Forcing signal decays with EvtGen

Philip Ilten

Laboratory for Nuclear Science, Massachusetts Institute of Technology

Original date: 17 Sep 2015
Last changed: 17 Sep 2015

For many experimental collaborations, particularly those specializing in B-physics, more detailed hadron decay models are needed than those provided by default in Pythia 8. The EvtGen [1] package specializes in B-hadron decays, including sophisticated models, spin correlations, and the ability to implement new models. To include spin correlations EvtGen does not just decay a single particle at a time, but instead performs the entire decay tree for each given initial particle. Consequently, decays from EvtGen cannot be included in Pythia 8 via the provided `DecayHandler` class, called during the decay stage of the hadron level, but must rather be performed after full event generation. Such an interface for EvtGen is provided by the plugin class `EvtGenDecays`, with technical details provided in both the HTML manual and examples.

In B-physics, particularly at hadron colliders, one oftentimes wishes to produce a large sample of events where each event contains one or more rare signal decays, *e.g.* $\mathrm{B}^0_\mathrm{s} \to \mu^+\mu^-$. The first step, of course, is to generate an event with at least one signal particle candidate, while the second step is to force the signal decay for one of these candidates. The weight for an event containing one candidate with a forced signal decay is simply the branching fraction for the signal decay. However, when multiple candidates are present, the event weight becomes slightly more complex, requiring non-trivial bookkeeping. Consequently, the `EvtGenDecays` class in Pythia 8 provides a generalized mechanism by which to force signal decays for given particle species while still providing an appropriate event weight.

1 Event weighting

Signal particle candidates, c_i, do not all need to be the same particle species. Here, a particle species differentiates not only between particle types, *e.g.* B^0_s and τ^+, but also between particles and anti-particles, *e.g.* τ^+ and τ^-. Additionally, the signal decay for a candidate, with branching fraction $\mathcal{B}_\mathrm{sig}(c_i)$, can include multiple channels. Consequently, arbitrarily complex signal decays can be forced. As an example, events can be required to contain one or more of the following decays: $\tau^+ \to \bar{\nu}_\tau\pi^+$, $\tau^+ \to \bar{\nu}_\tau\pi^0\pi^+$, $\mathrm{B}^0_\mathrm{s} \to \mu^+\mu^-$, and $\tau^- \to \tau_\nu\pi^-\pi^-\pi^-\pi^+\pi^+$. Here, assuming equal production of the three particle species (which is almost certainly not the case), the decay $\tau^+ \to \bar{\nu}_\tau\pi^0\pi^+$ of the four signal decays will be the most commonly forced decay. Following this notation, the event weighting is performed as follows.

1. An event is generated and all n signal particle candidates, c_i, are found. If there are no candidates, $n = 0$, then an event weight, $\mathcal{W}_\mathrm{event}$, of 0 is returned.

2. If $n > 0$ then a candidate c_i is randomly chosen with probability

$$P(c_i) = \frac{\mathcal{B}_{\mathrm{sig}}(c_i)}{\sum_{j=1}^{m} \Big(1 - \mathcal{B}_{\mathrm{sig}}(c_j)\Big)} \tag{1}$$

   where $\mathcal{B}_{\mathrm{sig}}(c_i)$ is the signal branching fraction for each candidate c_i.
3. A channel is selected for the chosen candidate c_i from one of the signal channels contributing to $\mathcal{B}_{\mathrm{sig}}(c_i)$.
4. Channels for all remaining candidates are selected, using all allowed decay channels, not just the signal channels.
5. The number of candidates with a selected signal channel, m, is determined. The channel selection for the candidates is then kept with probability $1/m$. If the channel selection is rejected the algorithm returns to step 2 and a new set of channels is selected.
6. All candidates are decayed via their selected channel and

$$\mathcal{W}_{\mathrm{event}} = 1 - \prod_{i=1}^{n} \Big(1 - \mathcal{B}_{\mathrm{sig}}(c_i)\Big) \tag{2}$$

   is calculated as the event weight.

An unweighted sample of events can be obtained by randomly selecting events, each with probability $\mathcal{W}_{\mathrm{event}}/\mathcal{W}_{\mathrm{max}}$. The maximum possible event weight, $\mathcal{W}_{\mathrm{max}}$, can be determined by the maximum weight from a sufficiently large sample of events.

Acknowledgments

The event weighting logic is based on discussions with Torbjörn Sjöstrand. Input, especially from Ian Counts and Torben Ferber, is gratefully acknowledged. The author should solely bear the blame for opinions, simplifications and errors in this note, however.

References

[1] D. J. Lange, Nucl. Instrum. Meth. A **462** (2001) 152.