A workaround for EPOS: $E^2 \neq m^2 + \vec{p}^2$

August 30, 2018

1 Summary

In Cosmos, there are 3 versions of EPOS. They have a minor bug; at high energies, some of generated particles break the relation

$$E^2 = \vec{p}^2 + m^2 \tag{1}$$

more than that expected from normal numerical computation accuracy.

The effect on usual air shower simulations is expected to be negligible. However, in some applications, it must be corrected. We use a workaround employed in the CRMC package (https://github.com/alisw/crmc).

The three models can be seen in Cosmos/Import/Hidden/EPOS as:

epos-lhc-3400
epos-lhc-3700
epos199

Inside each of these directories, there is a directory, "Interface" which, in tern, contains cepos.f. In each of cepos.f, the user will find

```
      subroutine ceposGenOneEvent(pj, ia, iz, a, n)
```

and in this subroutine,

```
      do i = 1, nhep
          kf = idhep(i)
```

Between these two lines, the user may insert next two lines:

```
      phep(4,i)=sqrt( dot_product(phep(1:3,i), phep(1:3,i))
     *      +   phep(5,i)**2  )
```

so that the program will look like

```
      do i = 1, nhep
          phep(4,i)=sqrt( dot_product(phep(1:3,i), phep(1:3,i))
     *      +   phep(5,i)**2  )
          kf = idhep(i)
```

1

After this, the user may issue "intModel.sh" and select a desired model (even if it is already indicated as the current EPOS model). For Cosmos8.032, this modification is enough for the user to make it compatible with the Cosmos version 8.033, which includes this workaround. In v8.033, the workaround may be disabled by changing `#define  TUNE4M  1` to `#define  TUNE4M  0` at the top of cepos.f (see each cepos.f) for special purpose.

2 Story

A master course student K.Ohashi of the LHCf group in Nagoya Univ. found that some of the particles generated in the Cosmos package with the EPOS interaction model break the relation, Eq. 1.

His calculation using the CRMC package didn't show such a break so that a bug in Cosmos is suspected. However, it was found that EPOS itself keeps the feature observed in Cosmos. The reason why the break is not observed in CRMC is that the CRMC puts a workaround to modify the EPOS output so that the energy is recalculated using Eq. 1. The break happens more at higher energies, but the energy change by the equation is so small that the effect would be very small in air shower calculations. However, the break results in a serious error for invariant mass calculations etc. Therefore, we decided to introduce the same workaround in Cosmos.

2.1 Testing EPOS standalone

In the course of checking the problem, testing EPOS itself without using Cosmos at all was tried. So the directory, epos-lhc-3700, was copied in a directory completely different from Cosmos. A simple "example.f" contained there was tested. But the things didn't go well as "README" says. As a memorandum, Example.pdf is left.

How to run example.f in EPOS-LHC-3700 completely independently of Cosmos

0) Go to Cosmos/Import/Hidden/EPOS/ and
   cp -r epos-lhc-3700 somedir/

1) Go to somedir
   cp MakefileOrig Makefile
   All 3400 in the Makefile—>3700

2) Edit example.f, and change the write statement so that
   id, px,py,pz, E, sqrt(E^2-P^2) is written only if
   abs(idhep(i)) is 211(pion)
   make exampl
   will produce "EposEx"

3) Edit example.param and add lines below.

```
set istmax 0
application hadron
model LHC ! can be "epos" for old epos 1.99
set nevent 1000    !number of events
set modsho 1     !printout after modsho events
set ecms  7000    !energy per nucleon in cms system
set laproj 1     !proj Z
set maproj 1     !proj A
set latarg 1    !targ Z
set matarg 1    !targ A
```
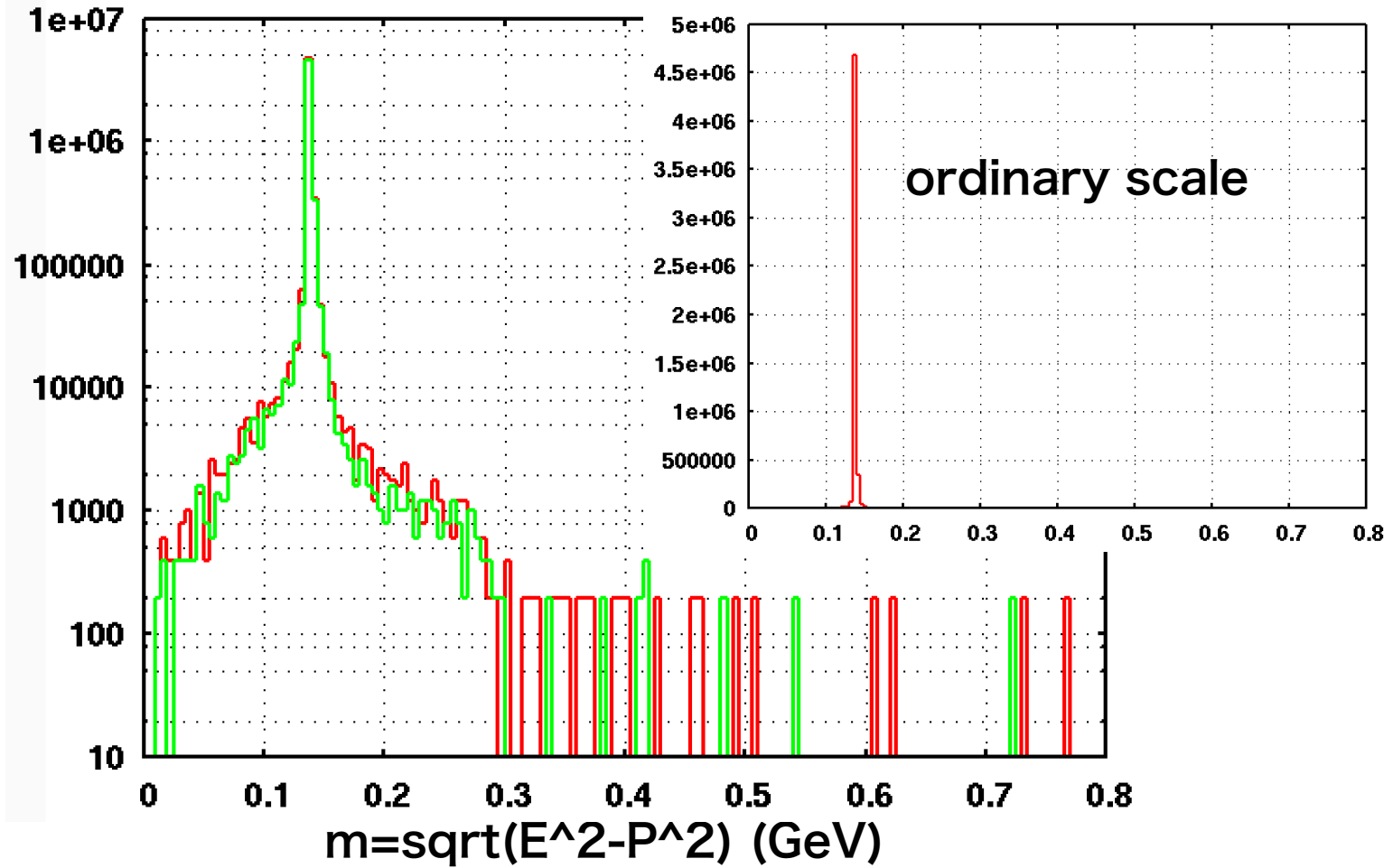
pp col. @ roots=7TeV

4) ./EposEx example.parm > outputfile

dN/dm (dm=0.005) 1000 interactions

ordinary scale

m=sqrt(E^2-P^2) (GeV)

	E^2>P^2	E^2< P^2:
pi+	26582	206
pi-	25961	121