Subject: [CLOSED] Specific event selection in Dpm generator
Posted by donghee on Thu, 09 Feb 2012 09:25:50 GMT

Hello everyone,

I'm trying to select and save a specific event type from DPM generator.

For instance, I would like to select and save only pi+pi- event from DPM generation.
In PndDpmDirect I can simply select those event after pid checking for generated event type.
Then I do not add any tracks into output file, if this condition is not satisfied. That mean only
pi+pi- event have track information, and other event types are set to zero tracks.

After this modification in FairPrimaryGenerator, PndDpmDirect file, I want to save only pi+pi-
event without any zero track event, which is marked by my decision, into my final output file.
Now, I have a trouble to discarding zero track from my output file.

I'm wondering, are there some possible and simple way to select and save specific event.
One general way is of course handling of source code dpm_gen_.f.
I am not prefer this way.
Second convenient way for me is control setting or option at FairPrimaryGenerator,
PndDpmDirect, and FairMCEventHeader after MC generation from DPM.

Thank you for your reading!
Donghee

Subject: Re: Specific event selection in Dpm generator
Posted by donghee on Thu, 09 Feb 2012 10:27:25 GMT

Hello everyone,

I solve this problem with quite simple trick using while loop only in PndDpmDirect code.(posted
below)
It might be useful that someone prepare a selector function inside PndDpmDirect in the future,
if it is not exsiting.
Anyhow, thank you again for your reading...

Quote:
```
// -----   Public method ReadEvent   --------------------------------------
Bool_t PndDpmDirect::ReadEvent(FairPrimaryGenerator* primGen) {

 int  npart, i;
 Double_t fX, fY, fZ, radius;
```

```
double Px[1000],Py[1000],Pz[1000],E[1000],Pm[1000],Wh[1000];
int Id[1000];

float Generator=0.; //  Format in which events are produced (0=pythia, 1=pluto)

Double_t weight = 1.0;
Int_t activeCnt=0;


Int_t NIDs=0;
Int_t event_type = 844;
Int_t particle_number = 4;
double PID_type_1 = 211;
double PID_type_2 = -211;

//4pi = 844
//2pi = 422

 /////////////////////////////////////////////////////////////////// /////////////////////////Donghee Kang
 //Selector!!!

 while(1){

   bool selector = false;
   // run generator
   dpm_gen__(&Generator, &fSeed);

   // Loop over all produced particles
   npart = lujets_.n;

  if(npart == particle_number){///////////////////////////////////////////Donghee Kang
    NIDs = 0;
    for (i= 0; i< npart; ++i) {
Id[i]=lujets_.k[i+1000];
     if( (Id[i] == PID_type_1) || (Id[i] == PID_type_2) ){
         NIDs += TMath::Abs(Id[i]);
     } //Id[i] check!!!
    }//loop
    if(NIDs == event_type){
     for (i= 0; i< npart; ++i) {
        Id[i]=lujets_.k[i+1000];
    Px[i]=lujets_.p[i];
        Py[i]=lujets_.p[i+1000];
        Pz[i]=lujets_.p[i+2000];
        Pm[i]=lujets_.p[i+4000];
        E[i]=lujets_.p[i+3000];
        Wh[i]=1.0;

        /* Check if fGasmode is set */
        fX = 0.;
        fY = 0.;
        fZ = 0.;
```

```
    if (fGasmode == 1) {

            // define position of track start
            // Random 2D point in a circle of radius r (simple beamprofile)
            radius = gRandom->Gaus(0,fRsigma);
        gRandom->Circle(fX, fY, radius);

        // calculate fZ according to some (probability) density function of the gas
            fZ=fDensityFunction->GetRandom();

        }

        // add track
        //printf("- I -: new particle at: %f, %f, %f ...\n", fX, fY, fZ);
        primGen->AddTrack(Id[i], Px[i], Py[i], Pz[i], fX, fY, fZ);
    }//loop
    selector= true;

  }//  /////////////////////////////////////////
  else{
    selector = false;
  }
 }//npart == 4
 if(selector){
  //cout<<"Yes I have 4 pi+"<<endl;
  break;
 }
 }//End of while
 return kTRUE; //////////////////////////////////////////


}
```

Hello all,

we ran now into the same problem as Donghee did more than a year ago (in the sense that we also wanted to filter the events which are produced by the dpm generator) while performing a background study. A solution was found pretty fast for our specific problem, but I believe that it would make sense to have a more general solution. Therefore, I implemented a filter for PndDpmDirect which can handle pretty general requests from the simulation macro in the

form:

```
Dpm->AddFilterMinMax( 1, 5, 11, -211 ); // request at least 1 and at most 5 e- OR pi-
Dpm->AddFilterMinMax( 1, 5, -11, 211 ); // AND request at least 1 and at most 5 e+ OR pi+
Dpm->AddFilterMinMax( 3, 9999, 22 ); // AND request at least 3 and at most 9999 gamma
```

The standard behaviour is the same as before (i.e. no event filtering).

You can tell PndDpmDirect how often it should try to find a suitable event before giving up:

```
Dpm->SetFilterMaxTries(99999);
```

and it can tell you at the end of the simulation macro how many events dpm simulated in total to get the number of filtered events that you wanted as well as how many events reached the limit of tries without success:

```
if( UseDpm ){
cout << Dpm->GetNumberOfSimulatedEvents() << " events were simulated in dpm\n";
cout << Dpm->GetNumberOfFilterFailedEvents() << " unsuccessful attempts to find an event
that suits your filters\n\n";
}
```

The code is currently still in testing and if you agree I would like to upload it to the trunk once it was tested to be reliable.

EDIT:
Here was a paragraph about a restriction of the code which is obsolete now.

The restriction that you can add a filter for a specific pdg code ONLY ONCE has been removed in versions later than 2013-10-21:

I am looking forward to reading your feedback.

Kind regards,
Martin

Subject: Re: Specific event selection in Dpm generator
Posted by StefanoSpataro on Wed, 28 Aug 2013 22:15:55 GMT
View Forum Message <> Reply to Message

Hi Martin,
I think it would be good to have such code not only for DPM, but also for all the generators.
In this sense, instead of having it as an extension of DpmDirect, it would be good to implement

the method inside FairPrimaryGenerator, if of course Mohammad and Florian agree. What do you think?

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Thu, 29 Aug 2013 09:10:16 GMT
View Forum Message <> Reply to Message

Hello Stefano,

sure, it would certainly be possible to integrate it in FairPrimaryGenerator if the experts agree.

I can either provide the code that I wrote for PndDpmDirect as an inspiration for the experts or I would need the advice of at least one of them to help me integrate it in FairPrimaryGenerator without breaking any functionality of PandaRoot or possibly other frameworks. Ideally, one could implement it so that no changes to all derived classes would be necessary.

EDIT:
After talking to Klaus Götzen and Andreas Pitka I implemented the possibility for an arbitrary number of conditions per pdg code and also a maximum number of particles per event.

Also a filter just on particles' charges would be desirable.
In addition, a minimum and maximum (total/transversal/z) momentum should be implemented (both in the laboratory system as well as in arbitrary center of mass systems).

Let's see what Mohammad and Florian think about it.

Kind regards,
Martin

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Thu, 29 Aug 2013 10:20:56 GMT
View Forum Message <> Reply to Message

I have just uploaded a first working and tested implementation to my development folder in the svn repository. You can find the files here:

 PndDpmDirectWithFilter

EDIT:
Note that I have not yet implemented all the functionality which is described in the EDIT regions.

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Fri, 18 Oct 2013 13:03:35 GMT
View Forum Message <> Reply to Message

I have uploaded an updated version of the code. Now one can ask for a minimum number and a maximum number. I have adjusted the example code snippets in this forum, too.

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Mon, 21 Oct 2013 14:34:51 GMT
View Forum Message <> Reply to Message

The restriction of only one condition per pdg code has been removed. The filter can handle multiple conditions for the same PDG code now. I have just uploaded a debugged, compiling and running version to my development branch (see above).
The momentum and geometry cuts are still missing. Other than that the filter is fairly complete.

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Mon, 21 Oct 2013 14:56:14 GMT
View Forum Message <> Reply to Message

You can find more information and a presentation concerning the filter here:

http://panda-wiki.gsi.de/cgi-bin/view/Computing/Minutes21October2013

Subject: Re: Specific event selection in Dpm generator
Posted by Anastasia Karavdina on Wed, 22 Jan 2014 17:41:54 GMT
View Forum Message <> Reply to Message

Hi Martin!
In your talk on the latest collaboration meeting, you mentioned as missing option filtering on kinematics (theta and phi angles). Are you working on it? Or maybe those are already implemented?

Cheers,
Anastasia.

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Wed, 22 Jan 2014 17:59:47 GMT
View Forum Message <> Reply to Message

Hello Anastasia,

the filter project is ongoing, a master student (as a side-project) and I (as a "hobby") are working part-time on it and that item is on the todo list. However, due to time pressure from the FTS TDR and a learning curve for the student the progress is not very fast. If you like to add such functionality in the meantime I am happy to supply the latest version of the code.

Kind regards,
Martin

Subject: Re: Specific event selection in Dpm generator
Posted by Anastasia Karavdina on Wed, 22 Jan 2014 18:05:34 GMT
View Forum Message <> Reply to Message

Thank you for prompt reply!
We also have have TDR to write and I have no idea how implement this feature. But I'll think about it.
Is version in your development brunch the latest one?

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Wed, 22 Jan 2014 18:08:41 GMT
View Forum Message <> Reply to Message

Hello Anastasia,

yes, it is the latest version.

If you have questions about the existing implementation I am happy to provide you with more information / add more comments to the source code.

Kind regards,
Martin

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Fri, 28 Feb 2014 16:35:44 GMT
View Forum Message <> Reply to Message

Katja Kleeberg and I have just ported the filter code to a class and modified FairPrimaryGenerator to use it. The code is still under development and has not been tested rigorously yet, but if someone would like to have the code, please let me know.

Subject: Re: Specific event selection in Dpm generator
Posted by MartinJGaluska on Sun, 09 Mar 2014 11:15:36 GMT
View Forum Message <> Reply to Message

I would recommend to close this topic. We have a new implementation of an event filtering framework independet of PndDpmDirect. I created a new topic and would suggest to continue the discussion there. Here is the link:
https://forum.gsi.de/index.php?t=msg&goto=15933&#msg_15933