The DEA cut

Konstantin Olchanski, olchansk@b1.phy.bnl.gov v.1.0, 1996-Aug-25

Herein lays the ultimate documentation on the DEA cut. The TeX-formatted Postscript version of this document suitable for printing is *here* <http://lemond.phy.bnl.gov/documentation/detectors/DEA/deaCut.ps>

Contents

1 Purpose of the DEA cut

The purpose of the DEA cut is to veto the events that have photons that miss the LGD and the CSI.

As can be easily seen from a drawing of the E852 experimental setup, ideally, CSI, DEA and LGD give E852 an almost hermetic photon detection capability.

The photons either hit the LGD and are measured, or hit the CSI and cause the event to fail the CSI cut or hit the DEA and cause the event to fail the DEA cut. Ideally, after the DEA and CSI cuts, the final data sample will not contain events with unmeasured photons.

Many different approaches can be used to veto the events that have photons hitting the DEA.

The simplest DEA cut would veto events that have hits in any of the DEA segments. It's drawback is that it will also veto the events where charged patricles go through the DEA, leading to an excessive loss of events.

A more sneaky cut would try to recover some of these events by looking at hits in CPVC. In this approach it is assumed that a charged track will fire both a CPVC segment and the corresponding DEA segment. The cut would ignore all the DEA segments where the corresponding CPVC segment has a hit (assuming it was a charged particle) and only would only look at the DEA segments where the corresponding CPVC segment did not fire. If any of such DEA segments has a hit, the event is vetoed.

	DEA hit	no DEA hit
track hit	keep	keep
no track hit	veto the event	keep

Table 1: Table 1: DEA segment cut rules

This cut was fairly successfully used to analyze the 1994 data. However a couple of problems were discovered- it turned out that there are "holes" between CPVC and DEA, so that charged particles manage to hit DEA segments without hitting the corresponding CPVC segments. Such events foul the cut and get erroneously vetoed, creating a "picture frame" effect on the CPVC Z-plane. The other problem was discovered while processing the 1995 data. It turned out that the bottom CPVC segment (channels 35 and 36) did not work very well. The ADC signals are not large enough and create a problem in telling real CPVC hits from pedestal fluctuations. For more details, see the *report* `<../cpvc/cpvc.html>` on the CPVC and DEA performance during the 1995 data run.

To solve these two problems, a more advanced cut was implemented. Instead of using CPVC, it relies on the reconstructed charged tracks to determine which segments of the DEA are hit by charged particles.

In addition to fixing (avoiding) the problems with CPVC, it also helps to veto events with missing charged particles. If the reconstruction program failed to find a charged track, and this track makes a hit in the DEA, the event will be vetoed by the DEA cut, unless the same segment is hit by another, correctly reconstructed track.

The details of the implementation and performance of this tracking-based DEA cut is discussed in the rest of this paper.

2 Implementation of the DEA cut

The DEA cut is done in two steps: first, tag the DEA segments, then, either pass or veto the event.

The tagging of the DEA segments is done in the function *deaCut_tagDEA()*. It determines which segments of DEA are hit by charged tracks by swimming the tracks stored in *GROUP_TrackPositions* to the DEA front and back faces and checking which DEA segments are hit. The result is encoded into an 8-bit bitmap (4 bits for the front and 4 bits for the back of DEA, each of the 4 bits corresponding to the top, left, right and bottom DEA segments). This 8-bit bitmap is then stored into the bits 8 through 15 of the DEA bitmap of *GROUP_VETO_BITMAP*.

The results of the tagging can be seen with the event display program- the tagged segments are plotted in a special color (green by default).

The decision to pass or veto the event is made in the function *ProcessDEACuts*. This function looks at all the DEA segments that were not hit by charged tracks. If any one of them has a hit (presumably from a photon), the event is vetoed.

The vetoing decision rule for one segment is given by in table 1.

3 Function calls

The functions are declared in the header file *veto.h*, their sources live in the *libanalyzer/dealib.c* and the object files are in *libUtil.a*.

Since the DEA cut relies on trackign charged particles, the MPS field map has to be initialized prior to calling the DEA cut functions.

3.1 int ProcessDEACuts(itape_header_t* event)

This is the main function. It does all the DEA-cut processing and returns *1* if the event passes the cut and *0* if it fails.

This function uses/requires the following data groups:

- *GROUP_VETO_BITMAP*
- *GROUP_TrackPositions*

3.2 int deaCut_tagDEA(itape_header_t *event);

This function performs tagging of the DEA with the reconstructed charged tracks. It is automatically called from *ProcessDEACuts()*.

This function uses/requires the following data groups:

- *GROUP_VETO_BITMAP*
- *GROUP_TrackPositions*

4 Effect of the DEA cut on the missing mass distribution

If the DEA cut functions correctly and vetoes events with photons that have missed the LGD, we expect to see an effect of the DEA cut on the missing mass distribution.

The events with missing photons will tend to have a missing mass greater than the mass of the proton/neutron, so we would expect the DEA cut to remove events with high missing mass and keep events with low missing mass.

Three distributions are shown on figure 1. The unshaded plot shows the distribution of the square of the missing mass for a typical sample of 1995 14 GeV 0-2-2-LGD events with one positive, one negative charged tracks, and two photons. The plot with horizontal stripes shows the events removed by the DEA cut. The plot with 45 degree stripes shows the events kept by the DEA cut.

The conclusion is that the DEA cut is working more or less as expected.

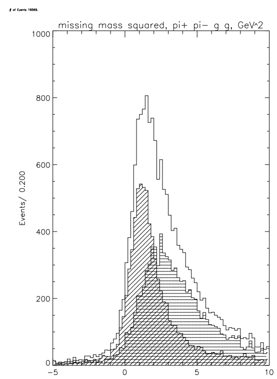

Figure 1: Figure 1: Missing mass squared

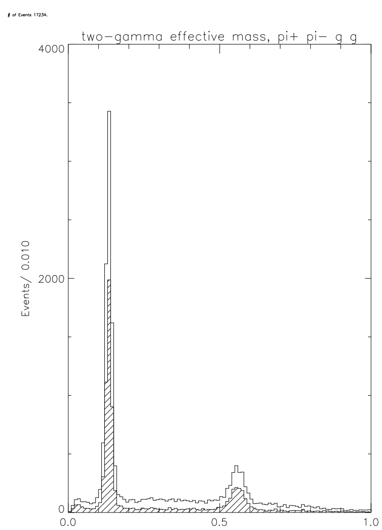

Figure 2: Figure 2: Two-photon effective mass

5 Effect of the DEA cut on the gamma-gamma effective mass distribution

If the DEA cut functions correctly, it also should have an effect of the two-photon effective mass distributions.

If we select a data sample with 2 photons in the LGD and plot the two-photon effective mass, the distribution contain a background from events that initially had more than 2 photons, but the extra photons missed the LGD. The DEA cut is expected to remove events with "missing" photons, so it can be expected that the DEA cut will reduce the background on the two-photon effective mass distribution, and so decrease the background under the pi0 and the eta.

Figure 2 shows the two-photon effective mass distribution for a typical sample of 1995 14 GeV 0-2-2-LGD events with two photons detected in the LGD. The unshaded plot contains all events before the cut and the shaded plot shows the events left after the DEA cut.

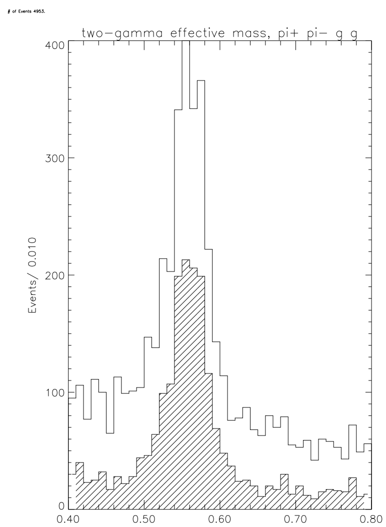

Figure 3: Figure 3: Two-photon effective mass, in the *eta* region

One can see that while the height of the pi0 and eta peaks decreases by a factor of two, the background decreases by about a factor of 3.

Figure 3 contains the same data as figure 2, but expanded around the eta peak.

The conclusion is that again, the DEA cut works as expected.