Pluto: Weight-Based Event Sampling

I. Fröhlich[1]

[1]Institut für Kernphysik, Johann Wolfgang Goethe-Universität, 60486 Frankfurt, Germany.

1 Definitions

Sampling distribution/model: Determines the number of events per phase space bin by a physics model.

Generator distribution/model: The number of events is sampled by any (un-physical) distribution.

Weighting distribution/model: The physics model assigns only weights and does no sampling.

2 Motivation

It is very often the case, that experiments measure regions of the phase space, where the "physics" produces a rare number of events. Such an example is the electromagnetic Dalitz decays, measured by the HADES spectrometer, where the di-lepton yield spans over orders of magnitide, with a high differential cross section for the low-mass pairs, whereas the high-mass pairs have a much lower cross section $d\Gamma/dM$ (however the focus of the HADES program is on the latter).

On the other hand, Monte-Carlo simulations need an adeqate statistics for the region of interest. The usual event sampling, as it is also done in the Pluto event generator, takes such physics distributions and samples (lets say the di-lepton mass) according to the provided differential cross sections.

The consequence is that a high number of events have to be sampled before an acceptable number of events in the region of interest have been collected. This problem is even more drastic if the detector setup removes the sampled events preferrably in the region with high statistics. E.g., in the HADES di-lepton analysis with the cut on the opening angle of the pair removes most of the low-mass region which have been sampled before.

The solution for such a waste of computing performance is weight-based event sampling, where a "weight" is assigned to each of the particle (or the event). This means that each single Monte-Carlo event stands for a hidden number of real events if the weight is un-like one.

Such a weight-based event sampling is also neccessary for differential cross sections with a high number of free parameters (e.g., a 3-body decay with 5 degrees of freedom). Whereas a 1- or 2-dimensional sampling can be handeled numerically, random sampling over 5 dimensions is difficult (if not impossible). The approach taken in Pluto (v5.11) is to treat the Dalitz decays as a 2-step process, via an intermediate "dilepton" which decay into e^+e^-. But this is restricted to uncorrelated decay steps.

Connected to the weights is the question of the absolute normalization of the final spectrum, since weights can also be used to obtain the absolute scale, which should be independent from the chosen event generator, and the number of events.

3 Normalization: The simple case

The normalization can be obtained in a very simple way if the usual "sampling" algorithm (sampling in the sense that physical shape is directly sampled, as it is the default in Pluto) is used. In such a case the default weight W_{parent} of the overall parent particle is $1/N_{ev}$ with N_{ev} the number of events which are produced. The weight of the daughter particle(s) of the k-th decay step is $W_{parent} \cdot b^{\mathrm{k}}$, where b^{k} is the static branching ratio.

In addition, the weight of the parent particle can be merged with an enhancement factor. This is usefull for the thermal macros, where a source multiplicity can be taken into account:

```
PFireball *fb1=new PFireball("pi0",Ebeam,T1,T2,frac,blast,A2,0.,0.,0.);
fb1->setTrueThermal(trueT);
fb1->SetMultiplicity(Mpi0);
fb1->Print();
```

4 Example: The Dalitz decay

As pointed out, weights can be helpfull in the sampling of the Dalitz decays. Instead of sampling the di-leptons with the differential cross sections, sampling is done using any di-lepton distribution. In the analysis macro, the histogram has to be filled with the stored weight as a statistical factor and corrected for the number of chosen bins. The weight calculation is done by Pluto such that finally the correct distribution is shown. This means that 2 steps have to be done: First, the event sampling and then weighting by the physical model.

4.1 The event generator

In the first step the di-leptons are sampled using an additional "generator" distribution:

```
TF1 *flat=new TF1("flat","1",0,1);
PInclusiveModel *dilepton_generator = new
    PInclusiveModel("flat@eta_to_g_dilepton/generator","Dilepton generator",-1);
dilepton_generator->SetSampleFunction(flat);
dilepton_generator->EnableGenerator();

dilepton_generator->Add("eta,parent");
dilepton_generator->Add("g,daughter");
dilepton_generator->Add("dilepton,daughter,primary");


makeDistributionManager()->Add(dilepton_generator);
```

The syntax of these lines have to be understood as follows: In the first 3 lines, the TF1 object is created and used for the PInclusiveModel class, which samples the mass of one (called "primary") particle. The function is valid between 0 and 1 (total free energy).

The first string inside the constructor is interpreted by a parser: The word before the "@" is the unique identifier for the distribution manager. The last word after the "/" is the alias path which marks the model to be not a primary but a secondary model (here: it goes into a hidden "generator" path). The string in the middle contains the decay with the usual pid strings.

The 4th line is very importing since it set all flags correctly such that the PChannel recognizes the model as a generator.

The next 3 lines are the usual template.

Non-uniform generators can be created very simply by exchanging the TF1 object:

```
TF1 *nonflat=new TF1("nonflat","0.2+x*x",0,1);
dilepton_generator->SetSampleFunction(nonflat);
```

The consequence is that regions of the phase space are are arteficially enhanced, thus they have to be re-weighted by $1/W_{gen}$. Due to the fact that the total normalization must not be changed the requirement $\overline{1/W_{gen}} = 1$ should be fulfilled.

This could be done by the user, but to avoid mistakes it is more save that the Pluto framework checks this requirement and does a renormalization. The mean of the weight is monitored and re-scaled for each (new!) event with the factor: $S_{gen} = 1/\overline{1/W_{gen}}$.

Unless enough events have been sampled to have a mean value which is precise, the (first) events do not have the proper normalization. To avoid this problem, the loop can be invoked by dummy events "pre-heating" the normalization:

```
PReaction->Preheating(100);
```

4.2 The weighting model

After the di-leptons have been sampled by a generator as described above each di-lepton has to be weighted by the physical model W_{mod} in addition:

$W_{final} = W_{mod} \cdot S_{gen}/W_{gen}$

Here, the weighting feature of the physics model (which might be already implemented in the Pluto framework) has to be enabled:

```
makeDistributionManager()->GetDistribution("eta_dalitz")->EnableWeighting();
```

But also in this case the normalization has to be correct. As described in Sec. 3 the integral of the model weights have to be be equal to the branching ratio. In our more complicated case the mean of all weights should show the branching ratio as well, which leads to the additional factor:

$S_{mod} = b^{\mathrm{k}}/\overline{W_{mod}}$.

But this works only if the generator is flat! In the case that the generator enhances regions of the phase space (where the mean of the weighting models is completety different) also the overall mean is wrong.

Thus, a weighted mean calculation has to be done:

$\overline{W_{mod}} = \frac{\sum W^i_{mod} \cdot \frac{1}{W^i_{gen}}}{\sum \frac{1}{W^i_{gen}}}$

The result of the different models are shown in Fig. 1, using the same number of Monte-Carlo events. As it can be seen, the high-mass region is much more precise for the weighting cases.

5 Outlook

The methods desribed above are very preliminary. In order to make the total normalization correct, an interface to add the total cross section of the reaction has to be implemented. For the Δ Dalitz decay with higher order distributions (5 dimensions) a generator as well as a model spanning 2 decay steps have to be created.

The weighting has to be applied to the Deuteron wave function as well (sub-threshold eta production). In addition, different gerator models have to be identified by the distribution manager and handeled as an alternative.

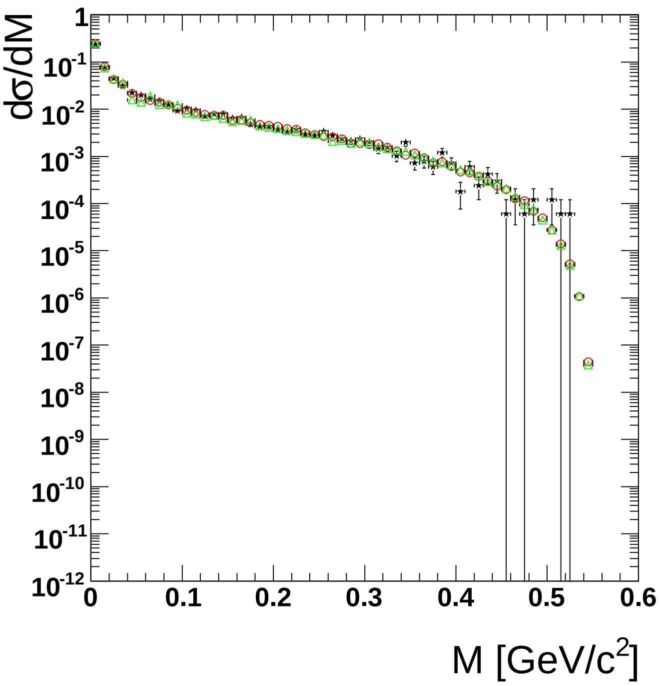

Figure 1: 10000 sampled η Dalitz events. Black stars: Sampling model. Red circles: Flat generator. Green triangles: $0.2 + x^2$ generator.