99.93%) – 103084	4.884% (98.65%) – 8753

These are actually not the final version of the Pass7 ACD cuts. The RibbonCut_p7 and the VetoTileCut_p7 were both expanded slightly...

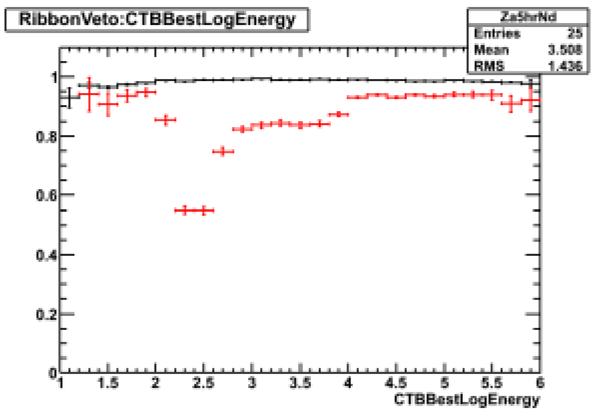
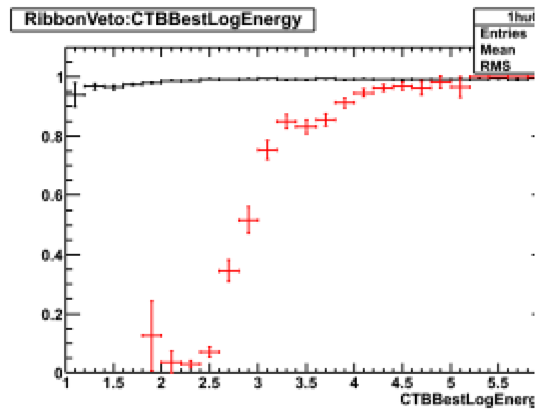
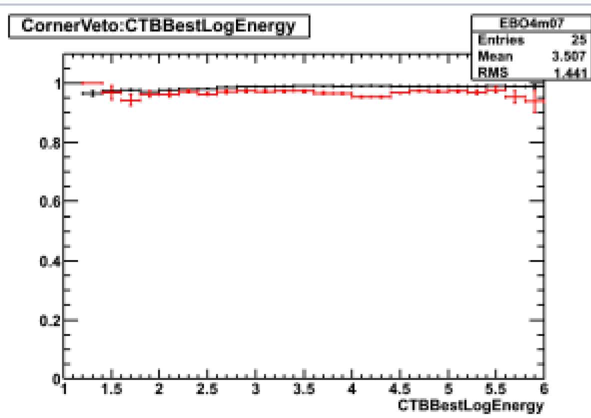
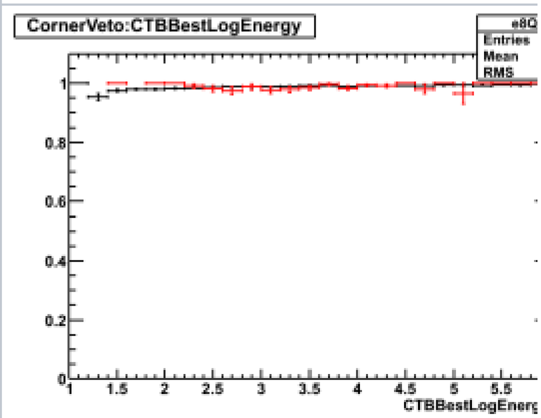
```
    "RibbonCut_p7":"(AcdTkr1RibbonActDistENorm > -40  && Tkr1SSDVeto < 3 &&  AcdTkr1RibbonActDistMaxTileEnergy
> .04) || (AcdTkr1RibbonDist > -1/(CTBBestEnergy/100)  && Tkr1SSDVeto < 2)",
    "VetoTileCut_p7":"(AcdTkrVActiveDistENorm > -100 && AcdActDistTileEnergy /sqrt(max(1., CTBBestLogEnergy-
3.5)) > .9 +.15* TkrVSSDVeto) || (abs(AcdTkrVActiveDistENorm) < 15 && AcdActDistTileEnergy > .25 && TkrVSSDVeto
< 2)",
```

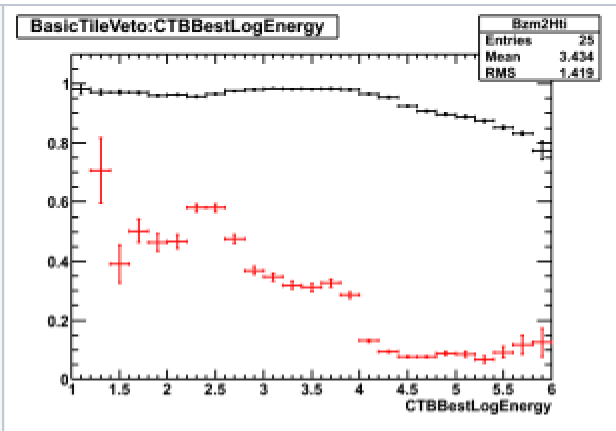
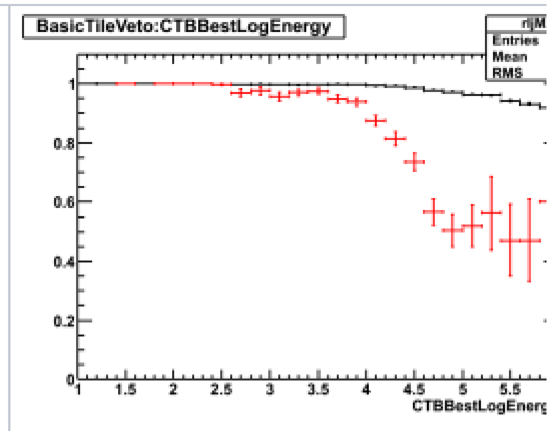
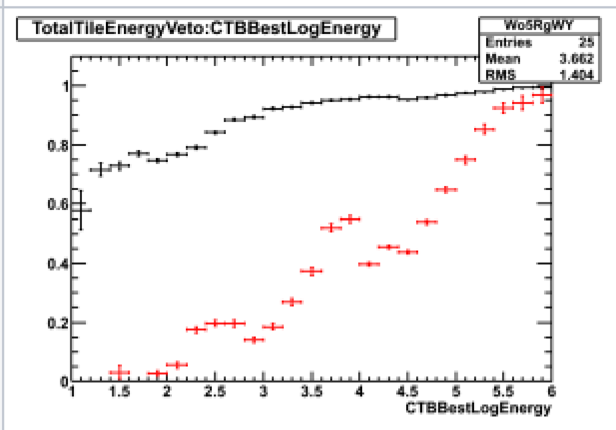
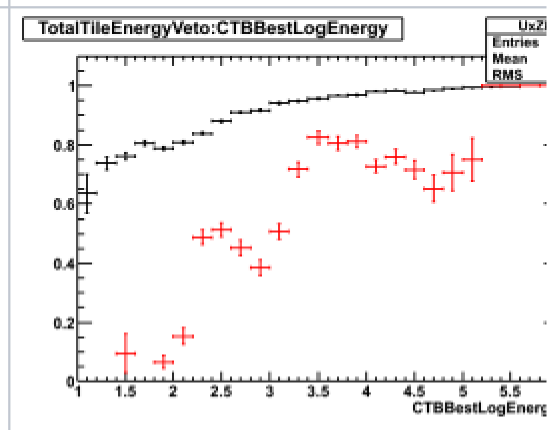
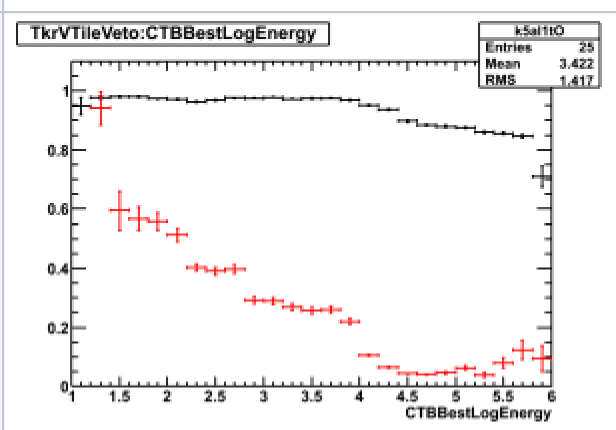
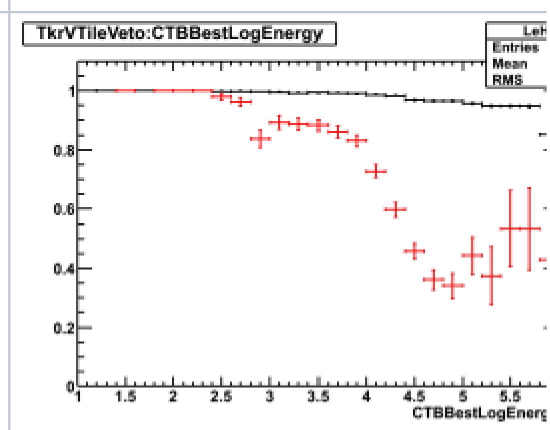
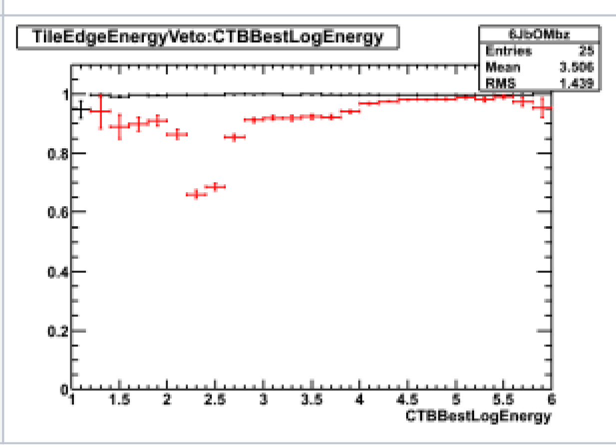
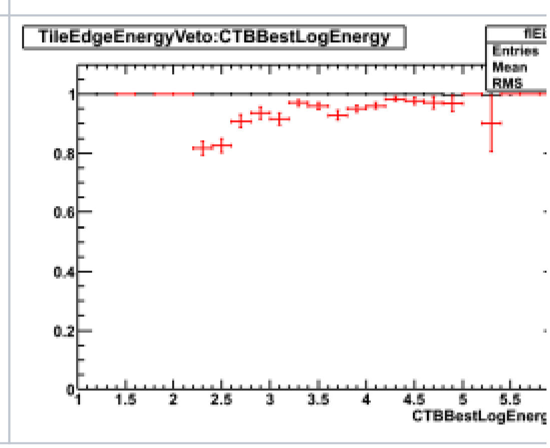
The net effect on the signal and background efficiency is small...

CUT	SIG	BKG
TileEdgeCut_p7	83.96% (99.96%) – 102135	4.755% (99.93%) – 8522

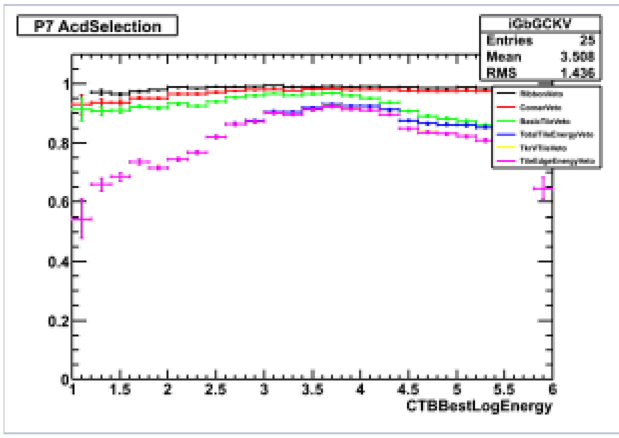
and I will use these final Pass 7 cuts in subsequent comparisons with Pass 8 data.

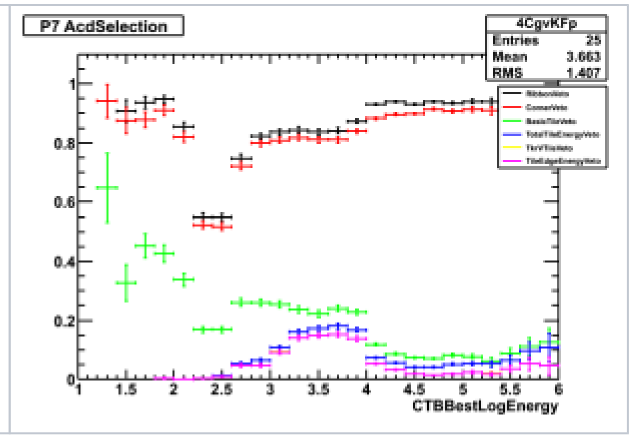
Selection Efficiency as a Function of Energy

Name	Individual Selection	Selection with Respect to Complement
RibbonVeto		
CornerVeto		

<table>
<tr><td>BasicTileVeto</td><td>
</td><td>
</td></tr>
<tr><td>TotalTileEnergyVeto</td><td>
</td><td>
</td></tr>
<tr><td>TkrVTileVeto</td><td>
</td><td>
</td></tr>
<tr><td>TileEdgeVeto</td><td>
</td><td>
</td></tr>
</table>

Sequential Cuts (Sig)	**Sequential Cuts (Bkg)**





Pass 8 Events

Data Set

The Pass8 event samples I use come from the standard Pass8 MC data sets with overlays corresponding to GR-v19r4p1gr13.

```
#All Gamma
root://glast-rdr//glast/mc/ServiceChallenge/AG-GR-v19r4p1gr13-FAKEOVL/merit/AG-GR-v19r4p1gr13-OVL-*-merit.root
#Background
root://glast-rdr//glast/mc/ServiceChallenge/BKG-GR-v19r4p1gr13-OVL/merit/BKG-GR-v19r4p1gr13-OVL-*-merit.root
```

AcdRecon vs AcdReconV2

We apply the same pass7 cuts to the Pass8 data. As expected, the cuts do not perform as well. Specifically, the BasicTileCut_p7 has lost background rejection power and the TotalTileEnergyCut_p7 has decreased efficiency for signal.

PASS7	SIG	BKG
RibbonCut_p7	99.01% (99.01%) – 60482	89.9% (89.9%) – 50790
CornerCut_p7	97.99% (98.96%) – 59856	86.69% (96.43%) – 48978
BasicTileCut_p7	92.92% (94.83%) – 56762	22.79% (26.29%) – 12875
TotalTileEnergyCut_p7	82.87% (89.18%) – 50623	11.72% (51.42%) – 6620
VetoTileCut_p7	79.49% (95.91%) – 48555	8.284% (70.69%) – 4680
TileEdgeCut_p7	79.45% (99.95%) – 48533	8.273% (99.87%) – 4674

Translating the Pass7 cuts to the new Acd2 variable equivalents, we find comparable results. This means that from the ACD precut point of view, we can turn off the old AcdReconAlg without any significant loss of efficiency. As a note, this study uses roughly 15% of the allGamma events and 25% of the background sample.

PASS7 ACD2	SIG	BKG
RibbonCut2_p7	99.01% (99.01%) – 60482	89.9% (89.9%) – 50790
CornerCut2_p7	97.99% (98.96%) – 59856	86.69% (96.43%) – 48978
BasicTileCut2_p7	92.92% (94.83%) – 56760	22.73% (26.22%) – 12843
TotalTileEnergyCut2_p7	83.05% (89.38%) – 50732	11.96% (52.63%) – 6759
VetoTileCut2_p7	79.64% (95.89%) – 48649	8.443% (70.57%) – 4770

TileEdgeCut2_p7	79.61% (99.95%) – 48627	8.433% (99.87%) – 4764

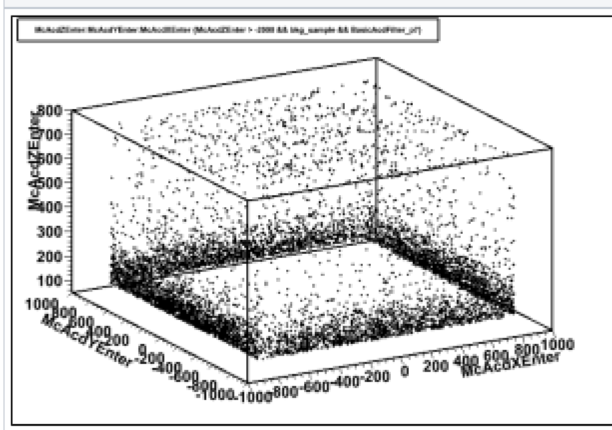
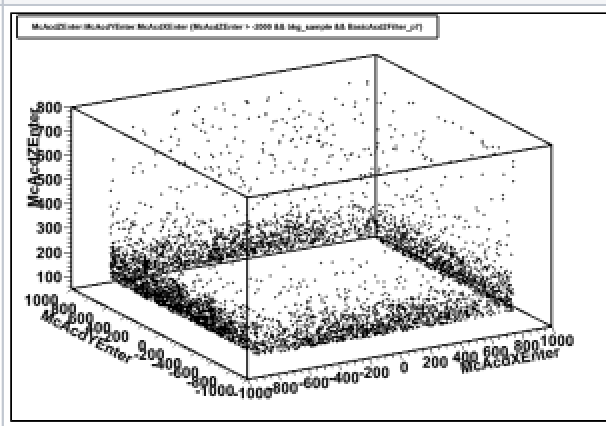
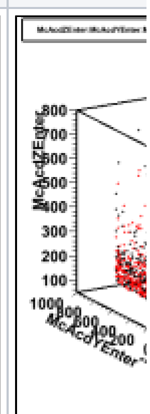
Pass 8 Cuts

Correcting the gamma efficiency of TotalTileEnergy Cut and appending the Cal1ConeCut_p8 at the end. It can be seen that without the Cal1ConeCut, nearly twice as much background sneaks through. The power of the Cal1ConeCut is quite impressive.

PASS7 ACD2	SIG	BKG
RibbonCut2_p7	99.01% (99.01%) – 60482	89.9% (89.9%) – 50790
CornerCut2_p7	97.99% (98.96%) – 59856	86.69% (96.43%) – 48978
BasicTileCut2_p7	92.92% (94.83%) – 56760	22.73% (26.22%) – 12843
TotalTileEnergyCut2_p7	86.05% (92.61%) – 52566	13.42% (59.03%) – 7581
VetoTileCut2_p7	82.59% (95.98%) – 50452	9.709% (72.35%) – 5485
TileEdgeCut2_p7	82.54% (99.94%) – 50421	9.693% (99.84%) – 5476
Cal1ConeCut_p8	80.34% (97.33%) – 49073	4.461% (46.02%) – 2520

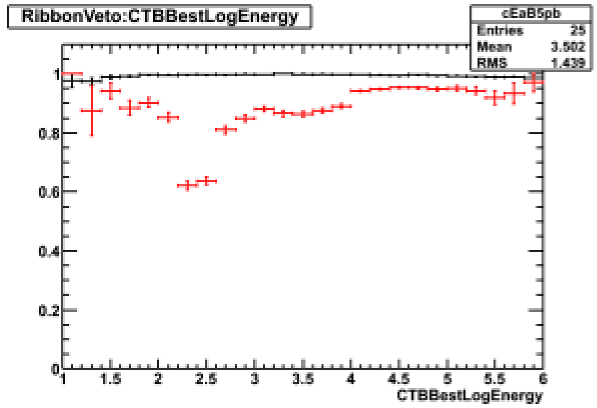
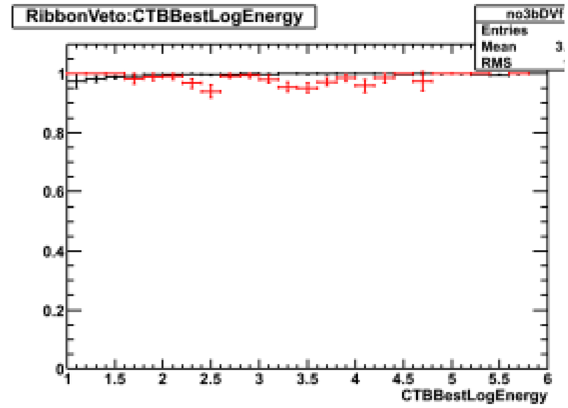
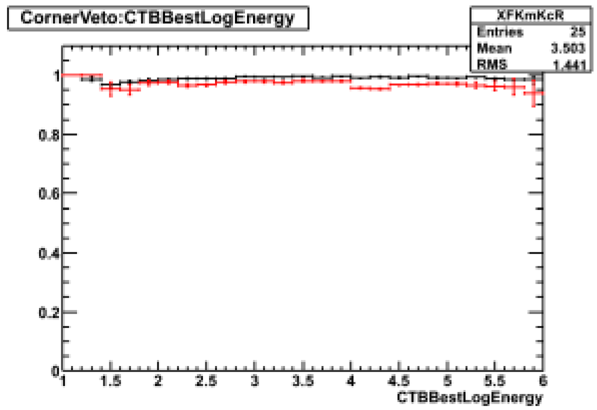
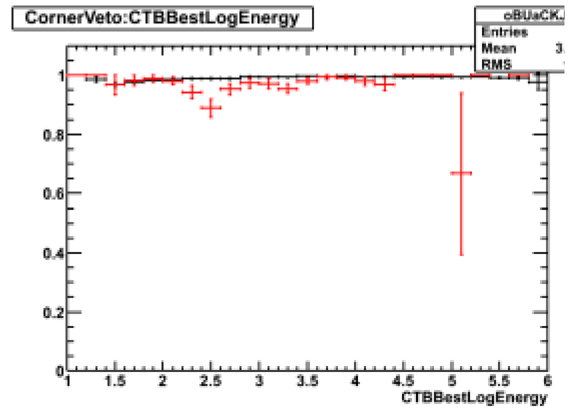
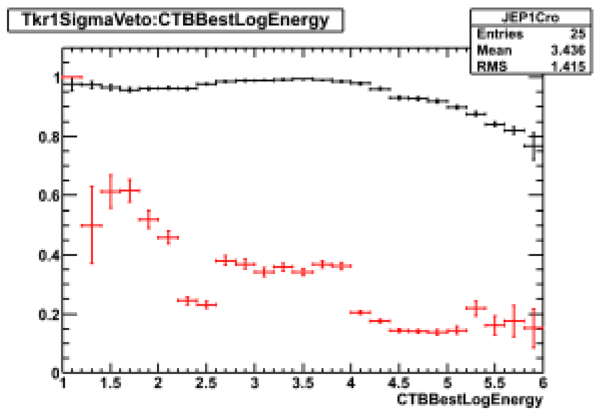
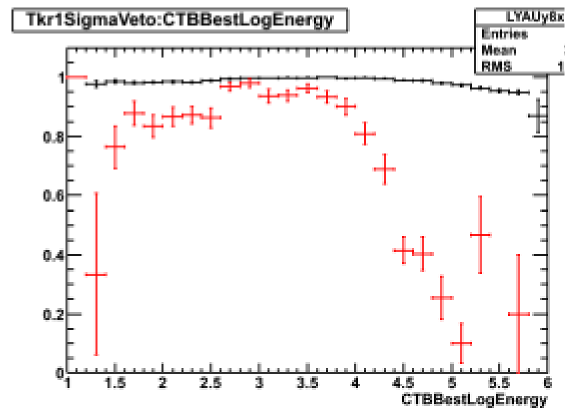
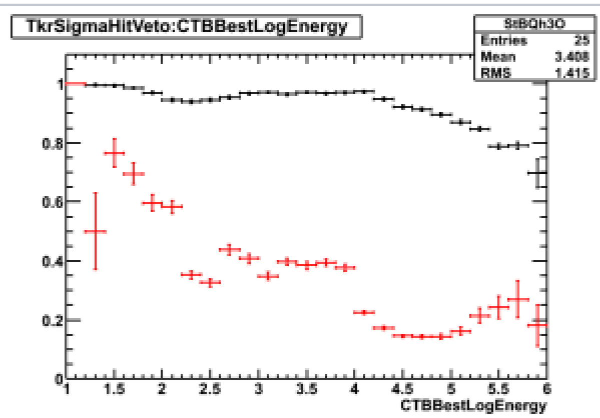
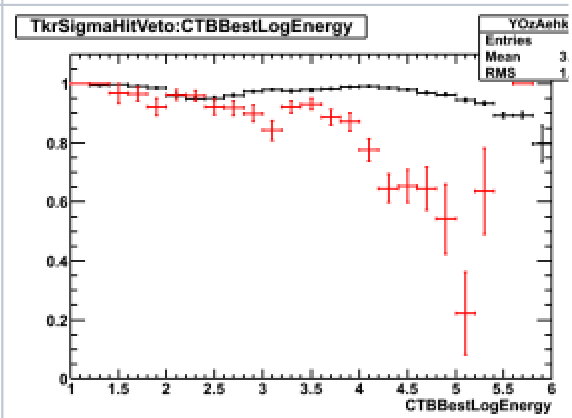
PASS8	SIG	BKG
RibbonVeto_p8	99.5% (99.5%)	89.9% (89.9%)
CornerVeto_p8	98.6% (99.1%)	86.9% (96.6%)
Tkr1SigmaVeto_p8	93.3% (94.7%)	22.5% (25.9%)
TkrSigmaHitVeto_p8	88.7% (95%)	16.6% (73.7%)
Tkr1ConeVeto_p8	83.7% (94.3%)	10.5% (63.6%)
Cal1ConeVeto_p8	81.7% (97.6%)	4.96% (47.1%)

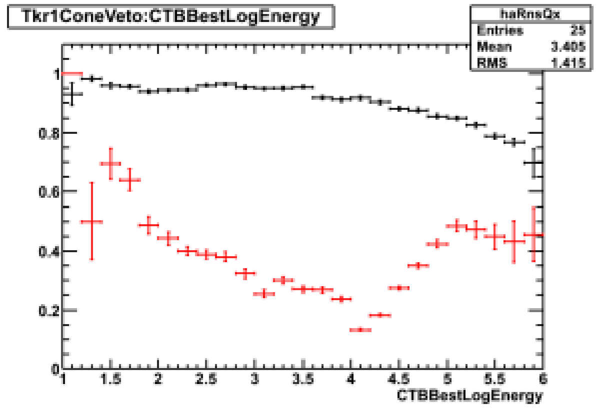
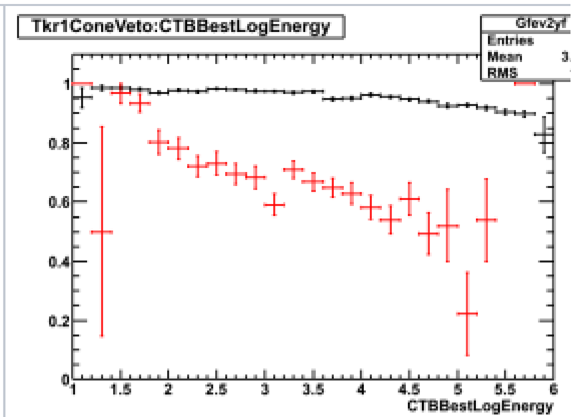
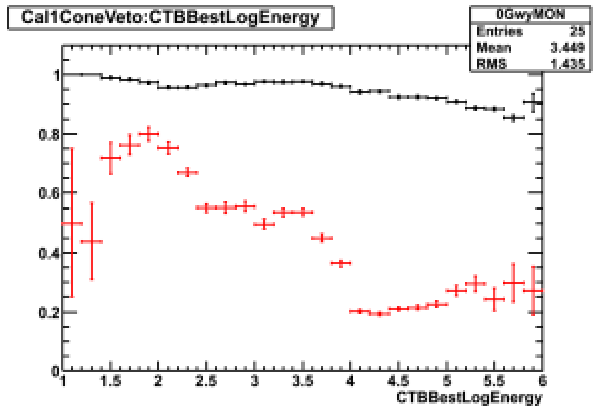
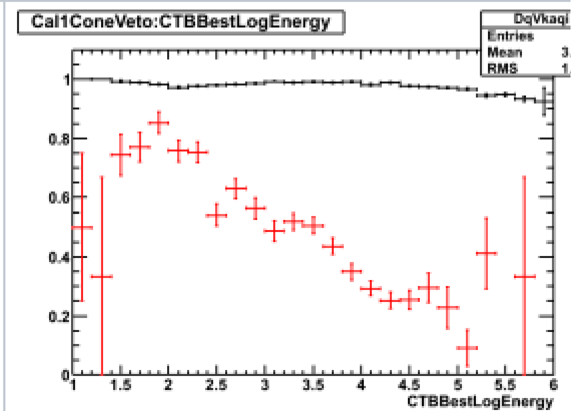
The power of the Cal1ConeCut comes from the fact that the majority of the residual background is sneaking through the bottom of the TKR. The direction of these events is poorly reconstructed since, especially at high energy, the longest straightest track will often come from back-splash. On the other hand, the CAL axis will point along the direction of the incident particle causing the ACD to query the proper tiles.

P7 Residual BKG	P8 Residual BKG (P7 CUTS)	P8 Residual BK
		

Selection Efficiency as a Function of Energy

Name	Individual Selection	Selection with Respect to Complement

RibbonVeto		
CornerVeto		
Tkr1SigmaVeto		
TkrSigmaHitVeto		

Tkr1ConeVeto		
Cal1ConeVeto		

Sequential Selections (Sig)	**Sequential Selections (Bkg)**