Deep_pi_pi_MC

0.1

Generated by Doxygen 1.9.1

1 Module Index

1.1 Modules

Here is a list of all modules:

2 File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

3 Module Documentation

3.1 Global event-by-event four-vectors

Functions

- TLorentzVector **Y4det** (0.0, 0.0, 1.0, 0.0)

Variables

- TLorentzVector **k4Beam0**
- TLorentzVector **P4Beam0**
- TLorentzVector k4Beam
- TLorentzVector **P4Beam**
- TLorentzVector **k4Scat**
- TLorentzVector **q4Virt**
- TLorentzVector **P4pipi**
- TLorentzVector **P4Scat**
- TLorentzVector **Delta4vec**
- TLorentzVector **P4piPlus**
- TLorentzVector **P4piMinus**
- TLorentzVector **P4Tot**
- TLorentzVector n4_e0
- TLorentzVector **n4Tilde_e0**
- TLorentzVector **X4_e0**
- TLorentzVector **Y4_e0**
- TLorentzVector **Z4_e0**
- TLorentzVector **T4_e0**
- TLorentzVector n4_e
- TLorentzVector **n4Tilde_e**
- TLorentzVector **n4_q**
- TLorentzVector **n4Tilde_q**
- TLorentzVector Y4_Det
- TLorentzVector **X4_e**
- TLorentzVector **Y4_e**
- TLorentzVector **Z4_e**
- TLorentzVector **T4_e**
- TLorentzVector X4_q
- TLorentzVector **Y4_q**
- TLorentzVector **Z4_q**
- TLorentzVector **T4_q**
- TLorentzVector **X4_pipi**
- TLorentzVector **Y4_pipi**
- TLorentzVector X4_qCM
- TLorentzVector **Y4_qCM**
- TLorentzVector **n4_qCM**
- TLorentzVector **n4Tilde_qCM**

3.1.1 Detailed Description

- $**$ Global (nominal) beam four-vectors $k_0^\mu, \quad P_0^\mu, \quad Y^\mu = [0,0,1,0] = \mathrm{VerticalUp}$

3.1.2 Variable Documentation

3.1.2.1 k4Beam `TLorentzVector k4Beam`

Event-by-event kinematic four-vectors

$$k^\mu,\, P^\mu,\, k'^\mu,\, q^\mu = (k = k')^\mu,\, P^\mu_{\pi\pi},\, P'^\mu, \Delta^\mu = (P' - P)^\mu, p^\mu_{\pi+}, p^\mu_{\pi-}$$

3.1.2.2 n4_e `TLorentzVector n4_e`

Event-by-event eP and qP light-cone four-vectors

$$n_e^\mu,\, \widetilde{n}_e^\mu,\, n_q^\mu \widetilde{n}_q^\mu$$

3.1.2.3 n4_e0 `TLorentzVector n4_e0`

Global (nominal) light-cone vectors

$$n_{e,0}^\mu,\, \widetilde{n}_{e,0}^\mu,\, X_{e,0}^\mu,\, Y_{e,0}^\mu,\, Z_{e,0}^\mu,\, T_{e,0}^\mu$$

3.1.2.4 X4_q `TLorentzVector X4_q`

cartesian four-vectors

$$X_q^\mu,\, Y_q^\mu,\, Z_q^\mu,\, T_q^\mu$$

3.1.2.5 X4_qCM `TLorentzVector X4_qCM`

Event-by-event, Boosted to $q + P$ CM frame

$$X_{q,CM}^\mu,\, Y_{q,CM}^\mu,\, n_{q,CM}^\mu,\, \widetilde{n}_{q,CM}^\mu$$

3.1.2.6 Y4_Det `TLorentzVector Y4_Det`

Event-by-event cartesian four-vectors

$$X_e^\mu, Y_e^\mu, Z_e^\mu, T_e^\mu$$

4 File Documentation

4.1 src/Deep_pi_pi_MC.cxx File Reference

$lN \to lN\pi\pi$ lepto-production cross sections (electron or muon) based on \`'Angular distributions in hard exclusive production of pion pairs'', B. Lehmann-Dronke, A. Schaefer, M. V. Polyakov, and K. Goeke, PHYSICAL REVIEW D, $\mathbf{63}$ (2001) 114001.

```
#include "Deep_pi_pi_MC.hpp"
#include "include/Deep_event.cxx"
#include <iostream>
```

Functions

- int Init (char ∗inFile)

  Read input file to initialize Monte-Carlo event generation.
- int Deep_pi_pi_MC ()

  Monte-Carlo Driver.

4.1.1 Detailed Description

$lN \to lN\pi\pi$ lepto-production cross sections (electron or muon) based on \`'Angular distributions in hard exclusive production of pion pairs'', B. Lehmann-Dronke, A. Schaefer, M. V. Polyakov, and K. Goeke, PHYSICAL REVIEW D, $\mathbf{63}$ (2001) 114001.

Input beam kinematics allow fixed target, head-on collisions, and crossing-angle collisions.

Created by Hyde, Charles E. on 5/15/2021.

Compile using make Makefile in bash shell (directory src/).

4.2 src/Deep_pi_pi_MC.hpp File Reference

File defines global variables (constants, and event-by-event variables).

```
#include <stdio.h>
#include <TCanvas.h>
#include <TH2D.h>
#include <TRandom3.h>
#include <TVector3.h>
#include <TLorentzVector.h>
#include <TH1D.h>
#include <TMath.h>
#include <TDatabasePDG.h>
```

Functions

- TLorentzVector **Y4det** (0.0, 0.0, 1.0, 0.0)
- bool **eSmear** (true)
- bool **iSmear** (true)

Variables

- const double **TwoPi** = 2.0∗TMath::Pi()
- auto **dbPDG** = TDatabasePDG::Instance()
- TRandom3 **ran3**
- TLorentzVector **k4Beam0**
- TLorentzVector **P4Beam0**
- TLorentzVector k4Beam
- TLorentzVector **P4Beam**
- TLorentzVector **k4Scat**
- TLorentzVector **q4Virt**
- TLorentzVector **P4pipi**
- TLorentzVector **P4Scat**
- TLorentzVector **Delta4vec**
- TLorentzVector **P4piPlus**
- TLorentzVector **P4piMinus**
- TLorentzVector **P4Tot**
- TLorentzVector n4_e0
- TLorentzVector **n4Tilde_e0**
- TLorentzVector **X4_e0**
- TLorentzVector **Y4_e0**
- TLorentzVector **Z4_e0**
- TLorentzVector **T4_e0**
- TLorentzVector n4_e
- TLorentzVector **n4Tilde_e**
- TLorentzVector **n4_q**
- TLorentzVector **n4Tilde_q**
- TLorentzVector Y4_Det
- TLorentzVector **X4_e**
- TLorentzVector **Y4_e**
- TLorentzVector **Z4_e**
- TLorentzVector **T4_e**
- TLorentzVector X4_q
- TLorentzVector **Y4_q**
- TLorentzVector **Z4_q**
- TLorentzVector **T4_q**
- TLorentzVector **X4_pipi**
- TLorentzVector **Y4_pipi**
- TLorentzVector X4_qCM
- TLorentzVector **Y4_qCM**
- TLorentzVector **n4_qCM**
- TLorentzVector **n4Tilde_qCM**
- double mLepton
- double **MIon**

- double **mPion**
- double **emitt_e** [3]
- double **emitt_i** [3]
- double **betaIP_e** [2]
- double **betaIP_i** [2]
- int **nEvents**
- double **Q2Min**
- double **Q2Max**
- double **yMin**
- double **yMax**
- double **MpipiMin**
- double **MpipiMax**
- double **csPiPiMin**
- double **csPiPiMax**
- double **W2Threshold**
- double **sqrtDL**
- double **deltaL**
- double **sqrtDQ**
- double **deltaQ**
- double **yInv**
- double **sMinusM2**
- double **xBj**
- double **psf**
- double **phi_e**
- double **s_e**
- double **W2**
- double **Q2**
- double **MpipiSq**
- double **k_dot_P**

4.2.1 Detailed Description

File defines global variables (constants, and event-by-event variables).

4.2.2 Variable Documentation

4.2.2.1 mLepton `double mLepton`

Invariants, definied in routine Init()

4.3 src/include/Deep_event.cxx File Reference

File generates a set of event four-vectors.

Functions

- int LeviCivita4vec (TLorentzVector vec1, TLorentzVector vec2, TLorentzVector vec3, double ∗vec4out)

    Initialize lightcone vectors n4_e, n4Tilde_e, X4_e, Y4_e.
- int LeviCivita4vec_old (double vec1[4], double vec2[4], double vec3[4], double ∗vec4out)
- double LeviCivitaScalar (TLorentzVector vec4_1, TLorentzVector vec4_2, TLorentzVector vec4_3, TLorentzVector vec4_4)
- int **EventLightCone** ()
- int Get_Event (int iEvt)

    Get_Event() generates ep->e p pi pi events uniformly in phase space.

4.3.1 Detailed Description

File generates a set of event four-vectors.

File contains event-by-event generation functions for $ep \to e'p'\pi\pi$.

4.3.2 Function Documentation

4.3.2.1 Get_Event()

```
int Get_Event (
    int iEvt )
```

Get_Event() generates ep->e p pi pi events uniformly in phase space.

$$\{Q^2, y = q\cdot P/(k\cdot P), \phi_e, M^2_{\pi\pi}, \cos(\theta^{CM}_{\pi\pi}), \phi^{CM}_{\pi\pi}, \cos\theta^{Rest}_{\pi+}, \phi^{Rest}_{\pi+}\}$$

Only basis 4-vectors e.g. $n_q, \widetilde{n}_q, X_q, Y_q$ are boosted, all other variables are invariants. After smearing the incident beam momenta, re-define the lepton-Ion light-cone vectors n4_e, n4Tilde_e, X4_e, Y4_e

Scattered lepton four-vector k4Scat = k'.

$$\left[\begin{array}{c} Q^2+2m_l^2 \\ 2(k\cdot P)(1-y) \end{array}\right] = 2\left[\begin{array}{cc} k\cdot\widetilde{n}_e, & k\cdot n_e \\ P\cdot\widetilde{n}_e, & P\cdot n_e \end{array}\right]\left[\begin{array}{c} k'\cdot n_e \\ k'\cdot\widetilde{n}_e \end{array}\right] \tag{1}$$

$$\left[\begin{array}{c} k'\cdot n_e \\ k'\cdot\widetilde{n}_e \end{array}\right] = \frac{1}{(k\cdot\widetilde{n}_e)(P\cdot n_e)-(k\cdot n_e)(P\cdot\widetilde{n}_e)}\left[\begin{array}{cc} P\cdot n_e, & -k\cdot n_e \\ -P\cdot\widetilde{n}_e, & k\cdot\widetilde{n}_e \end{array}\right]\left[\begin{array}{c} Q^2/2+m_l^2 \\ (k\cdot P)(1-y) \end{array}\right] \tag{2}$$

$$\mathbf{k}_\perp'^2 = 2(k'\cdot n_e)(k'\cdot\widetilde{n}_e) - m_l^2 \tag{3}$$

Define the lightcone vectors of q + P system

$$n_q^\mu = \frac{1}{\sqrt{2(1+\delta_Q)}}\left[q^\mu\frac{M_{Ion}}{q\cdot P} + P^\mu\frac{\delta_Q}{M_{Ion}\left(1+\sqrt{1+\delta_Q}\right)}\right], \qquad \delta_Q = \frac{Q^2 M_{\text{Ion}}^2}{(q\cdot P)^2} \tag{4}$$

$$\widetilde{n}_q^\mu = \frac{1}{\sqrt{2(1+\delta_Q)}}\left[P^\mu\frac{\left(1+\sqrt{1+\delta_Q}\right)}{M_{\text{Ion}}} - q^\mu\frac{M_{\text{Ion}}}{q\cdot P}\right] \tag{5}$$

Recoil proton kinematics:

- After boosting to the $q+P$ CM frame, X_q^μ, Y_q^μ are pure space-like (vanishing time components) and the space parts of n_q^μ, $\widetilde{n}_q^\mu$ are anti-colinear.
- In this frame, the incident proton is collinear with the unit three vector
- $\mathbf{v} = [\widetilde{n}_q.Px(), \widetilde{n}_q.Py(), \widetilde{n}_q.Pz()]/\,\widetilde{n}_q.E()$
  - The recoil nucleon and the $\pi\pi$ system are back-to-back in $q+P$ CM frame.
- Generate uniform distributions in
  - cosine of polar-angle: $\cos\left(\theta_{\pi\pi}^{CM}\right)$; and
  - azimuthal-angle $\phi_{\pi\pi}^{CM}$
-
$$\begin{bmatrix} E' \\ |\mathbf{P}'| \end{bmatrix}^{CM} = \begin{bmatrix} (W^2 + M_{\text{Ion}}^2 - M_{\pi\pi}^2)/(2\sqrt{W^2}) \\ \sqrt{[E'^{\text{CM}}]^2 - M_{\text{Ion}}^2} \end{bmatrix}$$

  Construct CM unit 3-vector in incident ion direction

$$\mathbf{v} = \left[\frac{\widetilde{n}_q.Px()}{\widetilde{n}_q.E()}, \frac{\widetilde{n}_q.Py()}{\widetilde{n}_q.E()}, \frac{\widetilde{n}_q.Pz()}{\widetilde{n}_q.E()}\right]^{CM}$$

  Define a longitudinal four-vector of the recoil proton in the CM frame:

$$\left[P_L'^{\mu}\right]^{CM} = \left[E', -|\mathbf{P}'|\cos\left(\theta_{\pi\pi}^{\text{CM}}\right)\mathbf{v}\right]^{\text{CM}}$$

  In any frame, the full recoil proton momentum four-vector is defined by:

$$P'^{\,\mu} = \left(P_L'^{\,CM}\cdot n_q^{CM}\right)\widetilde{n}_q^\mu + \left(P_L'^{\,CM}\cdot \widetilde{n}_q^{CM}\right)n_q^\mu - |\mathbf{P}'|^{\text{CM}}\sin\theta_{\pi\pi}^{CM}\left[\cos\phi_{\pi\pi}^{CM}X_q^\mu + \sin\phi_{\pi\pi}^{CM}Y_q^\mu\right]$$

  The momentum four-vector of the $\pi\pi$ system is simply

$$P_{\pi\pi}^\mu = q^\mu + P^\mu - P'^{\,\mu}$$

Similar method to construct four vectors of the final state pions

- Generate the $M_{\pi\pi} \to \pi\pi$ kinematics variables in the rest frame of the $M_{\pi\pi}$ system.
  - $\cos\theta_\pi^{\text{Rest}}, \phi_\pi^{\text{Rest}}$ are generated uniformly
  - for charge pion, the π^+ is the pion defined by θ_π^{Rest}
  - Is the phase space for a neutral pion 2π or 4π ?

4.3.2.2 LeviCivita4vec()

```
int LeviCivita4vec (
        TLorentzVector vec1,
        TLorentzVector vec2,
        TLorentzVector vec3,
        double * vec4out )
```

Initialize lightcone vectors n4_e, n4Tilde_e, X4_e, Y4_e.

$$n_e^\mu = \left[k^\mu\left(1+\sqrt{1-\delta_l}\right) - \frac{m_l^2}{k\cdot P}P^\mu\right] \Big/ \left[2\sqrt{(k\cdot I\right.$$
$$\widetilde{n}_e^\mu = \left[P^\mu - \frac{M_{\text{Ion}}^2}{(k\cdot P)\left(1+\sqrt{1-\delta_l}\right)}k^\mu\right] \Big/ \frac{}{\sqrt{(k\cdot I}} \tag{6}$$

$$n_e \cdot n_e = 0 = \widetilde{n}_e \cdot \widetilde{n}_e$$
$$n_e \cdot \widetilde{n}_e = 1$$

Transverse vectors

$$\texttt{Y4Det}^\mu = [0,0,1,0] = upinDetectorframe[X4_{e0}]_\sigma = \epsilon_{\mu\nu\rho\sigma} n4_e^\mu n4Tilde_e^\nu Y4_D et^\rho \tag{7}$$
$$X4_e^\mu = X4_{e0}^\mu / \sqrt{-X4_{e0}\cdot X4_{e0}} \tag{8}$$
$$[Y4_e]_\sigma = \epsilon_{\mu\nu\rho\sigma} n4_e^\mu X4_e^\nu n4Tilde_e^\rho \tag{9}$$
$$\tag{10}$$

If beam emmitance values in input file are positive, incident beam 4-vectors k^μ, P^μ are generated with gaussian longitudinal and transverse emmittance relative to nominal input values k_0^μ, P_0^μ.

The transverse 4-vectors X_e^μ, Y_e^μ are defined assuming neither incident beam can ever be in the vertical direction. Construct a 4-vector contraction

$$\texttt{vec4out}_{[0,1,2,3]} = \texttt{vec4out}_\mu = \epsilon_{\mu\nu\rho\sigma}\texttt{vec1}^\nu\texttt{vec2}^\rho\texttt{vec3}^\sigma$$
$$\epsilon_{0123} = 1 \tag{11}$$

4.3.2.3 LeviCivita4vec_old()

```
int LeviCivita4vec_old (
        double vec1[4],
        double vec2[4],
        double vec3[4],
        double * vec4out )
```

Construct a 4-vector contraction

$$*\texttt{vec4out}_\mu = \epsilon_{\mu\nu\rho\sigma}\texttt{vec1}^\nu\texttt{vec2}^\rho\texttt{vec3}^\sigma \tag{12}$$
$$\epsilon_{0123} = 1 \tag{13}$$
$$\tag{14}$$

4.3.2.4 LeviCivitaScalar()

```
double LeviCivitaScalar (
        TLorentzVector vec4_1,
        TLorentzVector vec4_2,
        TLorentzVector vec4_3,
        TLorentzVector vec4_4 )
```

Construct a the scaler anti-symmetric contraction of four space-time vectors

$$*\mathtt{Scalar} = \epsilon_{\mu\nu\rho\sigma}\mathtt{vec4_1}^{\mu}\mathtt{vec4_2}^{\nu}\mathtt{vec4_3}^{\rho}\mathtt{vec4_4}^{\sigma} \quad (15)$$

$$\epsilon_{0123} = 1. \qquad \text{In TLorentzVector Notation} \quad \epsilon_{xyzt} = -\epsilon_{0123} = -1 \quad (16)$$

$$(17)$$

Note from TLorentzVector documentation The components of TLorentzVector can also accessed by index:

```
xx = v(0);       or     xx = v[0];
yy = v(1);              yy = v[1];
zz = v(2);              zz = v[2];
tt = v(3);              tt = v[3];
```

Index