StringSpinner - adding spin to the PYTHIA string fragmentation

Albi Kerbizi\textsuperscript{a,}\textsuperscript{*}, Leif Lönnblad\textsuperscript{b}

\textsuperscript{a}Trieste Section of INFN, University of Trieste, Dept. of Physics, 34127 Trieste, Italy
\textsuperscript{b}Department of Astronomy and Theoretical Physics, Lund University, Sweden

Abstract

StringSpinner is a plugin for the introduction of the spin effects in the hadronization part of the PYTHIA 8 event generator. The spin effects are generated for pseudoscalar meson production by propagating the quark polarization along the fragmentation chain according to the rules of the quantum mechanical string+3\textit{P}$_0$ model. By using parametrizations for the transversity parton distribution functions, StringSpinner allows to simulate the polarized semi inclusive deep inelastic scattering process on protons and neutrons. The instructions to use StringSpinner and to modify the parametrizations of the transversity distribution functions are presented, as well as an example of the calculation of the Collins asymmetries as obtained from simulations of the transversely polarized semi inclusive deep inelastic scattering process.

Keywords: SIDIS, Spin, String+3P0, Transversity, Hadronization, Pythia;

PROGRAM SUMMARY

Program Title: StringSpinner
CPC Library link to program files:
Developer’s repository link: https://gitlab.com/albikerbizi/stringspinner.git
Code Ocean capsule:
Licensing provisions: GNU GPL v2 or later
Programming language: C++, Fortran
Nature of problem: To analyse and interpret the experimental data on the study of the nucleon spin structure, as well as to make predictions for future experiments

\textsuperscript{*}Corresponding author.
E-mail address: albi.kerbizi@ts.infn.it

Preprint submitted to Computer Physics Communications May 21, 2021
simulations are needed.

Solution method: Introduce spin effects in the hadronization part of a complete Monte Carlo event generator, largely used in the community of high energy particle physicists.

1. Introduction

The nucleon structure can be described in QCD by transverse momentum dependent parton distribution functions (PDFs) which encode information on correlations between the intrinsic transverse momenta of quarks, the spin of the quarks and the spin of the nucleon. In a fast moving nucleon, such functions depend on the lightcone fraction $x$ of the nucleon's momentum carried by the quarks, on the intrinsic transverse momentum of the quarks $k_{\perp}$ with respect to the nucleon's direction of motion and on the scale $Q^2$ at which the nucleon is probed.

To characterize the collinear ($k_{\perp}$-integrated) structure of the nucleon at leading twist only three PDFs are needed: the number density $f_q^1$ which describes the distribution of unpolarized quarks in an unpolarized nucleon, the helicity $g_q^1$ which describes quarks with longitudinal polarization in a longitudinally polarized nucleon, and the transversity $h_q^1$ which describes transversely polarized quarks in a transversely polarized nucleon. Among these three collinear PDFs, transversity is the less known and the most difficult to access due to its chiral odd nature.

The quark transverse polarization in nucleons is analyzed by using quark polarimeters. Different polarimeters have been proposed. Most of them use the conversion of quarks and gluons in observable hadrons described by means of fragmentation functions (FFs). One such polarimeter is the Collins effect [1], namely a left-right asymmetry in the azimuthal distribution of the hadrons produced in the fragmentation process of transversely polarized quarks with respect to the plane defined by the quark momentum and its polarization vector. It is characterized by the chiral odd Collins fragmentation function $H_{Tq}^{h}$. An other mechanism used to analyze the quark transverse polarization is the dihadron effect [2, 3, 4], described by the interference fragmentation function (IFF) $H_{Iq}^{h}$ which is thought to be originated from different physical processes compared to the Collins FF.

A powerful process to access transversity is the semi-inclusive deep inelastic scattering (SIDIS) process $lN \rightarrow l'hX$ where an incoming lepton $l$ scatters
off a target nucleon $N$ by exchanging a virtual photon and in the final state
the scattered lepton $l'$ and the hadron $h$ are observed in coincidence. In the
SIDIS cross section the Collins FF is coupled with the transversity PDF pro-
ducing a \( \sin \phi_C \) azimuthal modulation in the cross section. The azimuthal
angle \( \phi_C = \phi_h + \phi_S - \pi \) is the Collins angle of the produced hadron, \( \phi_h \)
being the azimuthal angle of the hadrons momentum transverse with respect
to the exchanged virtual photon momentum, and \( \phi_S \) the azimuthal angle of
the target nucleon transverse polarization \( S_\perp \) with respect to the exchanged
photon momentum. The amplitude of the \( \sin \phi_C \) modulation is the Collins asymmetry \( A_{Coll} \sim \sum_{q e} e_{q}^{2} h_{1}^{q} \otimes H_{1q}^{h} \) given by the convolution between the
transversity and the Collins functions over the involved transverse momenta.
Extractions of transversity and Collins functions have been performed by
combining measurements from Collins asymmetries in SIDIS \[5, 6\] with mea-
surements of the corresponding Collins asymmetries in \( e^+e^- \) annihilation to
hadrons processes \[7\]. Examples of such extractions can be found in Refs.
\[8, 9, 10\].

A useful tool to connect the theoretical and experimental studies on the
nucleon structure is a Monte Carlo event generator (MCEG). MCEGs provide
a full event description in all the allowed phase space and can be used to
guide the phenomenological interpretation of the experimental data or to
define new analyses to be performed in the data. One of the advantages of
MCEGs is the possibility to access the correlations among particles produced
in the same event, difficult to describe theoretically. At present, however,
a complete MCEG implementing the quark spin degree of freedom in the
hadronization part is still lacking.

In this article we present StringSpinner, a program module that can
be added to PYTHIA 8 \[11\] as a plugin for the simulation of the polarized
SIDIS process, with the aim to making the code available to the com-
unity. StringSpinner allows to emulate the recursive string+\(^3\)P\(_0\) model \[12\] of
polarized quark fragmentation in the hadronization part of PYTHIA. The
string+\(^3\)P\(_0\) model is based on the Lund Model \[13\] of string hadronization
already implemented in PYTHIA, with an additional assumption that the
\(q\bar{q}\) pairs at string breakups are produced with the vacuum quantum numbers
\(L = 0, S = 1\) and \(J = 0\). In addition to the phenomenological parame-
ters of the Lund Model, the \(^3\)P\(_0\) model introduces only one further complex
parameter responsible for the Collins and other spin effects.

The idea behind StringSpinner is to reweight the unpolarized hadroniza-
tion procedure in PYTHIA to reproduce the string+\(^3\)P\(_0\) model, where appli-
cable, to enable the study of the fragmentation of polarized quarks in realistic hadronic final states. Examples of such studies is to measure the Collins effect in transversely polarized quarks, and the effects in the SIDIS process of an unpolarized lepton off a polarized target. The spin effects are presently simulated for the production of pseudoscalar mesons without initial state radiation (ISR) and final state radiation (FSR). Parametrizations of the transversity distributions for the valence $u$ and $d$ quarks have been implemented and can be used to calculate the polarization vector of the struck quark. They are defined as C++ functions in a separate file and can be easily modified by the user. This package can therefore be used to simulate the Collins and the dihadron transverse spin asymmetries (TSAs) or, alternatively, the analysing powers (APs), namely spin asymmetries considering fully polarized quarks.

Some preliminary results obtained with StringSpinner were presented in Ref. [14]. More details on the string+$^3P_0$ model and on StringSpinner can be found in Ref. [15].

The article is organized as follows. Section 2 explains how the spin effects are introduced in the hadronization part of PYTHIA. Section 3 gives the practical instructions on how to utilize the plugin and to run a main program. Section 4 explains how to access different options for the simulations, as for instance how to change the parametrizations of the quark transversity distributions. Section 5 shows how to run the simulations and sec. 6 gives an example of the simulated Collins asymmetries. Finally we draw our conclusions.

2. Spin effects in PYTHIA for SIDIS

The plugin has been developed and tested with the version PYTHIA 8.2 of the event generator, but will soon also be available for PYTHIA 8.3. The spin effects in the hadronization part of PYTHIA are introduced by using the UserHooks class which is part of PYTHIA 8.2. This class enables a user to modify behaviours in the event generation at predefined points in the code. In particular one is allowed to accept or reject any hadron being produced in the fragmentation according to some logic, which in this case is that of the string+$^3P_0$ model described in the following. The output is the same as the standard PYTHIA output. No new information is added to the PYTHIA Event Record which reports directly the momenta of the final re-weighted hadrons.
When the SIDIS event generation starts, the Bjorken $x$ variable and the virtuality $Q^2$ of the exchanged virtual photon and the flavor of the initial quark $q_A$ are generated by PYTHIA as well as the hard scattering. Then a string between the scattered quark $q_A$ and the target remnant is set up for hadronization. Since ISR and FSR are switched off, this is a simple string stretched between $q_A$ and the target remnant, with no gluons in between.

2.1. Polarization vector of the fragmenting quark

To include the spin effects in the hadronization part of PYTHIA, as preliminary step the polarization vector $S_A$ of the scattered quark is calculated in the gamma nucleon reference system (GNS). The GNS system is defined by taking $\hat{Z} = \hat{q}$, $\hat{Y} = \hat{l} \times \hat{l}' / |\hat{l} \times \hat{l}'|$ and $\hat{X} = \hat{l}_\perp$. $q = \hat{l} - \hat{l}'$ is the exchanged virtual photon momentum, $\hat{l}$ and $\hat{l}'$ being the momenta of the incoming and scattered leptons respectively, and $\hat{l}_\perp$ is the transverse momentum of the incoming lepton with respect to $q$. The $\hat{Y}$ axis coincides with the normal to the lepton scattering plane. At leading order in $k_\perp/Q$ the GNS frame coincides with the string rest system, namely the reference system where the string is at rest and the string axis defines the $\hat{z}$ axis.

The vector $S_A$ can be either chosen as fixed, i.e. along some given direction, or calculated by using the parametrization of the corresponding transversity distribution $xh^{qA}_1(x, Q^2)$. In the latter case, the transverse polarization vector of the scattered quark with respect to $\hat{Z}$ is

$$S_{A\perp} = h^{qA}_1(x, Q^2) \times D_{NN}(y) \times \left[ S_{\perp} - 2(S_{\perp} \cdot \hat{l}_\perp)\hat{l}_\perp \right].$$

This equation takes into account the fact that quarks in a transversely polarized nucleon are partially polarized and that, after the hard scattering, the transverse polarization vector of the scattered quark is depolarized by the depolarization factor $D_{NN} = 2(1 - y)/(1 + (1 - y)^2)$, $y$ being the fraction of the virtual photon energy carried by the scattered lepton, and is reflected about the normal to the lepton scattering plane.

To evaluate Eq. (1) the following parameterizations for the valence $u$ and $d$ quark transversity PDFs

$$xh^u_1(x) = 3.2 x^{1.28} (1 - x)^4, \quad xh^d_1(x) = -4.6 x^{1.44} (1 - x)^4$$

are used. These parametrizations are obtained from a fit to the point by point extraction of transversity PDF performed in Ref. [8]. The $Q^2$ dependence
on $x h_1^{TA}$ is neglected. The dependence on the intrinsic transverse momentum of the struck quark is assumed to be the same as the distribution used by PYTHIA for the generation of the primordial transverse momentum. Concerning the unpolarized quark distribution $f_1^{qA}$, the same parameterizations set by the user in PYTHIA are used. As default we consider however the CTEQ5L set by using the option PDF:PS=2. This gives values for PDFs close to those used in Ref. [3].

The corresponding transversity distributions for the neutron are obtained automatically from the proton ones using isospin symmetry, namely $x h_1^{u/n} = x h_1^{d/p}$ and $x h_1^{u/p} = x h_1^{d/n}$. The other quarks are taken unpolarized.

Finally the polarization vector of the scattered quark in the string rest frame is taken $S_A = (S_A \perp, 0)$ and it is used to parametrize the $2 \times 2$ spin density matrix $\rho(q_A) = (1 + \sigma \cdot S_A)/2$, which provides the initial condition for simulating the polarized splittings.

2.2. Spin effects in PYTHIA hadronization

After the spin density matrix of the fragmenting quark is set up, PYTHIA starts the usual fragmentation process by choosing randomly if the first splitting (string break-up) has to be taken from the fragmenting quark side or from the target remnant side. Thus a splitting $q \to h + q'$ is simulated ($q$ is either $q_A$ or the target remnant) by generating a $q' \bar{q}'$ pair and the hadron $h$ is formed by using the standard PYTHIA procedure.

At this point the plugin inspects the produced hadron which is rejected if the splitting has been taken from the remnant side or if $h$ is not pseudoscalar. The former rejection forces the fragmentation chain to evolve from the $q_A$ side towards the remnant side to emulate the recursive string+$^3$P$_0$ model of Ref. [12], and this restriction is set as default in the plugin. Alternatively, if $h$ is not a pseudoscalar, it is rejected with a probability $1/2$ in order to not change the composition of hadrons in the event, but only the $k_T'$ distribution of the leading pseudoscalars.

If the hadron passes these selections it is accepted with probability

$$p(k'_T, S_q) = \frac{1}{2} \times \left[ 1 - \frac{2\text{Im}(\mu)}{|\mu|^2 + k'^2_T} S_q \cdot (\hat{z} \times k'_T) \right],$$

being always $0 < p < 1$. Otherwise the hadron is rejected and a new one is tried by PYTHIA until all selections are passed. The probability $p$ is inspired to the splitting function of the recursive string+$^3$P$_0$ model [12], $k'_T$. 

6
is the transverse momentum of $q'$ in the string rest frame and $\mu$ the complex mass parameter introduced by the $^3P_0$ mechanism to parametrize the spin effects and assumed to be equal for all quark flavors. The imaginary part of $\mu$ is responsible for the transverse spin effects in the fragmentation process.

The transverse momentum $k_T'$ is generated by PYTHIA following the standard recipe, namely according to the distribution $f_{\text{pyt}}(k_T'^2) = (1 - c) e^{-k_T'^2/\sigma_0^2} + c e^{-k_T'^2/\sigma_1^2}$. This is at variance with the string+$^3P_0$ model where $k_T'^2$ is generated according to the distribution $f_{^3P_0}(k_T'^2) \propto e^{-b_T k_T'^2 (|\mu|^2 + k_T'^2)/2}$.

However, since in PYTHIA it is $c \ll 1$, the distributions of the produced hadrons transverse momenta are the same as in the string+$^3P_0$ model for $b_T = \sigma_0^{-1/2}$ as shown in Ref. [14].

When the hadron is accepted the spin information of $q$ is transferred to $q'$ by calculating the spin density matrix $\rho(q')$ of $q'$ as in the recursive string+$^3P_0$ model [12]

$$\rho(q') = \frac{(\mu + \sigma_z \sigma \cdot k_T') \rho(q) (\mu + \sigma_z \sigma \cdot k_T')^\dagger}{\text{tr}[\text{numerator}]}$$

where $\sigma = (\sigma_x, \sigma_y, \sigma_z)$ is a vector of Pauli spin matrices.

This procedure is applied until the condition for the termination of the fragmentation chain is called by PYTHIA. This last step is treated by PYTHIA without taking into account spin effects.

3. The program files and instructions

The StringSpinner package can be downloaded from gitlab [3]. The package contains the following files

- StringSpinner.h
- mc3P0.f90
- Transversity.h

---

\[1\] The PYTHIA parameters corresponding to $\sigma_0^2$, $c$ and $\sigma_1^2$ are StringPT: sigma, StringPT: enhancedFraction and StringPT: enhancedWidth respectively. See also Tab. 1.

\[2\] The PYTHIA choice $f_{\text{pyt}}(k_T'^2)$ is equivalent to chose in the string+$^3P_0$ model the free input function $f_{^3P_0}(k_T'^2) = f_{^3P_0}(k_T'^2)/(|\mu|^2 + k_T'^2)^{1/2}$. In the limit for $c = 0$ it gives Eq. (27) in Ref. [12] with $\alpha = 0$.

\[3\] The repository can be found in https://gitlab.com/albikerbizi/stringspinner.git.
StringSpinner.h is a header file containing the definitions of classes and functions which constitute the core of the plugin. In particular, this file contains the implementation of the UserHooks class which allows to simulate the spin effects.

mc3P0.f90 contains a Fortran module for the calculation of the quark polarization related quantities, as for instance the $^3P_0$ inspired probability $p$.

Transversity.h is the header file containing the definitions of the quark transversity distributions and can be changed.

dis.cc is the main program providing a basic example of running the plugin for the simulation of the SIDIS process. It is a modified version of main36.cc shared among the examples of the PYTHIA 8.2 package.

Finally Makefile contains the instructions for the automatic compilation of the previous files and the link with the PYTHIA library.

3.1. The main program

3.1.1. Activation of spin effects

The main program dis.cc starts with

```cpp
#include "Pythia8/Pythia.h"
#include "StringSpinner.h"

using namespace Pythia8;

int main() {

Pythia pythia;
Event& event = pythia.event;
SimpleStringSpinner fhooks(pythia);
....
```

Note that StringSpinner.h must be included in the main. This allows to construct the fhooks class which is a SimpleStringSpinner class, the UserHooks base class for the implementation of the spin effects. fhooks must be given a Pythia object as input. The new methods introduced to deal with spin effects can be called as fhooks.method(). In the following the call of a particular method of fhooks will be referred to as method().
The first option which must be specified is if to simulate TSAs or the corresponding APs. TSAs are simulated if the target polarization vector is specified by using the method \texttt{setTargetPol()}. This method requires as input a \texttt{Vec4} object which spatial components specify the target polarization vector and in particular only the transverse part is considered in actual simulations. More precisely the line

\begin{verbatim}
    fhooks.setTargetPol( Vec4(0,1,0) );
\end{verbatim}

allows to simulate TSAs for a target which polarization vector is \((0,1,0)\), specified by its components along \(\hat{x}\), \(\hat{y}\) and \(\hat{z}\) axes in the laboratory rest frame.

The simulation of APs, instead, can be performed by using the method

\begin{verbatim}
    void setQuarkPol (int id,Vec4 Squark)
\end{verbatim}

which allows to specify the polarization vector \(\text{Squark}\) for the quark identified by \(\text{id}\). The integer \(\text{id}\) indicates the quark flavor and for this the usual flavor codes of PYTHIA are used, e.g. 1 for a \(d\) quark, 2 for a \(u\) quark, \(-2\) to a \(\bar{u}\) antiquark etc. The polarization vectors for the unspecified quarks are assumed to be zero. For instance the following line

\begin{verbatim}
    fhooks.setQuarkPol(2,Vec4(0,1,0));
    fhooks.setQuarkPol(1,Vec4(0,-1,0))
\end{verbatim}

initialises the polarization vector of \(u\) quarks along the \(\hat{y}\) axis and that of \(d\) quarks along \(-\hat{y}\) axis in the laboratory frame. For APs the scattered quark is not depolarized after the hard scattering, i.e. the depolarization factor \(D_{\text{NN}}\) is not taken into account.

The complex mass parameter introduced by the \(^3P_0\) mechanism can be set up by using the method \texttt{setMu(reMu,imMu)}, where the \texttt{double} variables \texttt{reMu} and \texttt{imMu} indicate the real and the imaginary parts of \(\mu\) respectively. This must be done before starting the loop over events. The values of the free parameters and other settings of PYTHIA can be modified by the standard procedure of the generator.

4. Further options

4.1. More general final states

As already mentioned, the spin effects are presently restricted to pseudoscalar meson production and other hadron types are by default simply
rejected. The production of more general final states can be activated with the method `setHadronMode(mode)`. `mode` is a `Mode` variable set by default to `mode=fhooks.reject`, meaning that all hadron types but pseudoscalar mesons are rejected. Selecting `mode=fhooks.disable` vector mesons and baryons are accepted with probability 1/2 as long as spin effects are active, but then spin effects are switched off as soon as a vector meson or baryon is produced. The fragmentation chain then continues as in standard PYTHIA.

This possibility provides more general final states from the point of view of particle content but spin effects are treated in an approximate way. The simulated TSAs could be diluted more, or less, than how much they would really be if all hadron types were correctly treated with spin effects. Moreover, vector mesons and baryons inherit a partial spin effect from the previous splitting where a pseudoscalar meson was produced, but such effect may not be the correct one expected from a true quantum mechanical treatment of such hadrons. Nevertheless this option could turn out be useful in some cases.

4.2. Splittings from both sides of the string ends

The same options are available also for the method `setRemnantMode(mode)`. By setting `mode` to `fhooks.reject`, PYTHIA rejects a hadron if it is produced after a string breakup carried from the remnant side. If `mode` is set to `fhooks.disable` then the hadrons produced from the remnant side are accepted and treated as in ordinary PYTHIA, i.e. without spin effects. In this case spin effects are applied only if the hadron is produced from the struck quark side. This option is more faithful to the PYTHIA procedure of taking splittings with equal probability from both the struck quark and the target remnant sides. Among the hadrons produced from the remnant side only pseudoscalar mesons are accepted if `setHadronMode` is set to `reject` whereas all hadron types are accepted if it is set to `disable`.

4.3. Modification of the transversity PDFs

The parametrizations of the transversity PDFs for the valence $u^v$ and $d^v$ quarks in a proton target are implemented in `Transversity.h` by the functions `uTransvPDF(x,Q2)` and `dTransvPDF(x,Q2)` respectively. These functions are given in input the values of the variables $x$ and $Q^2$ of the current event, and provide as output the values $xh_1^{u^v/p}$ and $xh_1^{d^v/p}$ obtained with the parametrizations given in Eq. (2). They can however be modified by the user as simple C++ functions.
4.4. Tuning of the parameters

The free parameters of the string+$^3P_0$ model are summarized in Tab. 1. In the first column are reported the parameters in the PYTHIA language. The correspondence with the parameters of the string+$^3P_0$ model is shown in the second column (see Ref. [12]). The values of the free parameters are given in the third and last columns, indicated as "$^3P_0$ tuning" and "PYTHIA tuning". The first three parameters of the $^3P_0$ tuning are the same as in Ref. [12] whereas the complex mass has been obtained by a chi-square minimization using the Collins asymmetries for pions as measured by COMPASS in SIDIS off protons [5] with the corresponding asymmetry simulated with StringSpinner assuming vanishing primordial transverse momentum.

To reproduce the COMPASS Collins asymmetries the complex mass $\mu$ has been changed from $(0.42 + i 0.76)\text{ GeV/c}^2$ quoted in Ref. [12] to $(0.78 + i 0.38)\text{ GeV/c}^2$. The same value for $\mu$ gives also a satisfactory comparison with the HERMES data [6]. Also, the same value of the complex mass can be used with the PYTHIA tuning as well, the change in the Collins asymmetries being small.

| PYTHIA            | string+$^3P_0$       | $^3P_0$ tum. | PYTHIA tum. |
|-------------------|----------------------|-------------|-------------|
| StringZ:aLund     | $a$                  | 0.9         | 0.3         |
| StringZ:bLund     | $b_L/(\text{GeV/c}^2)$ |              |             |
| StringZ:cLund     | $b_L/(\text{GeV/c})$ | 0.5         | 0.8         |
| StringPT:sigma    | $b_{\perp}^{1/2}/(\text{GeV/c})$ | 0.34        | 0.104       |
| StringPT:enhancedFraction | |             |             |
| StringPT:enhancedWidth | |             |             |
| reMu              | Re$\mu/(\text{GeV/c}^2)$ | 0.78        | 0.78        |
| imMu              | Im$\mu/(\text{GeV/c}^2)$ | 0.38        | 0.38        |

Table 1: Correspondence between the parameters of the fragmentation process in PYTHIA and in the string+$^3P_0$ model (for the meaning of the parameters see Ref. [12]), and the corresponding values for the two tunings.

4.5. Accessing event information

Methods to read spin dependent variables from the current event, as for instance the value of $xh_{1q}^{T_A}$ and the involved polarization vectors are available from the SimpleStringSpinner class. Such methods must be used inside the event loop. In particular, the value of the transversity PDF $xh_{1q}^{T_A}$ used in the current event can be accessed using the method `fhooks.xh1q()`. The method
**fhooks.STargetGNS()** provides a *Vec4* variable with the target polarization vector in the GNS frame. The polarization vectors of the quark before the hard scattering and after the hard scattering (fragmenting quark) in the GNS reference frame can be read off with the methods *SQuarkGNS()* and *SFragQuarkGNS()* respectively, which also provide a *Vec4* variable as output.

Information on the kinematic quantities of the current event can be accessed by using the *DISKinematics(Vec4 lin, Vec4 lout, Vec4 hin)* class, which requires in input the four momenta of the incoming lepton, of the scattered lepton and of the target hadron respectively. This class contains as member variables the DIS invariants $Q^2$, $W^2$, $x_B$, $y$, the matrix *GNS* which brings from the current frame to the GNS frame and the matrix *HCM* which brings from the current frame to the rest frame of the exchanged virtual photon and the target hadron. Both *GNS* and *HCM* are *RotBstMatrix* variables.

Finally, information on spin independent quantities such as the flavor of the struck quark, whether it comes from the valence or the sea and the corresponding value of the unpolarized PDF can be accessed via the standard PYTHIA methods.

### 5. Execution

To execute the **StringSpinner** main program the **Makefile** has to be set up by specifying the path to the PYTHIA installation directory in the variable **PYTHIADIR**. Then calling **make dis** produces the executable **dis** which is now ready to run with */dis*. The **make** command produces in addition the files **routines.mod, mc3P0.o** which are created as a consequence of the compilation of the Fortran module **mc3P0.f90**.

### 6. Example simulation

In Fig. 1 we provide an example of the calculation of the Collins asymmetry for positive pions (circles) and negative pions (triangles) as function of $x$. The asymmetry has been calculated in simulations of $8 \times 10^6$ SIDIS events in the COMPASS kinematics [5], namely scattering 160 GeV/c muons off a fully transversely polarized proton target at rest. We have used the $^3P_0$ tuning in Tab. 1 and in addition the primordial transverse momentum and the
Concerning the analysis of the simulated data, we applied the same binning and cuts as in the real COMPASS analysis [5]. In particular, the cuts \(0.1 < y < 0.9\), \(W > 5\) GeV/c\(^2\), \(0.003 < x < 0.7\), \(z_h > 0.2\) and \(P_{hT} > 0.1\) GeV/c have been applied, \(z_h\) and \(P_{hT}\) being the fraction of the exchanged virtual photon energy taken by the observed hadron and the hadron transverse momentum in GNS respectively.

![Collins asymmetry](image.png)

**Figure 1:** Collins asymmetry as function of \(x\) for positive pions (circles) and negative pions (triangles) calculated with Pythia+3P0 in the COMPASS kinematics [5].

### Conclusions and outlook

**StringSpinner** is a plugin to the PYTHIA 8 event generator which allows, for the first time, to simulate the polarized SIDIS process by using the rules of the string+\(^3\)P\(_0\) model for the propagation of the quark spin information in the hadronization part of the generator. Parameterizations of transversity PDFs have been introduced for the calculation of the fragmenting quark polarization and can be changed by the user. The spin effects are restricted to the production of pseudoscalar mesons and to string topologies without gluons having switched off ISR and FSR.

---

\(^4\)These settings are set as default in the dis.cc file.
**StringSpinner** can be used for the calculation of the Collins and di-hadron transverse spin asymmetries and analysing powers as well. This plugin is useful for the interpretation of the experimental data, to perform multidimensional studies and to make predictions for the future experiments like EIC.

Future developments includes the transition to version 8.3 of PYTHIA, but also to allow for spin dependence in the production of vector mesons, and possibly also baryons. The end-goal is to have a general plug-in to PYTHIA to handle spin-dependent hadronization in any collision system.

**Competing interests**

The authors declare that they have no common competing interests that may have influenced the work described in this article.

**Acknowledgement**

AK thanks Prof. Anna Martin, Prof. Xavier Artru, Prof. Franco Bradamante, Prof. John Collins and Prof. Torbjörn Sjöstrand for many fruitful and interesting discussions. He acknowledges also interesting discussions with physicists of the Lund group. The authors thank Dr. Markus Diefenthaler, Prof. Stefan Prestel and the LDTMDP group members for fruitful discussions. Part of the work was supported by the Jefferson Lab LDRD project LDTMDP.

**References**

[1] J. Collins, Fragmentation of transversely polarized quarks probed in transverse momentum distributions, Nuclear Physics B 396 (1) (1993) 161 – 182. [doi:https://doi.org/10.1016/0550-3213(93)90262-N](https://doi.org/10.1016/0550-3213(93)90262-N).

[2] J. C. Collins, S. F. Heppelmann, G. A. Ladinsky, Measuring transversity densities in singly polarized hadron hadron and lepton - hadron collisions, Nucl. Phys. B420 (1994) 565–582. [arXiv:hep-ph/9305309](https://arxiv.org/abs/hep-ph/9305309), [doi:10.1016/0550-3213(94)90078-7](https://doi.org/10.1016/0550-3213(94)90078-7).

[3] R. L. Jaffe, X.-m. Jin, J. Tang, Interference fragmentation functions and the nucleon’s transversity, Phys. Rev. Lett. 80 (1998) 1166–1169. [arXiv:hep-ph/9709322](https://arxiv.org/abs/hep-ph/9709322), [doi:10.1103/PhysRevLett.80.1166](https://doi.org/10.1103/PhysRevLett.80.1166).
[4] A. Bianconi, S. Boffi, R. Jakob, M. Radici, Two hadron interference fragmentation functions. Part 1. General framework, Phys. Rev. D62 (2000) 034008. [arXiv:hep-ph/9907475](https://arxiv.org/abs/hep-ph/9907475), doi:10.1103/PhysRevD.62.034008.

[5] C. Adolph, et al., Collins and Sivers asymmetries in muon-production of pions and kaons off transversely polarised protons, Phys. Lett. B744 (2015) 250–259. [arXiv:1408.4405](https://arxiv.org/abs/1408.4405), doi:10.1016/j.physletb.2015.03.056.

[6] A. Airapetian, et al., Effects of transversity in deep-inelastic scattering by polarized protons, Phys. Lett. B 693 (2010) 11–16. [arXiv:1006.4221](https://arxiv.org/abs/1006.4221), doi:10.1016/j.physletb.2010.08.012.

[7] A. Vossen, et al., Observation of transverse polarization asymmetries of charged pion pairs in $e^+e^-$ annihilation near $\sqrt{s} = 10.58$ GeV, Phys. Rev. Lett. 107 (2011) 072004. [arXiv:1104.2425](https://arxiv.org/abs/1104.2425), doi:10.1103/PhysRevLett.107.072004.

[8] A. Martin, F. Bradamante, V. Barone, Extracting the transversity distributions from single-hadron and dihadron production, Phys. Rev. D91 (1) (2015) 014034. [arXiv:1412.5946](https://arxiv.org/abs/1412.5946), doi:10.1103/PhysRevD.91.014034.

[9] M. Anselmino, M. Boglione, U. D’Alesio, J. O. Gonzalez Hernandez, S. Melis, F. Murgia, A. Prokudin, Collins functions for pions from SIDIS and new $e^+e^-$ data: a first glance at their transverse momentum dependence, Phys. Rev. D 92 (11) (2015) 114023. [arXiv:1510.05389](https://arxiv.org/abs/1510.05389), doi:10.1103/PhysRevD.92.114023.

[10] M. Radici, A. Bacchetta, First Extraction of Transversity from a Global Analysis of Electron-Proton and Proton-Proton Data, Phys. Rev. Lett. 120 (19) (2018) 192001. [arXiv:1802.05212](https://arxiv.org/abs/1802.05212), doi:10.1103/PhysRevLett.120.192001.

[11] T. Sjostrand, S. Mrenna, P. Z. Skands, A Brief Introduction to PYTHIA 8.1, JHEP05 (2006) 026, Comput. Phys. Commun. 178 (2008) 852–867. [arXiv:0710.3820](https://arxiv.org/abs/0710.3820), doi:10.1016/j.cpc.2008.01.036.

[12] A. Kerbizi, X. Artru, Z. Belghobsi, A. Martin, Simplified recursive $^3P_0$ model for the fragmentation of polarized quarks,
[13] B. Andersson, G. Gustafson, G. Ingelman, T. Sjostrand, Parton Fragmentation and String Dynamics, Phys. Rept. 97 (1983) 31–145. doi:10.1016/0370-1573(83)90080-7

[14] A. Kerbizi, L. Lønnblad, Inclusion of the $^3P_0$ model in PYTHIA 8, PoS DIS2019 (2019) 179. arXiv:1909.02280 doi:10.22323/1.352.0179

[15] A. Kerbizi, Recursive fragmentation of a polarized quark, Phd thesis (March, 2020). arXiv:2004.00524