ChiGen : event generator for $\chi_{c1,2}$- and $X(3872)$- mesons production in PandaRoot

Stanislav Poslavsky

IHEP collaborators:

Sofia Bukreeva, Dmitry Morozov,
Alexey Luchinsky, Sergey Ruzhikov

Institute for High Energy Physics, Protvino, Russia

SRRC RF ITEP NRC Kurchatov Institute, Moscow Russia

8 September 2014

Brief overview

- Available processes:

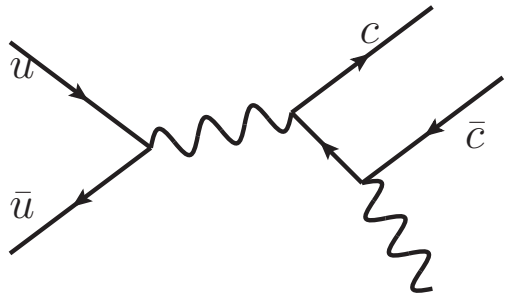
$$p + \bar{p} \to \mathcal{Q}(|\lambda|) + X \to J/\psi + \gamma + X \to e^+(\mu^+) + e^-(\mu^-) + \gamma + X$$

$$p + \bar{p} \to \mathcal{Q}(|\lambda|) \to J/\psi + \gamma \to e^+(\mu^+) + e^-(\mu^-) + \gamma,$$

  where $\mathcal{Q}$ is $\chi_{c1,2}$ or $X(3872)$ and λ — helicity in c.m. frame (or a sum over all helicities)
- `ChiGen` assumes that $X(3872)$ is 1^{++} pure $c\bar{c}$, i.e. it differs from χ_{c1} only by the mass
- Outline:
  - Theory overview
  - Implementation notes
  - `ChiGen` usage

Theory overview

- Theoretical background: *Luchinsky and Poslavsky, Phys.Rev. D85 (2012) 074016*
  - the dominated partonic subprocess: $u + \bar{u} \to \chi_c + g$
  - other subprocesses ($g + g$, $d + \bar{d}$, $g + u$ etc.) are suppressed due to small partonic distributions at large x
  - $c\bar{c}$ pair in P-wave mainly forms in color singlet combination *(Likhoded, Luchinsky and Poslavsky, Phys.Rev. D86 (2012) 074027)*
  - so, we can consider only one Feynman diagram



Inclusive channel: implementation features

Problems using Pythia8 :

- Pythia8 does not work at energies lower than 10 GeV (in PANDA $\sqrt{s} \sim 5.6$ GeV in production mode)
- Even if we adjust Pythia8 and make it works with low energies (by modifying BeamParameters.xml, PhaseSpaceCuts.xml, etc), it produces baryons in final state, which is crucial for charmonium production:

$$M_p + M_p + M_{\chi_c} \approx 5.5\,\text{GeV}$$

  - this is a consequence of Pythia8 underlying quark-diquark model of proton and theory tells that this should happen nearly in a half of events
- In terms of Pythia8 color flow it is impossible to describe $0_c \to 3_c 3_c 3_c$ and $\bar{3}_c \to 3_c 3_c$

$\Rightarrow$ We cannot use Pythia8 in the usual way

- So we need to implement generator "from scratch"
- We can use Pythia8 just for hadronization of color remnants

Inclusive channel: ChiGen workflow

ChiGen workflow:

1. MC simulation on partonic level

$$p + \bar{p} \to (uud) + (\bar{u}\bar{u}\bar{d}) \to \chi_c + g + (ud) + (\bar{u}\bar{d})$$

   - set kinematics and colors for all partons (neglect quark-diquark structure of proton)
   - the momentums of u and $\bar{u}$ are chosen using PDF's, while the momentums of remnants are randomly distributed
   - calculate distribution of $u + \bar{u} \to \chi_c + g$

2. Hadronizes color remnants using Pythia8 :

$$g + (ud) + (\bar{u}\bar{d}) \to X$$

3. Use EvtGen for radiative decays

$$\chi_{cJ} \to J/\psi + \gamma \to e^+ + e^- + \gamma$$

4. Pass all final particles to PandaRoot

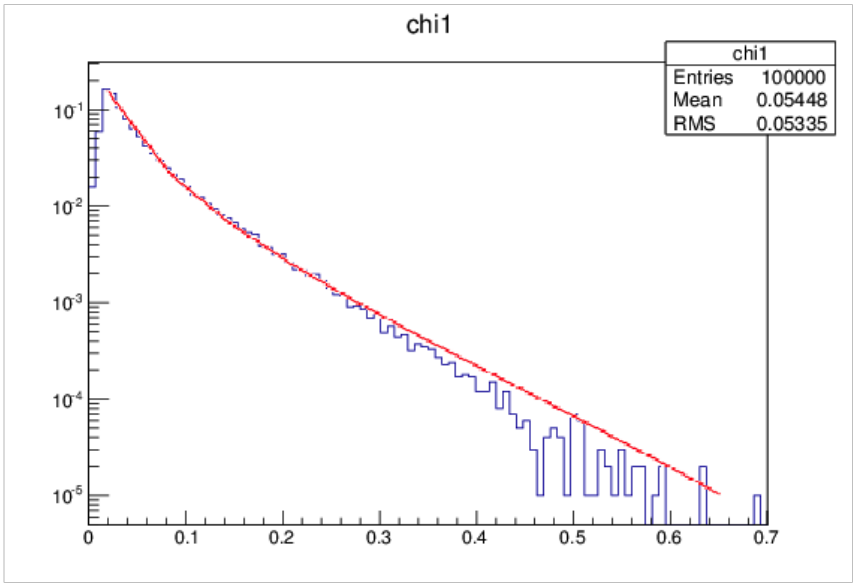
Inclusive channel: typical event structure

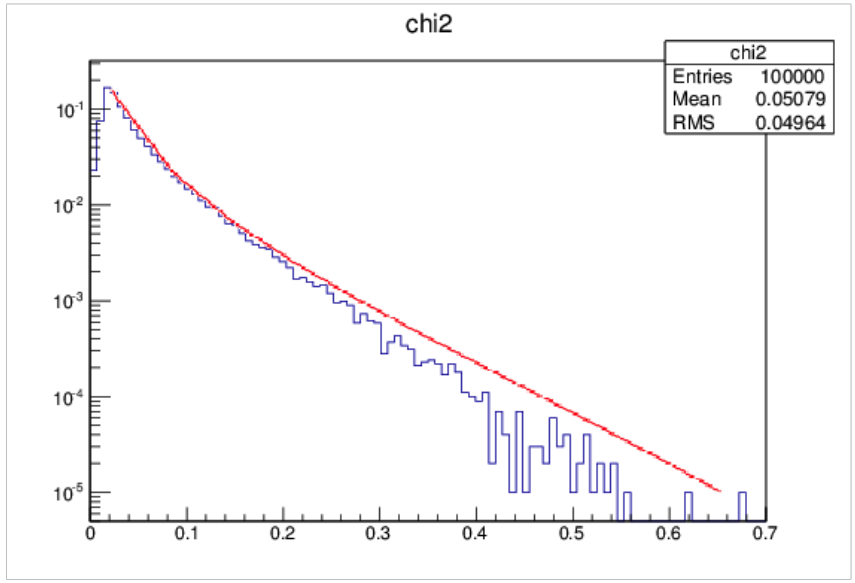
1. Abstract class `PartonicModel` implements MC of partonic subprocess ($u\bar{u} \to \chi_c + g$); it has the following implementations: `chi_1, x3872, chi_1_0, chi_1_1, chi_2, chi_2_0` etc.
2. Class `PythiaChiGen` is responsible for the hadronization
3. The decay of χ_c via `EvtGen` is performed also inside `PythiaChiGen` routines
4. Class `PndChiGen` wraps all the classes and pass produced particles to `PandaRoot`

	no	id	name	...	m
C++ parton level	0	90	(system)	...	5.500
	1	9998	(chi_c1)	...	3.510
	2	21	(g)	...	0.000
	3	2	(u)	...	0.000
	4	1	(d)	...	0.000
	5	-2	(ubar)	...	0.000
	6	-1	(dbar)	...	0.000
	.....				
Pythia8 hadronization	13	9998	(chi_c1)	...	3.510
	14	211	pi+	...	0.140
	15	-211	pi-	...	0.140
	.....				
EvtGen decay	22	443	J/psi	...	3.510
	23	22	gamma	...	3.510
	24	211	pi+	...	0.140
	25	-211	pi-	...	0.140

Inclusive channel: test generator kinematics etc.

- Comparison of simulated data and theory predictions (p_T-spectrum):





$\sqrt{s} = 5.5$ GeV

- The distributions over common kinematical variables (p_T, y, η etc.) are same as theory predicts
- Other obvious checks (energy/momentum conservation etc.) are also ok

Inclusive channel: radiative decays

- `VVP` and `TVP` models are used for χ_{c1} and χ_{c2} decays respectively
- An important bug in `TVP` model was fixed (in previous version there were incorrect numbers of photon and meson polarisations); Stefano Spataro updated `TVP.cpp` in `PandaRoot` svn repository
- For the subsequent decay of $J/\psi \to l\bar{l}$ a `VLL` model is used
- All these models take into account polarizations of particles

ChiGen usage

- One can use ChiGen as other PandaRoot generators

```
...
gSystem->Load("libEvtGen");
gSystem->Load("libEvtGenExternal");
gSystem->Load("libCHIGEN");
...
FairPrimaryGenerator* primGen = new FairPrimaryGenerator();
fRun->SetGenerator(primGen);
chigen::models::chi_1 chi1(5.5);
chigen::PndChiGen chigen(chi1);
primGen->AddGenerator(&chigen);
...
```

- The available models are chi_1, chi_2, x3872, chi_1_0, chi_1_1 etc.
- For exclusive production one can use:

```
...
chigen::models::chi_1 chi1;
chigen::PndChiGenExclusive chigen(chi1);
```

ChiGen usage: running in standalone mode

- One can run ChiGen in standalone regime:

```
gSystem->Load("libEvtGen");
gSystem->Load("libEvtGenExternal");
gSystem->Load("libCHIGEN");
chigen::models::chi_2 chi2(5.5);
chigen::ChiGenStandalone chigen(chi2, "file.root");
chigen.generateEvents(100000);
```

- file.root will contain only information about χ_c and J/ψ kinematical variables

ChiGen usage: options

Some options:

- Specifying custom decay models for $\chi_c \to J/\psi\gamma \to l^+ l^- \gamma$
- Specifying seed
- Printing debugging information

Full list of options is available at ChiGen wiki page:

https://panda-wiki.gsi.de/foswiki/bin/view/Computing/ChiGen

The source code is in the PandaRoot trunk

pgenerators/chigen