CB Note 199
9/7/1997

Splitoff recognition with Dolby-C

N P Hessey
CERN

Abstract

This note describes the Dolby-C splitoff recognition program: both how it works and how to use it. Everybody who analyses data with photons should read it.

Contents

1. Introduction

Dolby-C is a method for deciding which PEDs should be attributed to a photon and which should be attributed to a splitoff.

The terms cluster, PED, and local maximum as used in the Crystal Barrel reconstruction can be briefly defined as follows: A "cluster" is any region of contiguous crystals which have energy in them; A "local maximum" is a crystal in a cluster which has more energy than all its neighbouring crystals; A cluster has one or more "PEDs" (an acronym for Particle Energy Deposit), according to how many local maxima are in the cluster.

What is a splitoff? A single photon on interacting with the crystals produces a shower which in general spans several touching crystals. This shower is usually isolated from other showers, and has a single local maximum, at the crystal the photon first entered. Sometimes two photon showers can spread out to touch each other, giving a large cluster with two PEDs. Hence usually a PED corresponds to a photon. Unfortunately, sometimes a single photon can produce two PEDs because the shower formation is stochastic. In such cases, the extra PED is known as an electromagnetic splitoff. Charged hadrons also form PEDs, which are recognisable because the JDC has a track pointing to the PED. Again, a charged particle can produced secondary PEDs known as hadronic splitoffs.

It is important in data analysis to be able to distinguish which class a PED is from. GTRACK, the global tracking code, classifies PEDs as matched – i.e. due to charged particles – or unmatched, i.e. a photon candidate. Dolby-C attempts to classify the latter as due to photons or splitoffs. Dolby-C was developed on zero-prong triggered data, and so is optimised for finding electromagnetic splitoffs. However, it should also help find some hadronic splitoffs too. Further tests are needed for the latter.

In standard CBOFF analysis up to now (August 1992), any PED with less than 20 MeV was treated as a splitoff and ignored, whilst any PED over 20 MeV was treated as a photon. This is inaccurate: there are splitoffs above 20 MeV and photons below 20 MeV. Dolby-C is a significant improvement on this. For instance, it allows the threshold for photon detection to be reduced from 20 MeV to 10 MeV. It was originally developed for DSTprime production of zero-prong triggered data, where photon multiplicity is the main criterion for splitting the data. It is in many ways very similar to the original Dolby splitoff-reduction method as invented by Sven Dombrowski ("Dolby-A") and an improvement made on that by Marcus Englert ("Dolby-B").

Dolby-C is based on (i) the observation that splitoffs are usually close to the parent PED and of lower energy than it, and (ii) that the majority of cases where two photon-induced PEDs are close together are due to π^0 decays. One can make a two dimensional histogram for all pairs of PEDs of $(1-A^2)$ against $(1-\cos\Psi)$ (figure 1), where A is the energy decay asymmetry, which is defined in terms of the PED energies E_1 and E_2 by

$$A = \frac{(E_1 - E_2)}{(E_1 + E_2)} \tag{1}$$

and ψ is the angle subtended at the target by the two PEDs, called the opening angle from now on. On such an "asymmetry-opening angle plot" splitoffs mostly appear in the bottom left corner (high asymmetry and small opening angle), which is a region excluded from the phase space of π^0 decays for π^0 energies found in the Crystal Barrel experiment. The lower energy PED of any pair of PEDs which lies in this part of the histogram is labelled a splitoff by Dolby-C.

2. The Dolby-C Method

In the asymmetry-opening angle plot π^0 decay photons from monoenergetic π^0 lie on hyperbolae. This can be seen from the formula for the invariant mass of a particle decaying to two photons:

$$\begin{aligned} {m_{\pi^0}}^2 &= 2E_{\gamma_1}E_{\gamma_2}(1-\cos\Psi) \\ &= \frac{1}{2}E^2(1-A^2)(1-\cos\Psi) \end{aligned} \tag{2}$$

where $E = E_{\gamma_1} + E_{\gamma_2}$ is the total pion energy. For a given E this is the equation of a hyperbola:

$$1 - A^2 = \frac{\text{const.}}{(1-\cos\Psi)} \tag{3}$$

The higher the pion energy, the closer the hyperbola gets to the origin. In $\bar{p}p$ annihilitions at rest, the highest π^0 energy is about 940 MeV. This leaves a region below the 940 MeV hyperbola which is forbidden for π^0 decay photons,

Figure 1 Plot of $1 - A^2$ vs $1 - \cos\Psi$ for zeroprong data. Dolby-C interprets entries in the cross-hatched area as being due to splitoffs.

apart from resolution effects. This region is only populated by pairs of photons from different hadrons ("random overlaps", a small contribution), and splitoffs.

The contribution from random overlaps increases with photon multiplicity. This can be seen in figure 2 .

Figure 1 shows that in real data there are a lot of entries in the bottom left part of the forbidden region. These are mostly splitoffs. Unfortunately splitoffs spread into the π^0 region, so cannot be identified with certainty.

The region in which PEDs are considered to be splitoffs can be defined by the user. It is bounded by three lines: below a certain $1 - A^2$, to the left of a certain $1 - \cos\Psi$, and below a hyperbola for a certain π^0 energy. The user can also specify a maximum energy for splitoffs. Table 1 gives a reasonable set of cuts. These are illustrated in figure 1. The hyperbola was chosen to be for π^0 of 1150 MeV total energy, because resolution effects put 940 MeV π^0 from $\bar{p}p \to \pi^0\pi^0$ into the region between these two hyperbolae. Users may want to enlarge or reduce the splitoff region for their particular final state.

The fifth cut in Dolby-C sets a minimum energy for a PED to be considered a photon. This is effectively the photon detection threshold. Figure 3 shows that below 10 MeV most PEDs are not due to photons, since the photon energy spectrum is known to decrease to zero around 4 MeV. Most users will want to totally ignore PEDs below this cut. However, some users may want to exclude events with such PEDs to obtain a cleaner sample of some final state.

3. User Guide to the Dolby-C software

Dolby-C routines use only global track data (TTKS bank). So there is no need to retrack DST and DSTprime data. However, one of the main advantages of Dolby-C is that it allows the photon energy threshold to be lowered from 20 MeV to 10 MeV. To benefit from this on old data sets – ones with ECLUBC = EPEDBC = 20 MeV – the crystal data must be retracked. Suitable cuts are ECLUBC = 4 MeV and EPEDBC = 10 MeV.

There are two routines that the user must call. These are DBCINI to initialise the cuts, and DBCSPL to identify splitoffs. The user may also want to call a third routine, DBCNOS, which counts the multiplicities of splitoffs,

Figure 2 $1 - A^2$ vs. $1 - \cos\Psi$ for 10,000 phase-space n-π^0 events. (a) 2 π^0 (b) 3 π^0 (c) 4 π^0 (d) $5\pi^0$ events.

Figure 3 Energies of PEDs not found to be splitoffs; all-neutral data. The threshold for detection is around 10 MeV.

Dolby-C cut	Suggested Value
Max. splitoff energy	100.0 MeV
Max. splitoff $1 - A^2$	0.45
Max. splitoff $1 - \cos\Psi$	0.2
Max. implied π^0 energy	1150.0 MeV
Gamma threshold energy	10.0 MeV

Table 1. Suggested cuts for Dolby-C.

gammas etc. A fourth routine, DBCCOA, is used by DBCSPL to calculate the cosine of the opening angle. Users do not generally need to call this.

The calls communicate via a Fortran common block /DBC/, which stores the cuts. Note the cuts are not stored in the form supplied by the user.

In the following sub-sections, subroutine parameters preceded by an asterisk are read by the subroutine, and those followed by one are written to by the subroutine.

3.1 DBCINI

```
SUBROUTINE DBCINI(*EMAX, *APARAM, *PSIPAR,
  *EPIMAX, *ETHRES)
REAL EMAX, APARAM, PSIPAR, EPIMAX, ETHRES
```

Initialise Dolby-C and set cut levels. Table 1 gives suitable values for the cuts (in the order required by DBCINI). See the previous section for the function of each cut. This routine also prints out the cuts supplied, on unit LLOG (taken from the /CBUNIT/ common block of the standard CBOFF software). This printout may be removed in future releases if people prefer.

Usually called from the USINIT routine.

SUBROUTINE DBCSPL(GTTYPE$*$, $*$LENGTH, NTRAKS$*$)

INTEGER GTTYPE($*$), LENGTH, NTRAKS

Usually called from USER, for each event. This routine decides what type each PED is – splitoff, gamma etc.

GTTYPE should be an array with LENGTH elements. LENGTH = 50 is usually adequate. DBCSPL will not write beyond element LENGTH of GTTYPE. The number of elements actually written is returned in NTRAKS, and is the lesser of LENGTH and the number of TTKS tracks. DBCSPL writes a code for each TTKS track into the corresponding GTTYPE element.

The codes show what type the tracks are, as detailed in table 2. Splitoffs are given a code greater than 0. The code corresponds to the PED that Dolby-C thinks is its parent. This allows users who want to, to add the splitoff energy back to the parent. All other track types are 0 or less.

Track type	Code returned in GTTYPE
Due to photon	0
Due to charged track	−1
Sub-threshold	−2
Splitoff	TTKS-track-number of parent PED

Table 2. Track type codes returned by DBCSPL.

In detail, the classification is: If the track is charged according to its TTKS bank data (word 4), set GTTYPE to -1. Else, if the PED energy is greater than EMAX set it to 0. Else, loop over all other TTKS tracks except charged tracks not matched to a PED and tracks with lower PED energy, and see if it is a splitoff of this track; if so, set GTTYPE to the TTKS id of this track, and stop looping. If not a splitoff of any track, set GTTYPE to 0. Finally, loop over all tracks identified as due to photons (code 0) and see if they are above threshold (ETHRES): if not, set their code to -2.

DBCSPL uses strictly the *crystal* (PED) information of TTKS banks. So it is compatible with PEDs matched to charged tracks. However, so far the method is largely untested on charged events. An unmatched charged track cannot be a parent of a splitoff and cannot be a splitoff. It is compatible with old (pre version 11990) and new versions of TTKS banks.

3.3 DBCNOS

SUBROUTINE DBCNOS($*$GTTYPE, $*$NTRCKS, NGOODG$*$,

NCHARG$*$, NSUBTH$*$, NSPLIT$*$)

INTEGER GTTYPE($*$), NTRCKS, NGOODG, NCHARG,

NSUBTH, NSPLIT

DBCNOS counts the multiplicities of the different particle types. GTTYPE and NTRCKS are as returned by DBCSPL. On return, NGOODG is the number of good gammas, NCHARG is the number of charged tracks, NSUBTH is the number of non-charged, non-splitoff PEDs below the energy threshold cut, and NSPLIT is the number of splitoffs.

3.4 DBCCOA

REAL FUNCTION DBCCOA(THT1$*$, PHI1$*$, THT2$*$, PHI2$*$)

REAL THT1, PHI1, THT2, PHI2

Called by DBCSPL to calculate the cosine of the opening angle between two PEDs. THTi and PHIi are the PED directions in spherical polar coordinates (e.g. supply TTKS data words 44 and 43). The calculation uses double precision routines.

3.5 Using the code

The results of DBCSPL can be used in various ways. The two main ways envisaged are: (i) to keep events with splitoffs, but ignore PEDs due to splitoffs. This gives high acceptance, and clean events. (ii) ignore events which have splitoffs. This gives very clean samples.

For most users, method (i) is recommended. However, it then needs care to make sure that only the correct PEDs are used in the analysis, namely those which Dolby-C interprets as good gammas. For example, if looking for π^0 π^0 events, one would select events with 4 good gammas. These may correspond to TTKS tracks 1, 3, 4, and 6. TTKS tracks 2 and 5 may be splitoffs. Then the user must use only tracks 1, 3, 4, and 6 in kinematic fitting etc.

Users may also want to recombine PEDs interpreted as splitoffs with their parents. For example, the splitoff energy could easily be added to the parent energy, leaving the parent direction unchanged. Care should be taken to update all relevant data in the TTKS bank of the parent, and to avoid skimming adjusted banks to output and repeating the process when you reanalyse the output!

3.6 Linking to the code

The compiled object code is included in the BCTRAK object code library, and so most users will find their code links with their normal script/exec/kumac etc.

4. Performance

In the following, the “standard method” means version 1.43/08 of BCTRAK, with ECLUBC = EPEDBC = 20 MeV.

With cuts as in table 1 (but with ETHRES 0), Dolby-C reduces the energy threshold for detection of photons from about 20 MeV to 10 MeV. That is, an unmatched PED above 10 MeV which is not found to be a splitoff is more likely to be a gamma than anything else. The evidence for this is in figure 3.

Samples of seven gamma data tend to have a large background contribution from six gamma events, with one of the 6 gammas giving a splitoff. Dolby-C can give striking improvements in purity of final events selected for such data. This is true *even after 7C kinematic fitting.* Figure 4 compares Dolby-C with the standard method for events which pass a 7C kinematic fit to $\bar{p}p \to 3\pi^0\gamma$. The energy spectrum of the gamma shows an increase due to splitoffs at low energies. This increase is very much smaller for the Dolby-C method.

Dolby-C improves photon multiplicity counting over the standard method. Figure 5 shows the multiplicity with standard and Dolby-C methods. The peaks at 6 and 10 gammas are enhanced relative to most other multiplicities when Dolby-C is used. The standard method tends to spread N-gamma events over $(N-1)$ (due to the high threshold) and $(N+1)$ (due to counting splitoffs as gammas). This reduces the spikes at 6 and 10 in the multiplicity distribution, relative to other multiplicities.

Dolby-C misinterprets some photons as splitoffs. This effect increases as photon multiplicity increases. The fractions of events where one or more photons are interepreted as splitoffs are 0, 6%, 12% and 15% for 2, 3, 4, and 5 π^0 events. These numbers are the results of phase-space only Monte Carlo simulations (GENBOD), with no cuts and no resolution folded in. Many of these misinterpreted gammas are below 20 MeV, and so would also be treated as splitoffs in the standard method. Furthermore, the probability that a photon produces a splitoff in the standard method is around 0.05. Hence for 10 gamma events the standard method loses $1 - 0.95^{10} = 40\%$ through splitoffs. So as long as Dolby-C recovers a significant fraction of these, it will improve the net efficiency. The probability of misinterpretation increases as the photon energy decreases. Figure 6 quantifies this for $3\pi^0$ events.

The purity of the 6-photon events is somewhat higher with Dolby-C than the standard method. This is seen from figure 7 which shows two-photon invariant masses for all photons which do not combine with any other photon to make a π^0 or η. The peak at about 770 MeV is due to $\omega \to \pi^0\gamma$, with one of the π^0 decay photons missed because it is low energy. These arise from the $\pi^0\pi^0\omega$ going to 7 photons final state etc. This peak is reduced in the Dolby-C case because of the lower energy threshold. The improvement is limited

Figure 4 Non-π^0 gamma energy for events satisfying a 7C fit to $3\pi^0\gamma$ events. (a) Standard method (b) Dolby-C method.

Figure 5 Photon multiplicities from all-neutral events with total four-momentum consistent with $\bar{p}p$ annihilations at rest. Broken line – standard method. Solid line – Dolby-C.

Figure 6 Probability that Dolby-C misinterprets a photon as a splitoff, as a function of photon energy. Derived from $3\pi^0$ GENBOD events.

because about 30% of the photons between 10 and 20 MeV in $\pi\pi\omega$ events are misinterpretted as splitoffs.

Note that the loss of signal events due to misinterpretation is very well modelled in Monte Carlo; it mainly depends on the four-momenta of the particles, with some secondary dependance on resolution etc. There is no direct dependence on shower shape etc. On the otherhand, the production of splitoffs may not be well modelled in CBGEANT; it depends critically on the detailed microscopic interactions of the electromagnetic showers in matter. So having a splitoff finder, developed on data and based on kinematics as here, will give more reliable estimates of efficiencies. To illustrate what I mean, consider the following example (where I make no claim as to how realistic the numbers are):

Suppose (unknown to us) 30% of $3\pi^0$ events have splitoffs in data using the standard method, and (known to us) only 15% have splitoffs in CBGEANT. Then without Dolby-C, the branching ratio estimate is wrong by $(1-0.15)/(1-0.30) = 21\%$. Suppose Dolby-C finds most splitoffs, but still leaves 10% of events with splitoffs in real data and 5% in Monte Carlo. Then the mistake is only $(1-0.05)/(1-0.10) = 5\%$, a considerable improvement.

Figure 7 gamma-gamma invariant masses for 6 photon events. hatching – Dolby-C. The peak for "$\omega \rightarrow \gamma\gamma$" at about 770 MeV is reduced in the latter case.

Addendum 1 to CB Note 199
9/7/1997

Splitoff recognition with Dolby-C: Storing the results with DBCSTO

N P Hessey
CERN

Abstract

A new routine, DBCSTO, has been added to the suite of Dolby-C routines. It stores the results of Dolby-C in the Global track banks (TTKS banks). This note describes what it does and how to use it. Dolby-C may in future be incorporated into standard production, in which case people who cannot be bothered to find out what they are doing can just read this addendum and will know how to use the results of Dolby-C without ever calling a Dolby-C routine.

1. Addendum

Dolby-C is a method for deciding which PEDs should be attributed to a photon and which should be attributed to a splitoff. It consists of a set of user callable routines. Previously, the results of Dolby-C were not stored on tape. Now they can be, using a new routine called DBCSTO. It was introduced at version 1.05 of Dolby-C.

1.1 DBCSTO

SUBROUTINE DBCSTO($*$GTTYPE, $*$NTRAKS)

INTEGER GTTYPE($*$), NTRAKS

Usually called from USER, for each event. This routine stores the results of DBCSPL.

GTTYPE and NTRAKS are as returned by DBCSPL. The routine checks for the existence of the global track banks and checks that they were produced by version 1.21/09 or later of the Global Tracking software. If all is well, it stores the value of GTTYPE(I) in the I'th TTKS bank, at word 52. It also stores MAX(0, GTTYPE(I)) in word 53. Michael Doser claims the repetition is a good system. I ranges from 1 to NTRAKS, unless NTRAKS is bigger than the number of TTKS banks when it stops at the last bank.

Recall that GTTYPE(I) is -2 for PEDs below the energy threshold, -1 for PEDs matched to charged tracks, 0 for photons, and is the TTKS track number of the parent for splitoffs.