

SIDIS-RC EvGen: a Monte-Carlo event generator of semi-inclusive deep inelastic scattering with the lowest-order QED radiative corrections

Duane Byer^{a,b}, Vladimir Khachatryan^{a,b}, Haiyan Gao^{a,b}, Igor Akushevich^a,
Alexander Ilyichev^{c,d}, Chao Peng^e, Alexei Prokudin^{f,g}, Stan Srednyak^{a,b},
Zhiwen Zhao^{a,b}

^a*Department of Physics, Duke University, Durham, NC 27708, USA*

^b*Triangle Universities Nuclear Laboratory, Durham, NC 27708, USA*

^c*Belarusian State University, Minsk, 220030, Belarus*

^d*Institute for Nuclear Problems, Belarusian State University, Minsk, 220006, Belarus*

^e*Physics Division, Argonne National Laboratory, Lemont, IL 60439, USA*

^f*Division of Science, Penn State Berks, Reading, PA 19610, USA*

^g*Theory Center, Thomas Jefferson National Accelerator Facility, Newport News, VA
23606, USA*

Abstract

SIDIS-RC EvGen is a C++ standalone Monte-Carlo event generator for studies of semi-inclusive deep inelastic scattering (SIDIS) processes at medium to high lepton beam energies. In particular, the generator contains binary and library components for generating SIDIS events and calculating cross sections for unpolarized or longitudinally polarized beam and unpolarized, longitudinally or transversely polarized target. The structure of the generator incorporates transverse momentum-dependent parton distribution and fragmentation functions, whereby we obtain multi-dimensional binned simulation results, which will facilitate the extraction of important information about the three-dimensional nucleon structure from SIDIS measurements. In order to build this software, we have used recent elaborate QED calculations of the lowest-order radiative effects, applied to the leading order Born cross section in SIDIS. In this paper, we provide details on the theoretical formalism as well as the construction and operation of SIDIS-RC EvGen, e.g.,

*Corresponding author.

E-mail address: vladimir.khachatryan@duke.edu (V. Khachatryan)

how we handle the event generation process and perform multi-dimensional integration. We also provide example programs, flowcharts, and numerical results on azimuthal transverse single-spin asymmetries.

Keywords: Monte-Carlo event generators for radiative events; FOAM Monte-Carlo event generator; ROOT, GSL, VEGAS, Cubature packages; Semi-inclusive deep inelastic scattering; Transverse momentum-dependent distribution, fragmentation functions; QED radiative corrections.

PROGRAM SUMMARY

Program Title: SIDIS-RC EvGen

Licensing provisions: GNU General Public License Version 3 (GPLv3)

Developer's repository link: <https://github.com/duanebyer/sidis>

Maintainers: Duane Byer; Department of Physics, Duke University

Distribution format: GitHub repository

Programming language: C++, Python

Operating System: Linux, macOS and their distributions

External packages: FOAM, ROOT, GSL, VEGAS, Cubature, Cog, WW-SIDIS, MSTWPDF

Nature of problem: The task is to first create a code for calculations of the leading order Born cross section as well as radiative corrections (RCs) at the next-to-leading order (NLO) of the cross section of lepton-hadron semi-inclusive deep inelastic scattering (SIDIS) at medium to high beam energies with incident unpolarized or longitudinally polarized lepton beam and unpolarized, longitudinally or transversely polarized target, enabling to compute azimuthal single-target and double-beam-target spin asymmetries. Afterwards, a Monte-Carlo event generator based upon this code is developed, where in the coding and simulation processes multi-dimensional integrals need to be calculated precisely to obtain the exact NLO RCs to the SIDIS cross section with high precision beyond ultra-relativistic limit, which means that the lepton mass is taken into account.

Solution method: The project for building the SIDIS-RC EvGen software is divided into a library component called `libsidis`, for the purpose of calculating the inelastic tail to the SIDIS total cross section, and a binary component called `sidisgen`, for generating events. The Monte-Carlo event generation is performed through the usage of the FOAM library, by applying a spatial partitioning method with hypercubical “foam of cells”. The multi-dimensional numerical integration is carried out by the GSL, VEGAS, and Cubature packages. The SIDIS structure functions that describe the SIDIS scattering process, and which are given in terms of transverse momentum-dependent parton distribution and fragmentation functions, are calculated in Gaussian and Wandzura–Wilczek-type approximations and implemented

in the generator accordingly.

Additional comments including restrictions and unusual features: The restrictions depend on the complexity of problems, limited by CPU time. As a consequence, the program running time is of order of several minutes to hours. SIDIS-RC EvGen does not comprise yet the contribution coming from the radiative tail of exclusive lepton-hadron reactions, which is a separate contribution involving exclusive structure functions that are currently not well known.

Contents

1	Introduction and physics motivation	4
2	Some basics of the theoretical framework for studying 3D momentum structure of the nucleon with SIDIS	8
2.1	SIDIS process and functions describing the nucleon structure	8
2.1.1	Kinematics and cross section of the SIDIS process . . .	8
2.1.2	TMD PDFs, TMD FFs, and SIDIS structure functions	12
2.2	Lowest-order QED radiative effects in SIDIS	15
2.2.1	Leading order cross section	15
2.2.2	Lowest-order radiative corrections to the SIDIS cross section	18
3	MC event generator SIDIS-RC EvGen for studying SIDIS processes with lowest-order radiative effects	23
3.1	Dividing the SIDIS inelastic cross section into soft and hard parts	23
3.2	Library component	26
3.2.1	Flexible mode	27
3.2.2	Streamlined mode	29
3.2.3	Structure function implementation	29
3.2.4	Leptonic/Hadronic coefficients	31
3.3	Binary (generator) component	33
3.3.1	MC generation with the FOAM library	34
3.3.2	Improvement of the foam efficiency	35

4	Numerical results obtained from SIDIS-RC EvGen	36
4.1	Charged-hadron Collins and Sivers azimuthal transverse single-spin asymmetries	37
4.2	Prospects on charged-hadron multiplicity and asymmetry data comparisons	43
5	Summary and outlook	44

1. Introduction and physics motivation

A considerable amount of our knowledge on the nucleon’s quark-gluon composition has been obtained via experimental and theoretical studies of the collinear parton distribution functions (PDFs) [1] and fragmentation functions (FFs) [2]. The precise knowledge of PDFs is essential for understanding of Quantum Chromodynamics (QCD), and for making quantitative predictions for the observables related to the strong interaction sector of the Standard Model. In turn, the search of beyond Standard Model physics also requires precise understanding of QCD. Within the so-called collinear factorization of deep inelastic scattering (DIS), leading-twist (one-dimensional) PDFs can be interpreted at the leading order as probability densities for finding an unpolarized/longitudinally polarized parton in a fast moving unpolarized/longitudinally polarized nucleon¹. They depend on a fraction x_{B_j} of the nucleon momentum carried by the parton and on the “resolution scale” that is associated with the hard scale Q^2 of a lepton-nucleon scattering process. These PDFs are well-known, and over the course of more than two decades the frontier of exploration has been extended to the three-dimensional (3D) nucleon structure [3].

The 3D nucleon structure in the momentum space is encoded in the so-called transverse momentum dependent distribution and fragmentation functions, TMD PDFs and TMD FFs [2, 4–10]. These partonic functions are generalizations of the aforementioned collinear distribution and fragmentation functions appearing in the standard collinear factorization [11]. Both of them depend on two independent variables: x_{B_j} and k_\perp in the case of TMD PDFs, as well as z_h and p_\perp in the case of FFs. In the latter case, z_h is the fraction of the quark momentum transferred to the produced (final-state) hadron, and

¹Collinear FFs in turn describe how quarks and gluons transform into color-neutral particles such as hadrons.

p_{\perp} is the transverse momentum of the same final-state hadron but with respect to the direction of the fragmenting quark. QCD factorization theorems have been proven for processes with two distinct measured scales, $Q_1 \ll Q_2$, such as the semi-inclusive deep inelastic scattering (SIDIS) [6, 10], Drell-Yan process [12], and production of two hadrons in e^+e^- annihilation [2]. In case of SIDIS, the observed transverse momentum P_{hT} of the produced charged hadron with respect to the virtual photon momentum plays the role of the small scale Q_1 , while the virtual photon virtuality Q plays the role of the large scale Q_2 . In the TMD description, the observed transverse momentum P_{hT} is related to the active parton (quark) intrinsic transverse momentum k_{\perp} and fragmenting quark p_{\perp} as well as to the effects of the gluon radiation encoded in the evolution equations [9]. By utilizing juxtapositions of the lepton beam and nucleon target polarizations in SIDIS experiments, we will be capable of extracting pivotal information on various TMDs, whereby one can quantify the quark transverse motion inside the nucleon and observe spin-orbit correlations, as well as obtain quantitative insight on the quark orbital angular momentum (OAM) contribution to the proton spin. Note that the results coming from advances in target polarization techniques, like those employed by HERMES, COMPASS, LHCb and JLab experiments [13–17], are crucial in studies of TMDs together with the additional information that can be accessed from the Drell-Yan process [12] and e^+e^- annihilation [2].

However, in SIDIS experiments devoted to data extractions on TMDs, such as HERMES [18–22], COMPASS [23–27], and JLab [28, 29], it is indispensable to have a very good control over systematic uncertainties. In this regard, one of the dominant sources of the systematic uncertainties in these experiments, with and without polarization of the lepton beam and the nucleon target, are the radiative corrections (RC) due to the radiation of photons off leptons. Radiative corrections are calculated based on Quantum Electrodynamics (QED). Exact analytical formulas for the lowest-order model-independent part of QED RCs to the SIDIS cross section, with a longitudinally polarized lepton beam and arbitrarily polarized nucleon target, beyond ultra-relativistic approximation², are calculated in Ref. [30]. The methodology developed in that paper is anchored upon the covariant approach for RC calculations, so that the obtained formulas can be directly applied to any coordinate system. Besides, the Bardin-Shumeiko approach

²It means that the incident beam lepton mass is taken into account in all calculations.

[31, 32] is applied for covariant extraction and cancellation of the infrared divergence that stem from the real and virtual photon emissions in the SIDIS process. Within this approach, the RC cross-section results do not depend on an artificial parameter, which is usually introduced to separate the photon emission on hard and soft parts [33]. The calculations of Ref. [30] have been performed in a model-independent manner, involving the SIDIS cross section that contains eighteen SIDIS structure functions [6, 10]³. Note that an alternative method of accounting for QED radiative corrections in SIDIS based on factorization approach was introduced in Ref. [36]. It will be interesting to compare results of the two methods [37].

There exist other Monte-Carlo (MC) frameworks for evaluation of RC corrections in SIDIS. One can successfully evaluate RC effects by having information obtained during event generation, and such a generator can be constructed by utilizing a code for RC in SIDIS, in a similar way as the program RADGEN 1.0 [38, 39] was constructed based on the program POLRAD 2.0 [39–41]. RADGEN 1.0 is a MC generator of polarized/unpolarized DIS radiative events, which can be applied for RC generation in inclusive, semi-inclusive and exclusive DIS processes. POLRAD 2.0 is a FORTRAN code for treating experimental data with an implemented RC procedure, in polarized inclusive and semi-inclusive DIS. Note that there is also HAPRAD 2.0, a FORTRAN code for RC studies in the semi-inclusive hadron electroproduction processes, which has available versions with MC and numerical integration [41, 42]. HAPRAD 2.0 in turn stems from the original HAPRAD program [39, 41, 43, 44], which performs RC calculations to the five-fold differential cross section of unpolarized particles, $d\sigma_{\text{SIDIS}}/dx_{Bj}dydz_h dP_{hT}^2 d\phi_h$. In addition to the already specified variables x_{Bj} , z_h , P_{hT} , here y is the fraction of the lepton beam energy carried by the virtual photon, and ϕ_h is the hadron azimuthal angle measured with respect to the lepton scattering plane in the SIDIS process. We will discuss the SIDIS process in details in Sec. 2.1.1.

However, we now have exact and more general results on the six-fold differential cross section, $d\sigma_{\text{SIDIS}}/dx_{Bj}dydz_h dP_{hT}^2 d\phi_h d\phi_s$ (obtained in [30]),

³The results of Ref. [30] also include a contribution of the exclusive radiative tail to the lowest-order RC to the SIDIS process, however, currently the corresponding exclusive structure functions are not known yet. More theoretical [34] and [35] experimental efforts are needed for their determination, as well as for the complete derivations of all SIDIS structure functions.

where ϕ_s is another azimuthal angle describing the target-spin direction (spin-vector), if transversely polarized targets are applied. With all the past developments our task was to create a MC event generator that for the SIDIS process would be the following:

- (i) generate radiative and non-radiative channels of scattering: namely, the generated events could be selected to be either radiative or non-radiative, with a probability of being proportional to the radiative/non-radiative cross section;
- (ii) generate scattered lepton kinematics (i.e., Q^2 and x_{Bj}) and final-state hadron kinematics (i.e., z_h , P_{hT} , and ϕ_h);
- (iii) generate radiation real photon kinematics;
- (iv) calculate the full SIDIS cross section in any generated phase-space point with RCs included.

In this paper, we introduce the event generator **SIDIS-RC EvGen**, which includes implementation of the items (i) – (iv), based on the SIDIS RC studies in [30], and on using TMD PDFs and TMD FFs discussed extensively in [10]. The full package of **SIDIS-RC EvGen** can be found in [45], along with the installation and running instructions.

The generated final-state hadrons can be fed into detector simulations, allowing precise predictions and verification for some key aspects of entire experimental setups, such as SoLID, CLAS12 at JLab [46–48], EIC at BNL [49, 97], etc.. **SIDIS-RC EvGen** can be considered as one of the tools similar to HAPRAD 2.0, mLEPTO, LEPTO-PHI, the NJL-jet model-based generator (see [50, 51] for more details), which are aimed to help in multifaceted efforts study the internal spin structure, TMD evolution effects, spin-orbit and quark-gluon correlations, and the 3D momentum structure of the nucleon in general.

In this regard and by having all the above discussion in mind, let us present how this paper is structured in what follows. In Sec. 2, we discuss the SIDIS process, including the TMD PDFs, TMD FFs, and structure functions that are essential to study the 3D tomography of the nucleon in momentum space. We then discuss some key aspects of the lowest-order QED radiative effects in SIDIS, describing each cross-section component that contributes to the total inelastic cross section. In Sec. 3, we show how **SIDIS-RC EvGen**

is constructed, how the events are generated and cross sections computed. Our focus is in particular on a library component `libsidis` for calculating the total SIDIS cross section, and on a generator (binary) component `sidisgen` for generating events. We continue this discussion in Sec. 4, where we exhibit some numerical results obtained from the event generator. We notably demonstrate several results on the SIDIS final-state charged-hadron azimuthal transverse target single-spin asymmetries (SSAs). In the final Sec. 5, we give a summary of our work and also outline some prospects for a further development.

2. Some basics of the theoretical framework for studying 3D momentum structure of the nucleon with SIDIS

In this section, we discuss several aspects of a theoretical framework by which nucleon tomography is studied with SIDIS. On the other hand, we show all necessary formulas and variables/definitions, which underlie the structure of SIDIS-RC EvGen, and whereby the event generation and cross-section calculations are carried out. The discussion of Sec. 2.1 is partially based upon [10]. Its results are represented in a pertinent MATHEMATICA-implemented library called WW-SIDIS [52], which we use in SIDIS-RC EvGen. In Sec. 2.2, we aim to discuss the primary results of [30] since the main goal for building SIDIS-RC EvGen is to produce the next-to-leading order (NLO) RCs to the SIDIS cross section. We also extensively refer to [10] and [30], wherever it is relevant, such that the reader can obtain complete information regarding any specific point or matter under discussion.

2.1. SIDIS process and functions describing the nucleon structure

In this Section we will outline the basics of the SIDIS process following the formalism and notations described in Refs. [6, 10].

2.1.1. Kinematics and cross section of the SIDIS process

We are interested in the SIDIS process, in which a final-state hadron h is detected in coincidence with a lepton ℓ' scattered off a target N . That process is given by [6, 9, 10, 53]

$$\ell(k_1) + N(P) \rightarrow \ell'(k_2) + h(P_h) + X(P_X), \quad (1)$$

where k_1 and P are the four-momenta of both polarized/unpolarized incident lepton and nucleon target, k_2 and P_h are the four-momenta of the scattered

lepton ℓ' and the produced detected hadron h , P_X is the four-momentum of the unobserved state consisting of all undetected hadrons produced in the reaction. In this scattering process, the respective nucleon and hadron masses are given by M_N and M_h . This process is depicted in Fig. 1, made under the assumption of the one-photon exchange approximation, where the reference frame is adopted to be the γ^* - N center-of-mass frame, in which the virtual photon moves in the positive z direction and the azimuthal angles are counted from the lepton scattering plane. The cross section of the process is expressed by the following kinematic invariants:

$$x_{Bj} = \frac{Q^2}{2P \cdot q}, \quad y = \frac{P \cdot q}{P \cdot k_1}, \quad z_h = \frac{P \cdot P_h}{P \cdot q}, \quad \gamma = \frac{2M_N x_{Bj}}{Q}, \quad (2)$$

with q defined to be the virtual photon momentum $q \equiv k_1 - k_2$, and Q^2 is the virtuality (hard scale) $Q^2 \equiv -q^2$.

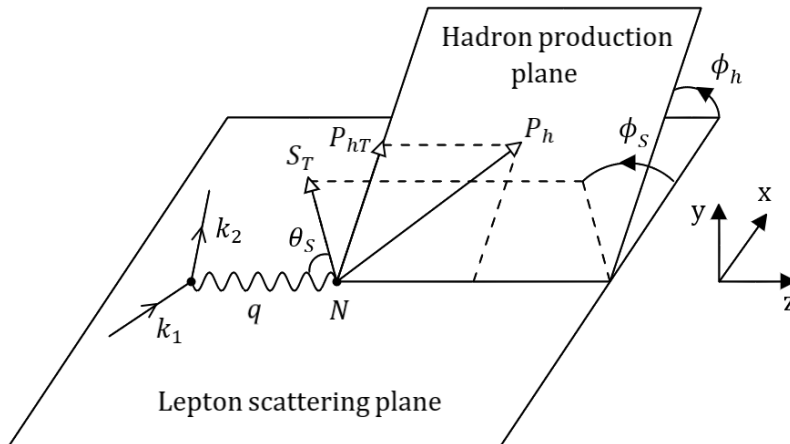


Figure 1: Kinematics of the SIDIS process in the one-photon exchange approximation, and the sketch is made by following the Trento conventions in [53]. The overall orientation of the lepton scattering plane around the incident lepton direction is characterized by an azimuthal angle ϕ_l taken with respect to an arbitrary reference frame, which however, in the DIS limit and for a transversely polarized target is $\phi_l \approx \phi_s$.

Thus, assuming the one-photon exchange approximation, the SIDIS differential cross section is expressed by a set of eighteen structure functions (which can be computed within the TMD factorization framework in the region

of validity of TMD factorization itself) in a model-independent way [6, 54]:

$$\begin{aligned}
\frac{d\sigma_{\text{SIDIS}}}{dx_{Bj} dy dz_h dP_{hT}^2 d\phi_h d\phi_s} &= \frac{\alpha^2}{x_{Bj} y Q^2} \left(1 + \frac{\gamma^2}{2x_{Bj}} \right) \times \\
&\times \left\{ \left[c_1 F_{UU,T} + c_2 F_{UU,L} + c_3 \cos(\phi_h) F_{UU}^{\cos(\phi_h)} + \right. \right. \\
&\quad \left. \left. + c_2 \cos(2\phi_h) F_{UU}^{\cos(2\phi_h)} + \lambda_e c_4 \sin(\phi_h) F_{LU}^{\sin(\phi_h)} \right] + \right. \\
&\quad \left. + S_L \left[c_3 \sin(\phi_h) F_{UL}^{\sin(\phi_h)} + c_2 \sin(2\phi_h) F_{UL}^{\sin(2\phi_h)} \right] + \right. \\
&\quad \left. + S_L \lambda_e \left[c_5 F_{LL} + c_4 \cos(\phi_h) F_{LL}^{\cos(\phi_h)} \right] + \right. \\
&\quad \left. + S_T \left[\sin(\phi_h - \phi_s) \left(c_1 F_{UT,T}^{\sin(\phi_h - \phi_s)} + c_2 F_{UT,L}^{\sin(\phi_h - \phi_s)} \right) + \right. \right. \\
&\quad \left. \left. + c_2 \sin(\phi_h + \phi_s) F_{UT}^{\sin(\phi_h + \phi_s)} + c_2 \sin(3\phi_h - \phi_s) F_{UT}^{\sin(3\phi_h - \phi_s)} + \right. \right. \\
&\quad \left. \left. + c_3 \sin(\phi_s) F_{UT}^{\sin(\phi_s)} + c_3 \sin(2\phi_h - \phi_s) F_{UT}^{\sin(2\phi_h - \phi_s)} \right] + \right. \\
&\quad \left. + S_T \lambda_e \left[c_5 \cos(\phi_h - \phi_s) F_{LT}^{\cos(\phi_h - \phi_s)} + c_4 \cos(\phi_s) F_{LT}^{\cos(\phi_s)} + \right. \right. \\
&\quad \left. \left. + c_4 \cos(2\phi_h - \phi_s) F_{LT}^{\cos(2\phi_h - \phi_s)} \right] \right\}, \tag{3}
\end{aligned}$$

where α is the fine-structure constant, and λ_e is the helicity of the lepton beam. In the XY subscripts of the most structure functions, the first one $X = U/L$ refers to the unpolarized or longitudinally polarized beam (with λ_e). The second one, $Y = U/L$ or ($= U/T$), is correspondingly ascribed to the unpolarized or longitudinally polarized (or transversely polarized) target with respect to q . In the XY, Z subscripts of the remaining four structure functions, $Z = T/L$ specifies the virtual photon polarizations. The superscripts in structure functions (e.g., $\cos(\phi_h)$ in $F_{UU}^{\cos(\phi_h)}$) denote the corresponding azimuthal dependence, and when the superscript is absent (i.e., in F_{LL}), the dependence is flat.

All structure functions in the R.H.S. of Eq. (3) are functions of x_{Bj} , Q^2 , z_h , and P_{hT}^2 . Eq. (3) corresponds to a SIDIS process, where the final hadron polarization is not measured, or a spin-0 hadron is produced such as the pion. The factors c_1 , c_2 , c_3 , c_4 , and c_5 (see eqs. (2.9)-(2.13) of [6] for more details) are given by

$$\begin{aligned} c_1 &= \frac{y^2}{2(1-\varepsilon)}, & c_2 &= \frac{y^2}{2(1-\varepsilon)}\varepsilon, & c_3 &= \frac{y^2}{2(1-\varepsilon)}\sqrt{2\varepsilon(1+\varepsilon)}, \\ c_4 &= \frac{y^2}{2(1-\varepsilon)}\sqrt{2\varepsilon(1-\varepsilon)}, & c_5 &= \frac{y^2}{2(1-\varepsilon)}\sqrt{1-\varepsilon^2}, \end{aligned} \quad (4)$$

where ε is the ratio of longitudinal and transverse photon fluxes:

$$\varepsilon = \frac{1-y-(\gamma^2 y^2/4)}{1-y+(y^2/2)+(\gamma^2 y^2/4)}. \quad (5)$$

The hadron azimuthal angle is defined in [6, 53]:

$$\cos(\phi_h) = -\frac{k_{1\mu} P_{h\nu} g_{\perp}^{\mu\nu}}{\sqrt{k_{1T}^2 P_{hT}^2}}, \quad \sin(\phi_h) = -\frac{k_{1\mu} P_{h\nu} \epsilon_{\perp}^{\mu\nu}}{\sqrt{k_{1T}^2 P_{hT}^2}}, \quad (6)$$

with $k_{1T}^{\mu} = g_{\perp}^{\mu\nu} k_{1\nu}$ and $P_{hT}^{\mu} = g_{\perp}^{\mu\nu} P_{h\nu}$ to be the transverse components of l and P_h with respect to q . The tensors $g_{\perp}^{\mu\nu}$ and $\epsilon_{\perp}^{\mu\nu}$ are respectively expressed by

$$g_{\perp}^{\mu\nu} = g^{\mu\nu} - \frac{q^{\mu} P^{\nu} + P^{\mu} q^{\nu}}{P \cdot q (1 + \gamma^2)} + \frac{\gamma^2}{1 + \gamma^2} \left(\frac{q^{\mu} q^{\nu}}{Q^2} - \frac{P^{\mu} P^{\nu}}{M_N^2} \right), \quad (7)$$

$$\epsilon_{\perp}^{\mu\nu} = \epsilon^{\mu\nu\rho\sigma} \frac{P_{\rho} q_{\sigma}}{P \cdot q \sqrt{1 + \gamma^2}}, \quad (8)$$

where the definition of the totally antisymmetric tensor is $\epsilon^{0123} = +1$, and the non-zero components of these tensors are $g_{\perp}^{11} = g_{\perp}^{22} = -1$, $\epsilon_{\perp}^{12} = -\epsilon_{\perp}^{21} = +1$. Note that the target covariant spin-vector, Sv , is decomposed as

$$\begin{aligned} Sv^{\mu} &= S_L \frac{P^{\mu} - [q^{\mu} M_N^2 / (P \cdot q)]}{M_N \sqrt{1 + \gamma^2}} + S_T^{\mu}, \\ \text{with } S_L &= \frac{Sv \cdot q}{P \cdot q} \frac{M_N}{\sqrt{1 + \gamma^2}}, \quad \text{and } S_T^{\mu} = g_{\perp}^{\mu\nu} Sv_{\nu}, \end{aligned} \quad (9)$$

and for the transversely polarized targets, ϕ_s is defined as

$$\cos(\phi_s) = -\frac{k_{1\mu} S_\nu g_\perp^{\mu\nu}}{\sqrt{k_{1T}^2 S_T^2}}, \quad \sin(\phi_s) = -\frac{k_{1\mu} S_\nu \epsilon_\perp^{\mu\nu}}{\sqrt{k_{1T}^2 S_T^2}}. \quad (10)$$

In the partonic description of the SIDIS process, the functions $F_{UU,T}$, $F_{UU}^{\cos(2\phi_h)}$, $F_{UL}^{\sin(2\phi_h)}$, F_{LL} , $F_{UT,T}^{\sin(\phi_h-\phi_s)}$, $F_{UT}^{\sin(\phi_h+\phi_s)}$, $F_{UT}^{\sin(3\phi_h-\phi_s)}$, and $F_{LT}^{\cos(\phi_h-\phi_s)}$, are “twist-2” (leading-twist) structure functions at leading order in the $1/Q$ expansion in Eq. (3). The functions $F_{UU}^{\cos(\phi_h)}$, $F_{LU}^{\sin(\phi_h)}$, $F_{UL}^{\sin(\phi_h)}$, $F_{LL}^{\cos(\phi_h)}$, $F_{UT}^{\sin(2\phi_h-\phi_s)}$, $F_{UT}^{\sin(\phi_s)}$, $F_{LT}^{\cos(\phi_s)}$, and $F_{LT}^{\cos(2\phi_h-\phi_s)}$ are “twist-3” (subleading-twist) structure functions at subleading order in the $1/Q$ expansion of the same formula. The other remaining two structure functions, namely $F_{UU,L}$ and $F_{UT,L}^{\sin(\phi_h-\phi_s)}$ contribute to the process at the order of $\mathcal{O}(1/Q^2)$ due to the longitudinal virtual-photon polarization, and hence can be neglected if one considers $\mathcal{O}(1/Q)$ accuracy.

2.1.2. TMD PDFs, TMD FFs, and SIDIS structure functions

In the Bjorken limit ($Q^2 \rightarrow \infty$ and $P \cdot q \rightarrow \infty$ with x_{Bj} fixed), the SIDIS structure functions in Eq. (3) are described at tree level in terms of convolution integrals of the unpolarized TMD PDF, $f^a(x, k_\perp^2)$, and TMD FF, $D^a(z, p_\perp^2)$ ⁴, for a given quark flavor a , Ref. [10], as:

$$\begin{aligned} \mathcal{C}[\omega f D] = x \sum_a e_a^2 \int d^2 \mathbf{k}_\perp d^2 \mathbf{p}_\perp \delta^{(2)}(z \mathbf{k}_\perp + \mathbf{p}_\perp - \mathbf{P}_{hT}) \times \\ \times \omega\left(\mathbf{k}_\perp, -\frac{\mathbf{p}_\perp}{z}\right) f^a(x, k_\perp^2) D^a(z, p_\perp^2), \end{aligned} \quad (11)$$

where ω is a weight function depending on \mathbf{k}_\perp and \mathbf{p}_\perp generally, along with a given defined unit vector $\hat{\mathbf{h}} = \mathbf{P}_{hT}/P_{hT}$. The integrals of the eight leading-twist and eight subleading-twist structure functions are all given in [10] by eqs. (2.17a)-(2.17h) and eqs. (2.18a)-(2.18h), respectively, along with the weight functions defined in eq. (2.19).

We first discuss the Gaussian Ansatz for the TMDs and FFs⁵, which is well supported by phenomenological analyses, such as [55–59]. All convolution integrals of the same type as in Eq. (11) can be solved analytically with

⁴The light-cone momentum fractions $x = k^+/P^+$ and $z = P_h^-/\kappa^-$ are identified with x_{Bj} and z_h up to the order of k_\perp/Q , where κ is the fragmentation quark momentum [9].

⁵We use shorthand notations: TMDs for TMD PDFs, and FFs for TMD FFs.

this ansatz. In this case, $f^a(x_{Bj}, k_\perp^2)$ and $D^a(z_h, p_\perp^2)$ are expressed as

$$f^a(x_{Bj}, k_\perp^2) = f_c^a(x_{Bj}) \frac{e^{-k_\perp^2/\langle k_\perp^2 \rangle}}{\pi \langle k_\perp^2 \rangle}, \quad D^a(z_h, p_\perp^2) = D_c^a(z_h) \frac{e^{-p_\perp^2/\langle p_\perp^2 \rangle}}{\pi \langle p_\perp^2 \rangle}, \quad (12)$$

where $f_c^a(x_{Bj})$ is the collinear PDF, and $D_c^a(z_h)$ is the collinear FF. For both collinear PDF and FF, we utilize the grid files taken from WW-SIDIS [52] and MSTWPDF [60, 61] libraries. Alternatively, one could use the LHAPDF CJ15lo set from [62] for the PDF, and DSSFFlo set used in [63] for the FF. The Gaussian widths $\langle k_\perp^2 \rangle$ and $\langle p_\perp^2 \rangle$ may have different forms of kinematic dependence, however, we will assume them to be flavor-, x_{Bj} -, and z_h -independent for simplicity.

All leading- and subleading-twist SIDIS structure functions in Eq. (3) are described in terms of a basis of six TMDs and two FFs under the assumption of validity of WW(-type) approximations to be discussed shortly. Two of these basis functions are given by Eq. (12), and the rest are $f_{1T}^{\perp a}$, g_1^a , h_1^a , $h_1^{\perp a}$, $h_{1T}^{\perp a}$, $H_1^{\perp a}$ (see their explicit forms in eqs. (4.5c)-(4.5h) of [10]). Now, as examples, let us write down the general analytical forms of three structure functions in the Gaussian approximation, which are used to produce some numerical results from SIDIS-RC EvGen, to be shown in Sec. 4:

$$\begin{aligned} \text{(i) Leading-twist } F_{UU}(x, z, P_{hT}) &= \{F_{UU} \equiv F_{UU,T}\} \\ &= x_{Bj} \sum_a e_a^2 f_c^a(x_{Bj}) D_c^a(z_h) \frac{e^{-P_{hT}^2/\langle P_{hT}^2 \rangle}}{\pi \langle P_{hT}^2 \rangle}, \end{aligned} \quad (13)$$

where $\langle P_{hT}^2 \rangle = \langle p_\perp^2 \rangle_D + z_h^2 \langle k_\perp^2 \rangle_f$. See table 1 of appendix A in [10] for the values of $\langle k_\perp^2 \rangle_f$ and $\langle p_\perp^2 \rangle_D$.

$$\begin{aligned} \text{(ii) Leading-twist } F_{UT}^{\sin(\phi_h + \phi_S)}(x, z, P_{hT}) &= \\ &= x_{Bj} \sum_a e_a^2 h_1^a(x_{Bj}) H_1^{\perp(1)a}(z_h) b_A^{(1)} \left[\frac{z_h P_{hT}}{\langle P_{hT}^2 \rangle} \right] \frac{e^{-P_{hT}^2/\langle P_{hT}^2 \rangle}}{\pi \langle P_{hT}^2 \rangle}, \end{aligned} \quad (14)$$

where $\langle P_{hT}^2 \rangle = \langle p_\perp^2 \rangle_{H_1^\perp} + z_h^2 \langle k_\perp^2 \rangle_{h_1}$, $\langle p_\perp^2 \rangle_{H_1^\perp} = [\langle p_\perp^2 \rangle_D M_C^2] / [\langle p_\perp^2 \rangle_D + M_C^2]$, and $b_A^{(1)} = 2M_h$. The values of $\langle k_\perp^2 \rangle_{h_1}$ and M_C^2 are given in Table 3, and the first moment of the Collins fragmentation function, $H_1^{\perp(1)a}(z_h)$, is given in

eq. (A.12), of appendix A in [10].

$$\begin{aligned}
\text{(iii) Leading-twist } F_{UT}^{\sin(\phi_h - \phi_S)}(x, z, P_{hT}) &= \left\{ F_{UT}^{\sin(\phi_h - \phi_S)} \equiv F_{UT,T}^{\sin(\phi_h - \phi_S)} \right\} \\
&= -x_{B_j} \sum_a e_a^2 f_{1T}^{\perp(1)a}(x_{B_j}) D_c^a(z_h) b_B^{(1)} \left[\frac{z_h P_{hT}}{\langle P_{hT}^2 \rangle} \right] \frac{e^{-P_{hT}^2 / \langle P_{hT}^2 \rangle}}{\pi \langle P_{hT}^2 \rangle}, \quad (15)
\end{aligned}$$

where $\langle P_{hT}^2 \rangle = \langle p_{\perp}^2 \rangle_D + z_h^2 \langle k_{\perp}^2 \rangle_{f_{1T}^{\perp}}$, $\langle k_{\perp}^2 \rangle_{f_{1T}^{\perp}} = [\langle k_{\perp}^2 \rangle_f M_1^2] / [\langle k_{\perp}^2 \rangle_f + M_1^2]$, and $b_B^{(1)} = 2M_N$. The value of M_1^2 is given in Table 2, and the first moment of the Sivers function, $f_{1T}^{\perp(1)a}(x_{B_j})$, is given in eq. (A.4), of appendix A in [10].

Twist-2 TMDs, like $g_1^b(x)$ ($b = q, g, \bar{q}$) and $h_1^c(x)$ ($c = q, \bar{q}$), have partonic interpretation. Twist-3 TMDs, like $g_T^b(x)$ and $h_L^c(x)$, can respectively be expressed by $g_1^b(x)$ and $h_1^c(x)$ plus additional $\bar{q}gq$ correlations. These correlations may have new insights on hadronic structure but their contributions to $g_T^b(x)$ and $h_L^c(x)$ are small. In the case of TMDs, FFs and collinear PDFs, one can assume that the $\bar{q}gq$ -correlation contributions are negligible compared with those of $\bar{q}q$ correlations, which makes an approximation named Wandzura-Wilczek (WW) [64]: $|\langle \bar{q}gq \rangle / \langle \bar{q}q \rangle| \ll 1$. Each correlation comprises matrix elements of different operators, and the nature of neglected matrix elements is different. Based upon this difference, one generally considers WW-type approximations in the context of TMDs/FFs. At this point we refer to section 3.2 of [10] for more detailed discussion of the WW-type approximations for TMDs/FFs. Besides, in sections 4.1 and 4.2 of the same reference, it is shown how some of the leading-twist and all the subleading-twist structure functions are treated in these approximations.

Except for F_{UU} , $F_{UT}^{\sin(\phi_h + \phi_S)}$, and $F_{UT}^{\sin(\phi_h - \phi_S)}$ described in the paragraphs (i), (ii), and (iii), the other structure functions given in the Gaussian Ansatz with combination of the WW-type approximations wherever relevant, can be found in [10] as follows (see also Sec. 3.2.3):

For leading-twist:

- $F_{UU}^{\cos(2\phi_h)}$ - see section 5.5; eqs. (A.12), (A.18); appendix B.5;
- $F_{UL}^{\sin(2\phi_h)}$ - see section 6.2; eqs. (A.12), (B.9b), (3.6b); appendix B.5;
- F_{LL} - see section 5.2; appendix A.2; eq. (4.5c);
- $F_{UT}^{\sin(3\phi_h - \phi_S)}$ - see section 5.6; eq. (A.12); appendix A.6, appendix B.5;

- $F_{LT}^{\cos(\phi_h - \phi_S)}$ - see section 6.1; eqs. (B.9a), (3.6a); appendix B.5.

For subleading-twist:

- $F_{UU}^{\cos(\phi_h)}$ - see section 7.8; eq. (B.9j); appendix B.5;
- $F_{LU}^{\sin(\phi_h)}$ - see section 7.1 (also section III of [65]);
- $F_{UL}^{\sin(\phi_h)}$ - see section 7.5; eqs. (A.12), (B.9f), (3.6b); appendix B.5;
- $F_{LL}^{\cos(\phi_h)}$ - see section 7.4; eq. (B.9e); appendix B.5;
- $F_{UT}^{\sin(2\phi_h - \phi_S)}$ - see section 7.7; eqs. (B.9g), (B.9h), (B.9i); appendix B.5;
- $F_{UT}^{\sin(\phi_S)}$ - see section 7.6; eqs. (A.12), (B.9g), (B.9h), (3.3g), (3.3h);
- $F_{LT}^{\cos(\phi_S)}$ - see section 7.2; eqs. (B.9c), (3.2a);
- $F_{LT}^{\cos(2\phi_h - \phi_S)}$ - see section 7.3; eq. (B.9d), (3.3d), (3.6a); appendix B.5.

All the other details necessary for making our event generator is provided in Sec. 3.2.

2.2. Lowest-order QED radiative effects in SIDIS

In this section, we discuss the main results of [30] for calculation of the NLO RCs to the SIDIS base cross section.

2.2.1. Leading order cross section

Let us represent Eq. (1) in a bit modified way:

$$\ell(k_1, \xi) + N(P, \boldsymbol{\eta}) \rightarrow \ell'(k_2) + h(P_h) + X(P_X), \quad (16)$$

where ξ and $\boldsymbol{\eta}$ are the incident lepton and target nucleon polarization vectors, respectively, described by x_{Bj} , y , z_h , ϕ_h , ϕ_S , and the variable $t = (q - P_h)^2$. Fig. 2 shows the lowest-order QED (Born) contribution to SIDIS, and the differential of the cross section of this process is given by a convolution of the hadronic ($W_{\mu\nu}$) and leptonic tensors ($L_B^{\mu\nu}$):

$$d\sigma_{\text{SIDIS}}^B = \frac{(4\pi\alpha)^2}{2\sqrt{\lambda_S}Q^4} W_{\mu\nu} L_B^{\mu\nu} d\Gamma_B, \quad (17)$$

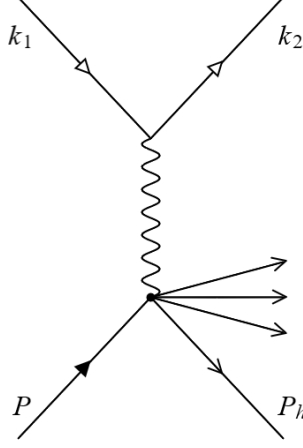


Figure 2: A Feynman diagram describing the base (lowest-order QED) SIDIS process.

where

$$\lambda_S = (2P \cdot k_1)^2 - 4M_N^2 m_l^2, \quad (18)$$

with m_l being the lepton mass. The phase-space is parameterized by

$$\begin{aligned} d\Gamma_B &= (2\pi)^4 \frac{d^3 k_2}{(2\pi)^3 2k_{20}} \frac{d^3 P_h}{(2\pi)^3 2P_{h0}} = \\ &= \frac{1}{4(2\pi)^2} \frac{SS_x dx_{Bj} dy d\phi_S}{2\sqrt{\lambda_S}} \frac{S_x dz_h dP_{hT}^2 d\phi_h}{4M_N P_{hL}}, \end{aligned} \quad (19)$$

where k_{20} is the scattered lepton energy, P_{h0} (P_{hL}) is the energy (longitudinal momentum) of the charged hadron. Also,

$$\begin{aligned} S &= 2P \cdot k_1, \quad S_x = 2P \cdot q, \quad P_{h0} = \frac{z_h S_x}{2M_N}, \quad P_{hT} = \sqrt{P_{h0}^2 - P_{hL}^2 - M_h^2}, \\ P_{hL} &= \frac{z_h S_x^2 + 2M_N^2 (t + Q^2 - M_h^2)}{2M_N \sqrt{\lambda_Y}}, \quad \text{with } \lambda_Y = S_x^2 + 4M_N^2 Q^2. \end{aligned} \quad (20)$$

The incident lepton is adopted to be longitudinally polarized. Its polarization vector reads as

$$\xi = \frac{\lambda_e S}{m_l \sqrt{\lambda_S}} k_1 - \frac{2\lambda_e m_l}{\sqrt{\lambda_S}} P = \xi_0 + \xi_1, \quad (21)$$

and the leptonic tensor is represented as

$$\begin{aligned} L_B^{\mu\nu} &= \frac{1}{2} \text{Tr} \left[(\hat{k}_2 + m_l) \gamma_\mu (\hat{k}_1 + m_l) (1 + \gamma_5 \hat{\xi}) \gamma_\nu \right] = \\ &= 2 \left[k_1^\mu k_2^\nu + k_2^\mu k_1^\nu - \frac{Q^2}{2} g^{\mu\nu} + \frac{i\lambda_e}{\sqrt{\lambda_S}} \epsilon^{\mu\nu\rho\sigma} (S k_{2\rho} k_{1\sigma} + 2m_l^2 q_\rho P_\sigma) \right]. \end{aligned} \quad (22)$$

As regards the hadronic tensor, it is partitioned to spin-independent, $H_{ab}^{(0)}$, and spin-dependent, $H_{abi}^{(S)}$, scalar structure functions:

$$W_{\mu\nu} = \sum_{a,b=0}^3 e_{\mu}^{\gamma(a)} e_{\nu}^{\gamma(b)} \left(H_{ab}^{(0)} + \sum_{\rho,i=0}^3 \eta^{\rho} e_{\rho}^{h(i)} H_{abi}^{(S)} \right), \quad (23)$$

where $e_{\mu}^{\gamma(a)}$ is the complete set of the basis for polarization four-vectors of the virtual photon, and $e_{\rho}^{h(i)}$ is that of the target nucleon⁶. There are five spin-independent $H_{00}^{(0)}$, $H_{11}^{(0)}$, $H_{22}^{(0)}$, $\text{Re}H_{01}^{(0)}$, $\text{Im}H_{01}^{(0)}$, as well as thirteen spin-dependent $H_{002}^{(S)}$, $\text{Re}H_{012}^{(S)}$, $\text{Im}H_{012}^{(S)}$, $\text{Re}H_{021}^{(S)}$, $\text{Im}H_{021}^{(S)}$, $\text{Re}H_{023}^{(S)}$, $\text{Im}H_{023}^{(S)}$, $H_{112}^{(S)}$, $\text{Re}H_{121}^{(S)}$, $\text{Im}H_{121}^{(S)}$, $\text{Re}H_{123}^{(S)}$, $\text{Im}H_{123}^{(S)}$, $H_{222}^{(S)}$, scalar functions.

The cross section of the Born contribution to SIDIS is given by

$$\frac{d\sigma_{\text{SIDIS}}^B}{dx_{Bj} dy dz_h dP_{hT}^2 d\phi_h d\phi_S} = \frac{\alpha^2 S S_x^2}{8M_N Q^4 P_{hL} \lambda_S} \sum_{i=1}^9 \theta_i^B \mathcal{H}_i, \quad (24)$$

where θ_i^B is expressed via the combination of various kinematic variables; \mathcal{H}_i is the generalized structure function, expressed through $H_{ab}^{(0)}$ and $H_{abi}^{(S)}$ that is based on the nucleon polarized three-vector $\boldsymbol{\eta} = (\eta_1, \eta_2, \eta_3)$ decomposition over the basis in eq. (A2). The components of $\boldsymbol{\eta}$ are the following:

$$\eta_1 = \cos(\phi_S - \phi_h) S_T, \quad \eta_2 = \sin(\phi_S - \phi_h) S_T, \quad \eta_3 = S_L. \quad (25)$$

\mathcal{H}_i and θ_i^B are explicitly shown in eq. (14) and eq. (16) of [30].

At this point, it is relevant to show here the relations of the scalar functions with the structure functions discussed in Sec. 2:

$$\begin{aligned} H_{00}^{(0)} &= C_1 F_{UU,L}, \\ H_{01}^{(0)} &= -C_1 \left(F_{UU}^{\cos(\phi_h)} + i F_{LU}^{\sin(\phi_h)} \right), \\ H_{11}^{(0)} &= C_1 \left(F_{UU}^{\cos(2\phi_h)} + F_{UU,T} \right), \\ H_{22}^{(0)} &= C_1 \left(F_{UU,T} - F_{UU}^{\cos(2\phi_h)} \right), \end{aligned}$$

⁶For the explicit expressions of both basis vectors, see eq. (A1) and eq. (A2) of [30].

$$\begin{aligned}
H_{002}^{(S)} &= C_1 F_{UT,L}^{\sin(\phi_h - \phi_S)}, \\
H_{012}^{(S)} &= C_1 \left(F_{UT}^{\sin(\phi_S)} - F_{UT}^{\sin(2\phi_h - \phi_S)} - i \left(F_{LT}^{\cos(\phi_S)} - F_{LT}^{\cos(2\phi_h - \phi_S)} \right) \right), \\
H_{021}^{(S)} &= C_1 \left(F_{UT}^{\sin(2\phi_h - \phi_S)} + F_{UT}^{\sin(\phi_S)} - i \left(F_{LT}^{\cos(2\phi_h - \phi_S)} + F_{LT}^{\cos(\phi_S)} \right) \right), \\
H_{023}^{(S)} &= C_1 \left(F_{UL}^{\sin(\phi_h)} - i F_{LL}^{\cos(\phi_h)} \right), \\
H_{121}^{(S)} &= C_1 \left(-F_{UT}^{\sin(3\phi_h - \phi_S)} - F_{UT}^{\sin(\phi_h + \phi_S)} + i F_{LT}^{\cos(\phi_h - \phi_S)} \right), \\
H_{123}^{(S)} &= C_1 \left(-F_{UL}^{\sin(2\phi_h)} + i F_{LL} \right), \\
H_{112}^{(S)} &= C_1 \left(F_{UT}^{\sin(3\phi_h - \phi_S)} + F_{UT,T}^{\sin(\phi_h - \phi_S)} - F_{UT}^{\sin(\phi_h + \phi_S)} \right), \\
H_{222}^{(S)} &= C_1 \left(F_{UT}^{\sin(\phi_h + \phi_S)} + F_{UT,T}^{\sin(\phi_h - \phi_S)} - F_{UT}^{\sin(3\phi_h - \phi_S)} \right). \tag{26}
\end{aligned}$$

We emphasize that if one uses the formulas in eq. (14) (for \mathcal{H}_i), and eq. (16) (for θ_i^B) of [30], as well as the formulas in Eq. (26) and inserts them in the R.H.S. of Eq. (24), then the result will be equivalent to the R.H.S. of Eq. (3). Therefore our two ways of expressing the SIDIS cross-section are equivalent.

2.2.2. Lowest-order radiative corrections to the SIDIS cross section

We can now discuss the real photon emission in the SIDIS scattering process, given by

$$\ell(k_1, \xi) + N(P, \boldsymbol{\eta}) \rightarrow \ell'(k_2) + h(P_h) + X(\tilde{P}_X) + \gamma(k), \tag{27}$$

where k is the four-momentum of the radiated real photon γ . The four diagrams depicted in Figs. 3(a)-3(d) are the lowest-order QED RC contributions to the cross section of the base SIDIS process shown in Fig. 2. The real photon is shown in the figures (a) and (b). The considered process is described by all kinematic variables of Eq. (16), plus three additional photonic variables that are

$$R = 2k \cdot P, \quad \tau = \frac{k \cdot q}{k \cdot P}, \quad \phi_k, \tag{28}$$

where ϕ_k is the angle between the $(\mathbf{k}_1, \mathbf{k}_2)$ and (\mathbf{k}, \mathbf{q}) planes. This angle is given by

$$\sin(\phi_k) = \frac{2\varepsilon^{\mu\nu\rho\sigma} k_\mu P_\nu k_{1\rho} q_\sigma \sqrt{\lambda_Y}}{R \sqrt{\lambda_1 (Q^2 + \tau(S_x - \tau M_N^2))}}, \tag{29}$$

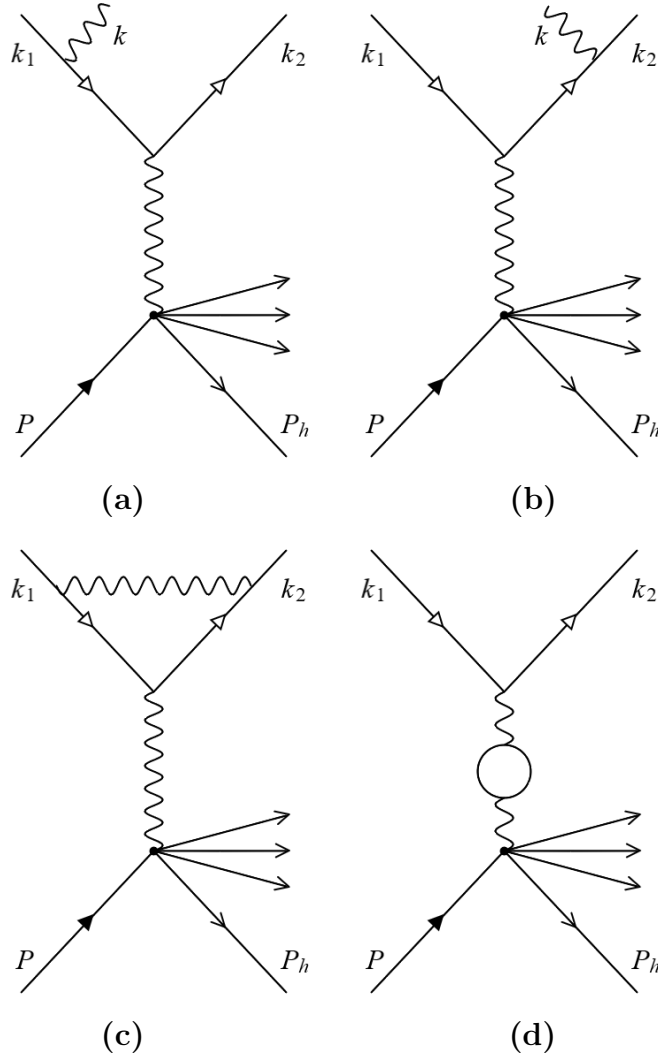


Figure 3: Feynman diagrams from (a) to (d) describing the NLO QED RC contributions to SIDIS scattering.

where

$$\lambda_1 = Q^2 (SX - M_N^2 Q^2) - m_l^2 \lambda_Y, \quad \text{with } X = 2P \cdot k_2, \quad (30)$$

and S , S_x , and λ_Y are defined in Eq. (20). The cross-section differential obtained from the real photon radiation off the leptonic leg is determined to

be

$$d\sigma_R = \frac{(4\pi\alpha)^3}{2\sqrt{\lambda_S} \tilde{Q}^4} \tilde{W}_{\mu\nu} L_R^{\mu\nu} d\Gamma_R. \quad (31)$$

In this case, the phase-space parameterization is the following:

$$d\Gamma_R = (2\pi)^4 \frac{d^3k}{(2\pi)^3 2k_0} \frac{d^3k_2}{(2\pi)^3 2k_{20}} \frac{d^3P_h}{(2\pi)^3 2P_{h0}}, \quad \text{with} \quad \frac{d^3k}{k_0} = \frac{R dR d\tau d\phi_k}{2\sqrt{\lambda_Y}}. \quad (32)$$

The ‘‘tilde’’ symbol of $\tilde{W}_{\mu\nu}$ in Eq. (31) means that the arguments of the hadronic tensor are defined via the shifted $q \rightarrow q - k$. With this shift, \tilde{Q}^2 is expressed by two photonic variables:

$$\tilde{Q}^2 = -(q - k)^2 = Q^2 + R\tau. \quad (33)$$

The leptonic tensor is separated into two parts:

$$L_R^{\mu\nu} = L_{R0}^{\mu\nu} + L_{R1}^{\mu\nu}. \quad (34)$$

The first term is given by

$$L_{R0}^{\mu\nu} = -\frac{1}{2} \text{Tr} \left[\left(\hat{k}_2 + m_l \right) \Gamma_R^{\mu\alpha} \left(\hat{k}_1 + m_l \right) \left(1 + \gamma_5 \hat{\xi}_0 \right) \bar{\Gamma}_{R\alpha}^\nu \right], \quad (35)$$

with

$$\begin{aligned} \Gamma_R^{\mu\alpha} &= \left(\frac{k_1^\alpha}{k \cdot k_1} - \frac{k_2^\alpha}{k \cdot k_2} \right) \gamma^\mu - \frac{\gamma^\mu \hat{k} \gamma^\alpha}{2k \cdot k_1} - \frac{\gamma^\alpha \hat{k} \gamma^\mu}{2k \cdot k_2}, \\ \bar{\Gamma}_{R\alpha}^\nu &= \gamma_0 \Gamma_{R\alpha}^{\nu\dagger} \gamma_0 = \left(\frac{k_{1\alpha}}{k \cdot k_1} - \frac{k_{2\alpha}}{k \cdot k_2} \right) \gamma^\nu - \frac{\gamma^\nu \hat{k} \gamma_\alpha}{2k \cdot k_2} - \frac{\gamma_\alpha \hat{k} \gamma^\nu}{2k \cdot k_1}. \end{aligned} \quad (36)$$

The second term is given by

$$L_{R1}^{\mu\nu} = -\frac{1}{2} \text{Tr} \left[\left(\hat{k}_2 + m_l \right) \Gamma_R^{\mu\alpha} \left(\hat{k}_1 + m_l \right) \gamma_5 \hat{\xi}_1 \bar{\Gamma}_{R\alpha}^\nu \right]. \quad (37)$$

ξ_0 and ξ_1 are the components of the polarization vector ξ shown in Eq. (21).

The convolutions of both separated leptonic tensors with the shifted hadronic tensor are given by

$$\begin{aligned} \tilde{W}_{\mu\nu} L_{R0}^{\mu\nu} &= -2 \sum_{i=1}^9 \sum_{j=1}^{k_i} \tilde{\mathcal{H}}_i \theta_{ij}^0 R^{j-3}, \\ \tilde{W}_{\mu\nu} L_{R1}^{\mu\nu} &= -2 \sum_{i=5,7,9} \sum_{j=1}^{k_i} \tilde{\mathcal{H}}_i \theta_{ij}^1 R^{j-3}, \end{aligned} \quad (38)$$

where the quantities θ_{ij}^0 and θ_{ij}^1 are not functions of R but are composite functions of the other kinematic variables, and are explicitly demonstrated in Appendix B of [30].

As was mentioned in the Introduction, for the extraction and cancellation of the infrared divergence in these RC calculations, the Bardin-Shumeiko method [31, 32] is applied. And here we represent the final formulas that we have implemented into SIDIS-RC EvGen. For convenience, let us designate the six-fold differential cross section in the form of

$$\sigma \equiv \frac{d\sigma}{dx_{Bj} dy dz_h dP_{hT}^2 d\phi_h d\phi_s}, \quad (39)$$

and use this notation henceforth. In this case, our master formula is the inelastic tail of the SIDIS total cross section (with several RC components included), expressed as

$$\sigma_{\text{SIDIS}}^{\text{in}} = \frac{\alpha}{\pi} (\delta_{VR} + \delta_{\text{vac}}^l + \delta_{\text{vac}}^h) \sigma_{\text{SIDIS}}^B + \sigma_R^F + \sigma^{\text{AMM}}, \quad (40)$$

where σ_{SIDIS}^B is shown in Eq. (24) or Eq. (3).

- δ_{VR} is the sum of the infrared divergent terms and is finite:

$$\begin{aligned} \delta_{VR} = & 2(Q_m^2 L_m - 1) \log\left(\frac{P_X^2 - M_{th}^2}{m_l \sqrt{P_X^2}}\right) + \frac{1}{2} S' L_{S'} + \\ & + \frac{1}{2} X' L_{X'} + S_\phi - 2 + \left(\frac{3}{2} Q^2 + 4m_l^2\right) L_m - \\ & - \frac{Q_m^2}{\sqrt{\lambda_m}} \left(\frac{1}{2} \lambda_m L_m^2 + 2\text{Li}_2\left[\frac{2\sqrt{\lambda_m}}{Q^2 + \sqrt{\lambda_m}}\right] - \frac{\pi^2}{2}\right), \end{aligned} \quad (41)$$

where $M_{th} = M_N + M_h$ is the minimum invariant mass value of the undetected hadron states P_X . Li_2 is Spence's dilogarithm; the variables Q_m^2 , P_x , S' , X' , and λ_m , are given by eq. (3); the functions L_m , $L_{S'}$, and $L_{X'}$, are given by eq. (C10); the function S_ϕ is given by eqs. (40)-(42) and (3); all in Ref. [30].

- δ_{vac}^l is the contribution of vacuum polarization by leptons:

$$\begin{aligned} \delta_{\text{vac}}^l &= \sum_{i=e,\mu,\tau} \delta_{\text{vac}}^{l,i} = \\ &= \sum_{i=e,\mu,\tau} \left(\frac{2}{3}(Q^2 + 2m_i^2)L_m^i - \frac{10}{9} + \frac{8m_i^2}{3Q^2}(1 - 2m_i^2 L_m^i)\right), \end{aligned} \quad (42)$$

where L_m^i is defined in eq. (D3) of Ref. [30].

• δ_{vac}^h is the contribution of vacuum polarization by hadrons, taken as a fit to hadronic cross section data from [66]:

$$\delta_{\text{vac}}^h = -\frac{2\pi}{\alpha} [A + B \log(1 + C|t_h|)], \quad (43)$$

where the constants A , B and C , are shown in Table 1.

$ t_h , (\text{GeV}/c)^2$	A	B	C
0 - 1	-1.345×10^{-9}	-2.302×10^{-3}	4.091
1 - 64	-1.512×10^{-3}	-2.822×10^{-3}	1.218
> 64	-1.344×10^{-3}	-3.068×10^{-3}	0.999

Table 1: The values of the three parameters in Eq. (43), shown in three consecutive ranges of $t_h = q^2$.

• σ_R^F is the infrared free contribution, integrated over the three photonic variables of Eq. (28):

$$\begin{aligned} \sigma_R^F = & -\frac{\alpha^3 S S_x^2}{64\pi^2 M_N P_{hL} \lambda_S \sqrt{\lambda_Y}} \int_{\tau_{\min}}^{\tau_{\max}} d\tau \int_0^{2\pi} d\phi_k \int_0^{R_{\max}} dR \times \\ & \times \sum_{i=1}^9 \left(\frac{\theta_{i1}}{R} \left(\frac{\tilde{\mathcal{H}}_i}{\tilde{Q}^4} - \frac{\mathcal{H}_i}{Q^4} \right) + \sum_{j=2}^{k_i} \tilde{\mathcal{H}}_i \theta_{ij} \frac{R^{j-2}}{\tilde{Q}^4} \right), \end{aligned} \quad (44)$$

where S , S_x , P_{hL} and λ_Y are shown in Eq. (20), λ_S in Eq. (18), \tilde{Q} in Eq. (33). Also, the composite functions θ_{i1} and θ_{ij} are explicitly shown in Appendix B, and \mathcal{H}_i is shown in eq. (14), all in Ref. [30]. The the integration limits are

$$R_{\max} = \frac{P_x^2 - M_{th}^2}{1 + \tau - \mu}, \quad \tau_{\max/\min} = \frac{S_x \pm \sqrt{\lambda_Y}}{2M_N^2}, \quad (45)$$

with μ defined as $(k \cdot P_h)/(k \cdot P)$ (see eq. (B3) of [30]).

• σ^{AMM} is the contribution of the anomalous magnetic moment:

$$\sigma^{AMM} = \frac{\alpha^3 m_l^2 S S_x^2}{16\pi M_N Q^2 P_{hL} \lambda_S} L_m \sum_{i=1}^9 \theta_i^{AMM} \mathcal{H}_i, \quad (46)$$

where the function L_m is given by eq. (C10), and θ_i^{AMM} is defined in eq. (54) of [30].

3. MC event generator SIDIS-RC EvGen for studying SIDIS processes with lowest-order radiative effects

This section describes the structure and functionality of the generator SIDIS-RC EvGen. The core of the generator is the `sidis` package, which is divided into a library component, called `libsidis`, and a generator (binary) component, called `sidisgen`. The library component `libsidis` is used to directly compute the SIDIS cross sections including NLO RCs, and is combined with a Monte-Carlo (MC) algorithm in the generator component `sidisgen` to produce random events. In particular, in the discussion of `libsidis`, we provide details on how cross-section calculations are fulfilled, along with describing two computational modes for doing it. In the discussion of `sidisgen`, we provide details on how the MC event generation is carried out by using a spatial partitioning method with hyper-cubical “foam of cells” based on the FOAM library [67] from ROOT [68].

3.1. Dividing the SIDIS inelastic cross section into soft and hard parts

For the purpose of efficiently generating events and calculating cross sections, we make use of the inelastic cross-section formula in Eq. (40) in a slightly modified form, which is obtained as follows.

According to the Bardin-Shumeiko approach [31, 32], the real photon radiation in Eq. (31) is split into infrared and infrared-free components:

$$d\sigma_R = d\sigma_R^{IR} + d\sigma_R^F, \quad (47)$$

where the infrared contribution is in turn separated into soft δ_S and hard δ_H photon emission parts:

$$\sigma_R^{IR} = \frac{\alpha}{\pi}(\delta_S + \delta_H)\sigma^B, \quad (48)$$

(see eq. (37) and appendix C of [30] for their explicit derivations). Ultimately, they are expressed as

$$\begin{aligned} \delta_S &= 2(Q_m^2 L_m - 1) \left(P_{IR} + \log\left(\frac{2\bar{k}_0}{\nu}\right) \right) + \frac{1}{2} S' L_{S'} + \frac{1}{2} X' L_{X'} + S_\phi, \\ \delta_H &= 2(Q_m^2 L_m - 1) \log\left(\frac{P_X^2 - M_{th}^2}{2\bar{k}_0 \sqrt{P_x^2}}\right). \end{aligned} \quad (49)$$

Let us again state that the variables Q_m^2 , P_X , S' , and X' , are given by eq. (3); the functions L_m , $L_{S'}$, and $L_{X'}$, are given by eq. (C10); the function S_ϕ is

given by eqs. (40)-(42) and (3); all shown in [30]. \bar{k}_0 is a small photon energy (soft cutoff) defined in the system $\mathbf{P} + \mathbf{q} - \mathbf{P}_h = 0$ that divides the soft and hard parts. The δ_S and δ_H contributions are \bar{k}_0 -dependent. However, their sum does not depend on \bar{k}_0 but includes the infrared divergence

$$P_{IR} = \frac{1}{n-4} + \frac{1}{2} \gamma_E + \log\left(\frac{1}{2\sqrt{\pi}}\right), \quad (50)$$

plus the arbitrary mass scale of dimensional regularization ν . Using an additional regularization with the parameter \bar{k}_0 , one can calculate δ_H at $n = 4$. The P_{IR} and ν contributions to $\delta_S + \delta_H$ are canceled by a contribution from the leptonic vertex correction given by

$$\begin{aligned} \delta_{vert} = & -2 (Q_m^2 L_m - 1) \left(P_{IR} + \log\left(\frac{m_l}{\nu}\right) \right) + \left(\frac{3}{2} Q^2 + 4m_l^2 \right) L_m - \\ & - 2 - \frac{Q_m^2}{\sqrt{\lambda_m}} \left(\frac{1}{2} \lambda_m L_m^2 + 2\text{Li}_2\left[\frac{2\sqrt{\lambda_m}}{Q^2 + \sqrt{\lambda_m}}\right] - \frac{\pi^2}{2} \right). \end{aligned} \quad (51)$$

Thereby, the sum of the three terms, $\delta_S + \delta_H + \delta_{vert}$, is free from the infrared-divergent term P_{IR} and the arbitrary mass scale ν . This sum is δ_{VR} in Eq. (41).

We can rewrite the cross-section formula from Eq. (40) as

$$\sigma_{\text{SIDIS}}^{in} = \frac{\alpha}{\pi} (\delta_{vert} + \delta_{vac}^l + \delta_{vac}^h) \sigma_{\text{SIDIS}}^B + \sigma^{AMM} + \int_0^\infty \bar{\sigma}_R d^3\mathbf{k}, \quad (52)$$

by using the notation $\sigma_R \equiv \int_0^\infty \bar{\sigma}_R d^3\mathbf{k}$ for the radiative part. For this integral we have the following terms:

$$\begin{aligned} \int_0^\infty \bar{\sigma}_R d^3\mathbf{k} &= \int_0^{\bar{k}_0} \bar{\sigma}_R^{IR} d^3\mathbf{k} + \int_0^{\bar{k}_0} \bar{\sigma}_R^F d^3\mathbf{k} + \int_{\bar{k}_0}^\infty \bar{\sigma}_R d^3\mathbf{k} \\ &= \frac{\alpha}{\pi} \delta_S \sigma^B + \int_0^{\bar{k}_0} \bar{\sigma}_R^F d^3\mathbf{k} + \int_{\bar{k}_0}^\infty \bar{\sigma}_R d^3\mathbf{k}. \end{aligned} \quad (53)$$

We can also regroup the terms in Eq. (52) to arrive at the following form:

$$\begin{aligned} \sigma_{\text{SIDIS}}^{in} &= \left[\frac{\alpha}{\pi} (\delta_{VS} + \delta_{vac}^l + \delta_{vac}^h) \sigma_{\text{SIDIS}}^B + \sigma^{AMM} + \int_0^{\bar{k}_0} \bar{\sigma}_R^F d^3\mathbf{k} \right]_{\text{non-rad. part}} + \\ &+ \left[\int_{\bar{k}_0}^\infty \bar{\sigma}_R d^3\mathbf{k} \right]_{\text{rad. part}} \equiv \\ &\equiv \sigma_{\text{SIDIS}}^{nrad} + \sigma_{\text{SIDIS}}^{rad}, \end{aligned} \quad (54)$$

with $\delta_{VS} = \delta_S + \delta_{vert}$.

In SIDIS-RC EvGen, the events are randomly selected to be non-radiative or radiative, based upon the total cross sections shown in Eq. (54). The radiative cross section is a nine-fold differential cross section (six SIDIS degrees of freedom + three photon degrees of freedom). The non-radiative cross section is a six-fold differential cross section, but has a computationally expensive integral over $\bar{\sigma}_R^F$ included. However, since $\bar{\sigma}_R^F$ is finite even to low photon energies, this integral can be neglected so long as the cutoff \bar{k}_0 is chosen small enough. Neglecting this integral introduces a weak cutoff dependence into the non-radiative cross section.

In other words, since the generator neglects the infrared-divergent-free part of the cross-section, the resulting cross sections unphysically depend on the soft photon cutoff \bar{k}_0 . At $\bar{k}_0 \rightarrow 0$, this approximation approaches the true cross section. However, by selecting too small a value for \bar{k}_0 will substantially reduce the generator's performance, as a greater fraction of the events will need to be drawn from the inefficient radiative part of the cross section. To choose a good value for \bar{k}_0 , a program at

https://github.com/duanebyer/sidis/blob/master/example/plot_cutoff.cpp has been provided, which demonstrates the user how the error in the cross-section changes with \bar{k}_0 . Fig. 4 shows what the output of that program looks like. Then \bar{k}_0 can be selected to be as large as possible while meeting whatever error requirements users have.

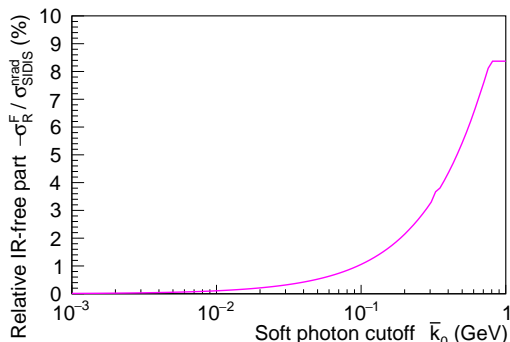


Figure 4: (Color online) Numerical estimate of the percentage of the total cross section $\sigma_{\text{SIDIS}}^{\text{nrad}}$ that stems from the non-radiative infrared-divergent-free part σ_R^F (taken with the minus sign), as a function of the soft photon cutoff \bar{k}_0 . As \bar{k}_0 goes to zero, σ_R^F contributes less and less to the total cross section. Selected kinematics: $E_b = 10.6$ GeV, $x_{Bj} = 0.2$, $Q^2 = 1.5$ GeV², $z_h = 0.35$, $P_{hT} = 0.3$ GeV, $\phi_h = 0$.

The structure functions can be decomposed further using the beam polarization ξ and target polarization $\boldsymbol{\eta}$ in the hadron frame (see Eq. (16)). Both σ_{SIDIS}^B and σ^{AMM} have the following form

$$\begin{aligned}\sigma &= \sigma^{UU} + \boldsymbol{\eta} \cdot \boldsymbol{\sigma}^{UP} + \lambda (\sigma^{LU} + \boldsymbol{\eta} \cdot \boldsymbol{\sigma}^{LP}), \\ \boldsymbol{\sigma}^{UP} &= \sigma^{UT_1} \mathbf{i} + \sigma^{UT_2} \mathbf{j} + \sigma^{UL} \mathbf{k}, \\ \boldsymbol{\sigma}^{LP} &= \sigma^{LT_1} \mathbf{i} + \sigma^{LT_2} \mathbf{j} + \sigma^{LL} \mathbf{k}.\end{aligned}\tag{55}$$

The eight polarized parts of the cross section σ^{UU} , σ^{UL} , etc., all depend on independent subsets of the SIDIS structure functions, so that they can be computed separately and then combined. For the radiative part (σ_R) of the cross section, the kinematic shift from the emitted photon causes the polarized parts σ^{UL} , σ^{UT_1} , and σ^{UT_2} to mix with each other (the same is also the case for σ^{LL} , σ^{LT_1} , and σ^{LT_2}). That means that there are only four independent polarized parts of the radiative cross section: σ_R^{UU} , $\boldsymbol{\sigma}_R^{UP}$, σ_R^{LU} , and $\boldsymbol{\sigma}_R^{LP}$.

3.2. Library component

<https://github.com/duanebyer/sidis/tree/master/src>
<https://github.com/duanebyer/sidis/tree/master/include/sidis>

The C++ library `libsidis` provides a set of functions for computing the six-fold Born cross section (σ_{SIDIS}^B), the six-fold non-radiative cross section ($\sigma_{\text{SIDIS}}^{nrad}$), and the nine-fold radiative cross section ($\sigma_{\text{SIDIS}}^{rad}$). Additionally, there are auxiliary functions for computing related cross sections/corrections (e.g., σ^{AMM} or δ^{VS}), kinematic variables, and structure functions.

The top-level `libsidis` namespace is `sidis`. It is further organized into several nested namespaces listed as follows:

- `xs`: functions for various cross sections;
- `kin`: kinematic calculations for the base SIDIS process (6D phase space) and for the radiative SIDIS processes (9D phase space)⁷;
- `lep`: “leptonic coefficients”, like θ_i^B , θ_{i1} , θ_{ij} and θ_i^{AMM} , employed for various cross sections (see Eqs. (24) (44) and (46));

⁷Had exclusive structure functions [34] been included in our framework, the radiative exclusive tail for SIDIS would be in a 8D phase space [30].

- **had**: “hadronic coefficients”, like \mathcal{H} and $\tilde{\mathcal{H}}$ for various cross sections (see again Eqs. (24) (44) and (46));
- **sf**: parameterizations of the SIDIS structure functions, including TMD PDFs and TMD FFs (see Sec. 2.1.2));
- **ps**: particle information, such as masses and charges [69];
- **math**: additional convenience functions.

Some computations may be duplicated between one cross-section evaluation and the next one, e.g., structure function evaluations when integrating over ϕ_h and/or ϕ_S . In order to reduce unnecessary computations, **libsidis** provides two main modes of use: a streamlined mode that hides the internal data flow, and a flexible mode that provides access to it. The streamlined mode should be used in nearly all cases, but the flexible mode API is exported for some occasional situations where it may provide a performance advantage.

3.2.1. Flexible mode

We start with the flexible mode of **libsidis**. The dataflow of this mode is diagrammatically shown in Fig. 5, and demonstrated in the program Listing 1 in Appendix A. The structure of this diagram is represented as follows:

- the particles involved and the six SIDIS phase-space variables are specified using the data structures `part::Particles` and `kin::PhaseSpace`;
- kinematic variables describing the SIDIS process are computed and cached in the `kin::Kinematics` data structure;
- leptonic and hadronic parts of the cross section are computed and cached using the data structures `lep::LepXX` and `had::HadXX`; these data structures come in several varieties, both for different polarizations (e.g., `had::HadUU`, `had::HadUL`, etc.), and for different types of cross sections (e.g., `lep::LepBornXX`, `lep::LepAmmXX`, etc.); the hadronic part of the cross section uses the SIDIS structure functions, as provided by the `sf::SfSet` interface;
- additional kinematic variables are computed for a specific type of cross section; for example, the `xs::Born` data structure is used for the Born cross section, or the `xs::Amm` data structure is used for the anomalous magnetic moment cross section, etc.;

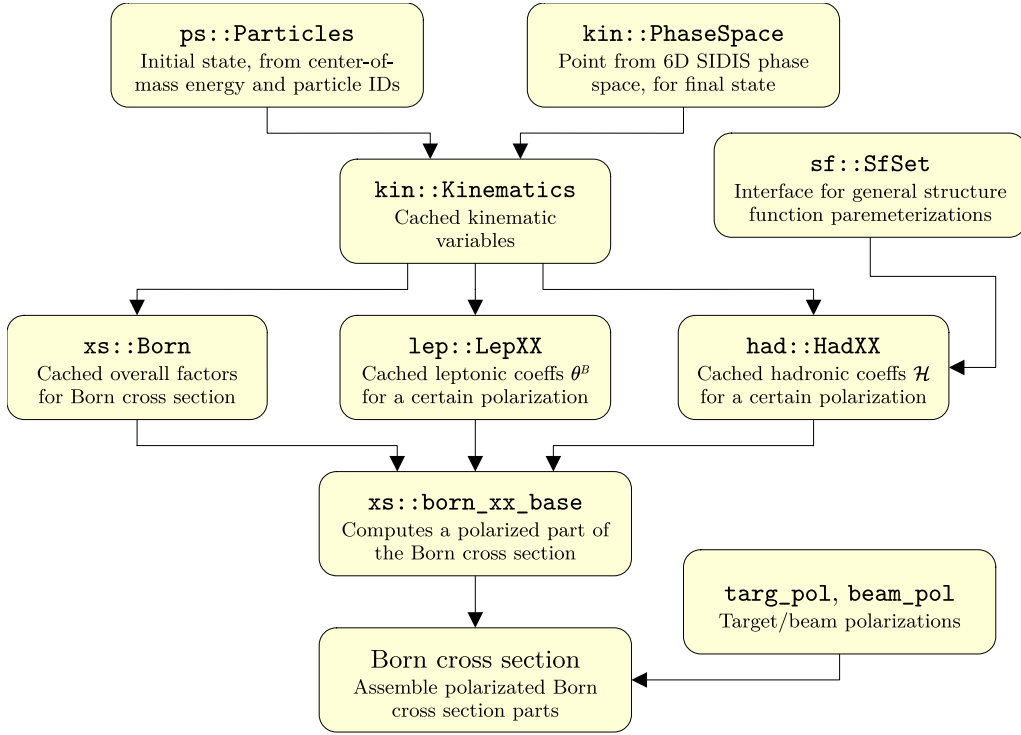


Figure 5: (Color online) Dataflow diagram for the flexible mode of `libsidis`. All type names are given in the `sidis::namespace`.

- both leptonic and hadronic coefficients are combined with those additional kinematic variables to compute one of the polarized parts of the cross section (as shown in Eq. (55)) using the `xs::born_xx_base` functions; these functions have variations for different polarizations and types of cross sections (e.g., `amm_uu_base`, `born_ll_base`, etc.).

The dataflow is the same for radiative cross sections, except that the kinematics-related data structures are substituted by their radiative versions:

$$\begin{aligned} \text{kin::PhaseSpace} &\rightarrow \text{kin::PhaseSpaceRad}; \\ \text{kin::Kinematics} &\rightarrow \text{kin::KinematicsRad}. \end{aligned}$$

The full polarized cross section must be constructed from the polarized parts of the cross section exactly, as described in Eq. (55).

3.2.2. Streamlined mode

The dataflow diagram for the streamlined mode is shown in Fig. 6, also demonstrated in the program Listing 1 in Appendix A. The streamlined

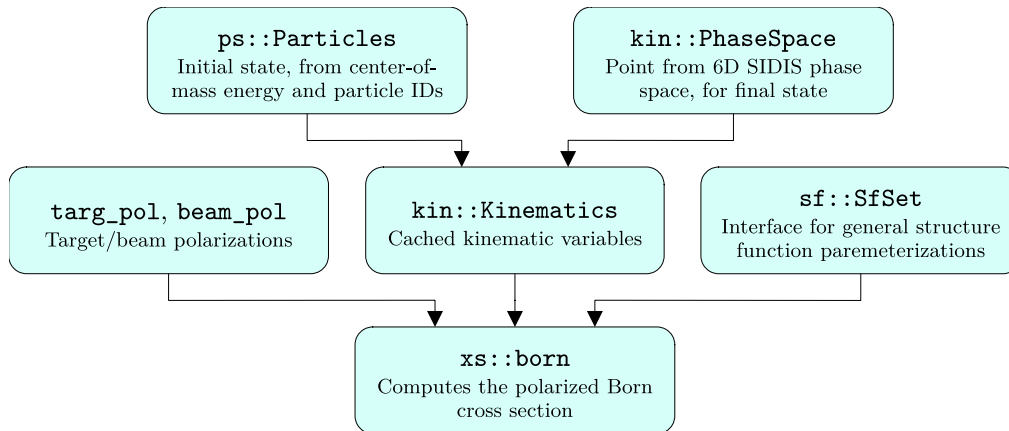


Figure 6: (Color online) Dataflow diagram for the streamlined mode of libsidis. All type names are given in the `sidis::namespace`.

mode makes use of functions with a simplified interface, such as `xs::born`, `xs::nrad`, or `xs::rad`. These streamlined functions compute the cross sections directly from the `kin::Kinematics` data structure and the `sf::SfSet` structure function set. The streamlined functions will automatically combine the polarized parts to form the full polarized cross section, as described in Eq. (55).

3.2.3. Structure function implementation

https://github.com/duanebyer/sidis/tree/master/share/sidis/sf_set/prokudin

In this section, we continue the discussion of Sec. 2.1.2 but in the context of the event generator. The interface `sf::SfSet` includes all eighteen leading- and subleading-twist SIDIS structure functions. These structure functions are often computed using the TMD factorization, which is valid at $P_{hT} \sim k_{\perp} \ll Q$, as mentioned in Sec. 2.1.1. For this situation, the library libsidis provides an implementation `sf::TmdSfSet`. The user-provided TMDs and FFs must then be given via another interface `sf::TmdSet`.

A 2D convolution needs to be performed when calculating the structure functions from these TMDs and FFs, based on which the type `sf::SfSet`

can compute the structure functions directly from the integral in Eq. (11). In order to simplify this convolution and further improve the generator’s performance, the TMDs and FFs are often taken to be Gaussian as in Eq. (12), such that the convolution can be evaluated analytically, which leads, e.g., to Eq. (13), (14), or (15). If the Gaussian approximation is turned on, then the user needs to provide only the 2D TMDs and FFs computed on (x_{Bj}, Q^2) grids [52, 60–63], with \mathbf{k}_\perp - and \mathbf{p}_\perp -dependence coming from the Gaussian width parameters. For this situation, the implementation `sf::GaussianTmdSfSet` can be used to provide the structure functions, combined with the Gaussian TMDs and FFs provided through another interface `sf::GaussianTmdSet`.

We already mentioned in Sec. 2.1.2 that section 3.2 of [10] has a detailed discussion of the WW-type approximations for TMDs/FFs, along with sections 4.1 and 4.2 of the same reference showing how to treat the twist-2 and all the twist-3 structure functions within these approximations. To twist-3, there are twenty-four TMDs and six FFs that must be provided. The twist-3 TMDs and FFs can be estimated in terms of the twist-2 TMDs and FFs, which reduce the independent TMDs to six and independent FFs to two. To use these approximations, the implementation `sf::WwTmdSfSet` can be used, combined with the WW-type basis TMDs and FFs given through another interface `sf::WwTmdSet`. Thereby, SIDIS-RC EvGen can apply WW-type approximations to reduce how many TMDs and FFs need to be provided by the user to improve performance.

While several WW-type approximations are compatible with the Gaussian approximation, the others are not. It happens because these latter WW-type approximations are not valid at $\mathbf{k}_\perp \rightarrow 0$ (see section 4.4 of [10] for more details). In this case, it is possible to combine the two types of approximations by first solving the convolution integrals (using the Gaussian approximation), and then applying the integrated WW-type approximations to simplify the obtained results for the structure functions. One can combine them using the implementation `sf::GaussianWwTmdSfSet` and the associated interface `sf::GaussianWwTmdSet`. Thus, in summary we have the following implementations that can be used for parameterizing the SIDIS structure functions provided by

- (i) a user-defined implementation of `sf::SfSet`;
- (ii) `sf::TmdSfSet`, with the TMDs/FFs specified as functions or on grids through the `sf::TmdSet` interface, and then numerically convoluted;

- (iii) `sf::GaussianTmdSfSet`, with the TMDs/FFs specified as Gaussian approximations through the `sf::GaussianTmdSet` interface, and then analytically convoluted;
- (iv) `sf::WwTmdSfSet`, with the WW basis TMDs/FFs specified through the `sf::WwTmdSet` interface (and unspecified TMDs/FFs computed through WW-type approximation) and then numerically convoluted;
- (v) `sf::GaussianWwTmdSfSet`, with the WW basis TMDs/FFs specified as Gaussian approximations through the `sf::GaussianWwTmdSet` interface, and then being analytically convoluted.

3.2.4. Leptonic/Hadronic coefficients

As mentioned in the discussion of the flexible mode in Sec. 3.2.1, the coefficients θ_i and \mathcal{H}_i , referred to as the leptonic coefficients and hadronic coefficients respectively, are packaged into two families of data structures `lep::LepXX` and `had::HadXX`. These data structures come in many varieties, both for different types of cross sections (Born, AMM, radiative), and different polarizations. The reason to separate these coefficients by polarization is to reduce the computational cost when only certain polarization parts of a cross section are used. For example, when the beam or target is unpolarized, the polarized parts of the cross section should not be computed at all. Or, when computing a transverse single spin asymmetry, only the σ^{UT} part of the cross section may be needed.

Here, we provide a taxonomy of the leptonic/hadronic coefficient structures. The polarizations are notated with two symbols XX . The first symbol denotes the lepton polarization, which can be U for unpolarized, L for longitudinally polarized, or X for arbitrary polarization (lepton transverse polarization is not allowed). The second symbol denotes the hadron polarization, which can be U for unpolarized, L for longitudinally polarized, T for transversely polarized, P for fully polarized in an arbitrary direction, or X for arbitrary polarization.

`had::HadXX` (lepton pol.: U, L, X ; hadron pol.: U, L, T, P, X)
 Standard set of structure functions \mathcal{H} , as explicitly shown in Eq. (14) of [30], used for all non-radiative cross sections.

`had::HadRadXX` (lepton pol.: U, L, X ; hadron pol.: U, P, X)
 Shifted structure functions $\tilde{\mathcal{H}}$, corresponding to the kinematic shift

introduced by real photon emission. The hadron polarization cannot be L or T alone, because $\sigma^{(U/L)L}$, $\sigma^{(U/L)T_1}$, and $\sigma^{(U/L)T_2}$ parts mix together in the shifted hadronic frame. T_2 refers to the part of the transverse polarization perpendicular to both \mathbf{q} and \mathbf{P}_{hT} in the target rest frame, while T_1 is the remaining part of the transverse polarization with a component along \mathbf{P}_{hT} . Used for the radiative cross section.

had::HadRadFXX (lepton pol.: U, L, X ; hadron pol.: U, P, X)

Subtracted shifted structure functions, used in the calculation of the infrared-divergent-free cross section σ_R^F (see Eq. (44)). This data structure stores both the shifted structure functions $\tilde{\mathcal{H}}$, and the difference $(\mathcal{H} - \tilde{\mathcal{H}})/R$, using the derivative to approximate when R is small.

The leptonic coefficients use a different set of hadron polarization symbols. The same subset of coefficients θ_i is used for “out-of-plane” hadron polarizations U and T_2 , and another subset is used for “in-plane” hadron polarizations L and T_1 . So, for the lepton coefficients only, the hadron polarization symbol U refers to both U and T_2 polarizations, while the hadron polarization symbol P refers to the others L and T_1 . The lepton polarization symbols remain unchanged.

lep::LepBornXX (lepton pol.: U, L, X ; hadron pol.: U, P, X)

Leptonic coefficients θ^B , as shown in Eq. (16) of [30], used for the Born cross section σ_{SIDIS}^B .

lep::LepAmmXX (lepton pol.: U, L, X ; hadron pol.: U, P, X)

Leptonic coefficients θ^{AMM} , as shown in Eq. (54) of [30], used for the anomalous magnetic moment (AMM) cross section σ^{AMM} .

lep::LepNradXX (lepton pol.: U, L, X ; hadron pol.: U, P, X)

Combination of `lep::LepBornXX` and `lep::LepAmmXX` coefficients that, combined with various correction factors, give the complete non-radiative cross section.

lep::LepRadXX (lepton pol.: U, L, X ; hadron pol.: U, P, X)

Leptonic coefficients θ , as shown in Eq. (B1) and Eq.(B2) of [30], used for the radiative part of the cross section.

Because of a large number of coefficient structures and the polarized variants, they are produced automatically using the `cog` code-generation tool [70].

3.3. Binary (generator) component

<https://github.com/duanebyer/sidis/tree/master/app/sidisgen>

The generator component, `sidisgen`, uses MC generation to produce points (these are the “events”) in the SIDIS phase space for numerical integration weighted by the SIDIS cross section $\sigma_{\text{SIDIS}}^{\text{in}}$ (from `libsidis`). The events are drawn from a probability distribution $\Gamma \sim P(\gamma)$ ⁸ and weighted by $w(\gamma) = \sigma_{\text{SIDIS}}^{\text{in}}(\gamma)/P(\gamma)$ (we define $W \equiv w(\Gamma)$). Then, the expectation value can be used to evaluate cross-section-weighted integrals over the SIDIS phase space:

$$\langle OW \rangle = \int O(\gamma) \sigma_{\text{SIDIS}}^{\text{in}}(\gamma) d\gamma. \quad (56)$$

The MC generator used by `sidisgen` attempts to minimize the variance in the weights W . If all of the weights are equal, then $P(\gamma) \propto \sigma_{\text{SIDIS}}^{\text{in}}(\gamma)$, so that the events produced by `sidisgen` can be used to mock data measured by SIDIS experiments at HERMES, COMPASS, and JLab.

The quality of the MC generator can be described by the ratio of the effective sample size to the true sample size [71, 72]. The square root of this quantity is labeled by ε , which we call the efficiency of the generator:

$$\varepsilon = \sqrt{\frac{N_{\text{eff}}}{N_{\text{tot}}}} = \sqrt{\frac{\langle W \rangle^2}{\langle W^2 \rangle}} = \sqrt{\frac{1}{1 + \sigma_W^2 / \langle W \rangle^2}}. \quad (57)$$

The efficiency varies in the range of $[0, 1]$. An efficiency of 1 corresponds to completely uniformly weighted events. The efficiency is directly related to the variance in the ratio $\sigma_{\text{SIDIS}}^{\text{in}}(\gamma)/P(\gamma)$. For defining $\sqrt{N_{\text{eff}}/N_{\text{tot}}}$, we take inspiration from [72], which can be motivated as the ratio of standard deviations of an observable \mathcal{O} between the importance distribution $P(\gamma)$ and the true cross section $\sigma_{\text{SIDIS}}^{\text{in}}(\gamma)$. The better $P(\gamma)$ approximates the cross section, the better the efficiency will be. The notation σ_W in Eq. (57) denotes the standard deviation (not to be confused with the cross-section notation).

Because non-radiative and radiative weighted events reside in different phase spaces (6D and 9D, respectively), `sidisgen` has a distinct MC sub-generator for each event type. Each time a new event is requested, it is

⁸The variable γ describes a point in the phase space.

selected at random to be either non-radiative or radiative. The ratio of non-radiative to radiative events is chosen to maximize the overall efficiency of the generator, corresponding to a relative probability of drawing events from each sub-generator of

$$P_{(n)rad} \propto \sqrt{\langle W_{(n)rad}^2 \rangle}. \quad (58)$$

3.3.1. MC generation with the FOAM library

The FOAM library from ROOT is used by `sidisgen` as the underlying MC engine for both non-radiative and radiative sub-generators, and approximates the (true) cross sections with a spatial indexing tree, called a foam. Under default settings, the foam is a k -d tree⁹, constructed through a recursive process in which the cross section is sampled randomly within each foam cell, and then the cells are divided to minimize the variance within each daughter cell. This process makes a nested tree structure of hyper-rectangles.

The foam is initialized once, and then used many times to produce events. For the event production, the FOAM library chooses a cell at random, with a probability chosen such that the overall efficiency is maximized:

$$P_k \propto V_k \sqrt{\int_{V_k} (\sigma_{\text{SIDIS}}^{in}(\gamma))^2 d\gamma}. \quad (59)$$

Then, a point γ_i is chosen randomly within the cell, and a weight $w_i = w(\gamma_i)$ is assigned to that point. As long as the foam is sufficiently dense such that the cross section does not vary significantly over each cell, the events will be weighted nearly uniformly.

Some kinematic cuts can be provided on the six-fold cross-section variables, x_{Bj} , y , z_h , P_{hT} , ϕ_h , ϕ_s . These cuts are accounted for during the FOAM tree initialization, and the library ensures that out-of-bound events are not produced outside of the cuts (meaning that no need to generate and then discard those events). The radiative contribution $\sigma_{\text{SIDIS}}^{rad}$ to the SIDIS cross section needs to be integrated over photon degrees of freedom to be compared to the non-radiative contribution $\sigma_{\text{SIDIS}}^{nrad}$. For ease of use, `libsidis` provides photon-integrated cross-section functions `xs::rad_integ` and `xs::rad_f_integ` that utilizes the VEGAS Monte-Carlo integrator [73]

⁹A k -d tree subdivides a rectangular k -dimensional region through recursive axis-aligned bisections.

from the GNU Scientific Library (GSL) [74]. The Cubature package [75] for the multi-dimensional adaptive quadrature method is used as a backup integrator. The photon-integrated cross sections make use of the variable transforms discussed in the next Section 3.3.2.

3.3.2. Improvement of the foam efficiency

As more cells are added to the foam, the efficiency improves. In order to describe this process, another useful quantity, $\beta = (1/\varepsilon) - 1$, is introduced, which ranges from 0 (being 100% efficient) to $+\infty$ (being 0% efficient). Empirically, β scales like a negative power law with increasing number of cells in the k -d tree: $\beta \propto N_{\text{cells}}^{-\alpha}$, as shown in Fig. 7. For most distributions, the power α is observed to vary between 0 and 2. If α is small, it takes far longer to reach the desired efficiency.

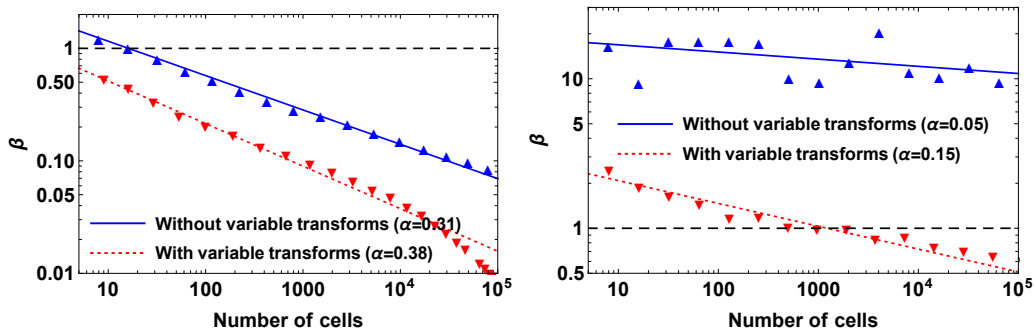


Figure 7: (Color online) The improvement of the foam efficiency as the number of cells in the foam increases, for the non-radiative (left plot) and radiative (right plot) cross sections. Smaller β corresponds to better efficiency, dashed line at 50% efficiency. The fluctuations of the points in the right plot are caused by a poor estimate of the foam’s efficiency at high β . The use of variable transforms (see later in Eq. 60) substantially improve the scaling parameter α in both cases.

When $\sigma_{\text{SIDIS}}^{\text{rad}}$ is expressed in terms of the photon variables (see Eq. (28)) R - proportional to photon energy, τ - related to photon polar angle, and ϕ_k - photon azimuthal angle, there are two radiative peaks in the (τ, ϕ_k) distribution, corresponding to photon emission collinear with the incoming and outgoing electrons. The locations of these peaks alter further with the other phase-space variables, leading to ridges in the full 9D distribution. The ridges are not well aligned with the hyper-rectangle axes, making it difficult for the foam to approximate them, and consequently resulting in a very poor scaling factor $\alpha_{\text{rad}} \sim 0.05$, corresponding to the upper triangle points and

the solid line in the right plot of Fig. 7. With such a poor scaling factor, it would take on the order of 10^{10} cells to reach an efficiency of only 75%. We mitigated this problem in several ways.

- First, the FOAM library is extended to improve performance, add support to its original multi-threading, and use more accurate estimators for the cell division. This makes it feasible to use an order more cells in the foam construction.

- Second, two variable transformations are used to smooth out the radiative peaks. The locations and widths of the peaks in (τ, ϕ_k) around the incoming and outgoing electron directions can be calculated from the electron kinematics. Then, a sigmoid transformation in ϕ_k and a double-sigmoid transformation in τ can be used to smooth the peaks, choosing the steepest parts of such transformations to coincide with the peak locations. We use

$$\begin{aligned}\phi'_k &= \operatorname{arcsinh}\left(\frac{\phi_k}{w_{\phi_k}}\right), \\ \tau' &= \operatorname{arcsinh}\left(\frac{\tau - \tau_{p,1}}{w_{\tau,1}}\right) + \operatorname{arcsinh}\left(\frac{\tau - \tau_{p,2}}{w_{\tau,2}}\right),\end{aligned}\tag{60}$$

where w_{ϕ_k} , $w_{\tau,1}$, $w_{\tau,2}$ are the peak widths, and $(0, \tau_{p,1})$, $(0, \tau_{p,2})$ are the peak locations. The result of these transformations is shown in Fig. 8. Additional variable transformations are applied to some of the other SIDIS variables as well, such as a $(\cdot)^{-1}$ transform on the Bjorken x_{Bj} , to roughly match the expected cross-section behavior. The transformations substantially improve the foam construction process, as shown in Fig. 7.

- Third, rejection sampling is used to reduce variance of events produced by the non-radiative and radiative sub-generators. The radiative sub-generator can often reach an efficiency of 75%, and the rejection sampling can push this number up further towards 95% or higher, at a cost to event generation performance.

4. Numerical results obtained from SIDIS-RC EvGen

By having at our disposal all the discussion in Sec. 3, in this section we demonstrate a number of results obtained after running SIDIS-RC EvGen: namely on the SIDIS Collins and Sivers azimuthal transverse SSAs. Fast computations of the asymmetries in general are provided in the `asym` namespace, combining the GSL VEGAS integrator with the flexible mode for efficient integration.

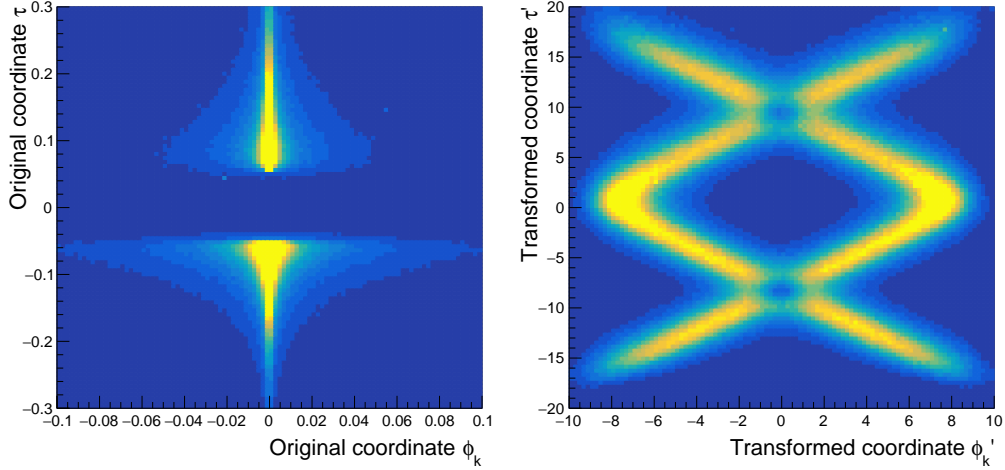


Figure 8: (Color online) The change in the radiative part of the distribution as a result of the variable transforms from Eq. 60. The left plot shows the two sharp peaks in the radiative distribution, which are difficult for the foam to approximate. The right plot shows much broader transformed peaks, allowing the foam to more accurately approximate it.

4.1. Charged-hadron Collins and Sivers azimuthal transverse single-spin asymmetries

Transverse SSAs are among the most important and fascinating observables in electron scattering and hadronic physics. During the last fifteen years or so, the transverse SSAs in SIDIS have been studied at HERMES, COMPASS, and JLab [19–21, 24–26, 28]. Their studies are crucial for our cardinal understanding of the 3D momentum structure of the nucleon.

The SIDIS differential cross section in Eq. (3) (or Eq. (24)), including all leading-twist (twist-2) and subleading-twist (twist-3) structure functions in the $1/Q$ expansion, can also be expressed in terms of SSAs in the following way [10]:

$$\begin{aligned} \sigma_{\text{SIDIS}}^B &= \frac{\alpha^2}{x_{Bj} y Q^2} \left(1 + \frac{\gamma^2}{2x_{Bj}} \right) c_1 F_{UU} \times \\ &\times \left\{ \left[1 + \left(\frac{c_3}{c_1} \right) \cos(\phi_h) A_{UU}^{\cos(\phi_h)} + \right. \right. \\ &\quad \left. \left. + \left(\frac{c_2}{c_1} \right) \cos(2\phi_h) A_{UU}^{\cos(2\phi_h)} + \lambda_e \left(\frac{c_4}{c_1} \right) \sin(\phi_h) A_{LU}^{\sin(\phi_h)} \right] + \right. \end{aligned}$$

$$\begin{aligned}
& +S_L \left[\left(\frac{c_3}{c_1} \right) \sin(\phi_h) A_{UL}^{\sin(\phi_h)} + \left(\frac{c_2}{c_1} \right) \sin(2\phi_h) A_{UL}^{\sin(2\phi_h)} \right] + \\
& +S_L \lambda_e \left[\left(\frac{c_5}{c_1} \right) A_{LL} + \left(\frac{c_4}{c_1} \right) \cos(\phi_h) A_{LL}^{\cos(\phi_h)} \right] + \\
& +S_T \left[\sin(\phi_h - \phi_s) A_{UT,T}^{\sin(\phi_h - \phi_s)} + \right. \\
& \quad + \left(\frac{c_2}{c_1} \right) \sin(\phi_h + \phi_s) A_{UT}^{\sin(\phi_h + \phi_s)} + \left(\frac{c_2}{c_1} \right) \sin(3\phi_h - \phi_s) A_{UT}^{\sin(3\phi_h - \phi_s)} + \\
& \quad \left. + \left(\frac{c_3}{c_1} \right) \sin(\phi_s) A_{UT}^{\sin(\phi_s)} + \left(\frac{c_3}{c_1} \right) \sin(2\phi_h - \phi_s) A_{UT}^{\sin(2\phi_h - \phi_s)} \right] + \\
& +S_T \lambda_e \left[\left(\frac{c_5}{c_1} \right) \cos(\phi_h - \phi_s) A_{LT}^{\cos(\phi_h - \phi_s)} + \left(\frac{c_4}{c_1} \right) \cos(\phi_s) A_{LT}^{\cos(\phi_s)} + \right. \\
& \quad \left. + \left(\frac{c_4}{c_1} \right) \cos(2\phi_h - \phi_s) A_{LT}^{\cos(2\phi_h - \phi_s)} \right] \Bigg\}, \tag{61}
\end{aligned}$$

with the asymmetries generally defined as various ratios of polarized and unpolarized cross sections:

$$A_{XY}^{\text{a.d.}} \equiv A_{XY}^{\text{a.d.}}(x_{Bj}, Q^2, z_h, P_{hT}) = \frac{F_{XY}^{\text{a.d.}}(x_{Bj}, Q^2, z_h, P_{hT})}{F_{UU}(x_{Bj}, Q^2, z_h, P_{hT})}, \tag{62}$$

where the superscript a.d. means angular dependence (azimuthal modulations), and the subscripts XY are explained in the paragraph below Eq. (3). Meanwhile, how these polarization indices are organized in the event generator framework (in somewhat different way) are shown in details in Sec. 3.2.4.

It is obvious from Eq. (61) that at the order of $\mathcal{O}(1/Q)$, there are totally five SSAs with an unpolarized beam and a transversely polarized target. Two of these SSAs, due to the Collins effect [76] and Sivers effect [77], are twist-2 observables: A_{UT}^{Collins} and A_{UT}^{Sivers} , respectively. The Collins effect emerges from

the convolution of the transversity TMD and the Collins FF, the Siverts effect stems from the convolution of the Siverts TMD and the unpolarized FF¹⁰.

The general form of the transverse non-separated SSA can be written with all three leading-twist (or twist-2) and two subleading-twist (or twist-3) terms:

$$A_{UT} = A_{UT}^{\text{Collins}} \sin(\phi_h + \phi_s) + A_{UT}^{\text{Siverts}} \sin(\phi_h - \phi_s) + A_{UT}^{\text{Pretzelosity}} \sin(3\phi_h - \phi_s) + A_{UT}^{\text{sl-t1}} \sin(\phi_s) + A_{UT}^{\text{sl-t2}} \sin(2\phi_h - \phi_s), \quad (63)$$

with the SSAs extracted from the Born cross section (see e.g. [9]) given by

$$\begin{aligned} A_{UT}^{\text{Collins}} &\equiv A_{UT}^{\sin(\phi_h + \phi_s)} \equiv 2\langle \sin(\phi_h + \phi_s) \rangle = \\ &= 2 \frac{\int_0^{2\pi} d\phi_s \int_0^{2\pi} d\phi_h \sin(\phi_h + \phi_s) \sigma^B}{\int_0^{2\pi} d\phi_s \int_0^{2\pi} d\phi_h \sigma^B} = \frac{c_2 F_{UT}^{\sin(\phi_h + \phi_s)}}{c_1 F_{UU}}, \end{aligned} \quad (64)$$

$$\begin{aligned} A_{UT}^{\text{Siverts}} &\equiv A_{UT}^{\sin(\phi_h - \phi_s)} \equiv 2\langle \sin(\phi_h - \phi_s) \rangle = \\ &= 2 \frac{\int_0^{2\pi} d\phi_s \int_0^{2\pi} d\phi_h \sin(\phi_h - \phi_s) \sigma^B}{\int_0^{2\pi} d\phi_s \int_0^{2\pi} d\phi_h \sigma^B} = \frac{1 F_{UT}^{\sin(\phi_h - \phi_s)}}{c_1 F_{UU}}, \end{aligned} \quad (65)$$

where the prefactors are given in Eqs. (4), (5), (2) and (9). $A_{UT}^{\text{Pretzelosity}}$ is defined similarly. For having more details on these SSAs, we refer to [10]: notably, its section 5.3 for A_{UT}^{Siverts} , section 5.4 for A_{UT}^{Collins} , section 7.6 for $A_{UT}^{\text{sl-t1}}$, and section 7.7 for $A_{UT}^{\text{sl-t2}}$. Meanwhile, F_{UU} , $F_{UT}^{\sin(\phi_h + \phi_s)}$, and $F_{UT}^{\sin(\phi_h - \phi_s)}$ are given in Eqs. (13), (14), and (15), respectively. For $F_{UT}^{\sin(\phi_s)}$ and $F_{UT}^{\sin(2\phi_h - \phi_s)}$, we refer back to our discussion at the end of Sec. 2.1.2.

When RCs are included, the asymmetries become modified through the substitution $\sigma^B \rightarrow \sigma_{\text{SIDIS}}^{\text{in}}$ in Eq. (64) and Eq. (65), using Eq. (54) for $\sigma_{\text{SIDIS}}^{\text{in}}$.

¹⁰An example showing the power of the TMD factorization (mentioned in Sec. 2.1.1), where the structure functions $F_{UT}^{\sin(\phi_h + \phi_s)}$ and $F_{UT}^{\sin(\phi_h - \phi_s)}$ ($\equiv F_{UT,T}^{\sin(\phi_h - \phi_s)}$) are expressed by those convolutions, is the agreement between model calculations and COMPASS/HERMES experimental results for the Siverts and Collins effects [57, 58].

However, because the radiated photon causes the hadron frame to shift, the RC-modified asymmetries do not have the simple relationship to the structure functions given by the right-hand-side of Eq. (64) and Eq. (65).

Figs. 9(a)-9(d) show the Collins SSA for positively charged pions as a function of x_{Bj} in given kinematic bins of Q^2 , z_h , and P_{hT} .

- The solid curves are obtained from direct calculations of A_{UT}^{Collins} at the bin center.
- The blue downward triangle marker shows the Collins SSA measured from pseudo-data generated by `sidisgen` using the Born cross section σ^B with structure functions at leading twist only.
- The green upward triangle marker shows the Collins SSA measured from pseudo-data using the Born cross section σ^B with structure functions at leading and subleading twists.
- The red circle marker shows the Collins SSA measured from pseudo-data using the full RC cross section $\sigma_{\text{SIDIS}}^{\text{in}}$ with structure functions at leading and subleading twists.

The pseudo-data errors are statistical uncertainties, for the treatment and propagation of which, we refer back to Sec. 3.2.1. Figs. 10(a)-10(d) show analogous plots made in the same bins but for the Sivers SSA. In Figs. 11(a)-11(d), one can see ratios describing the RC effects on both Collins and Sivers asymmetries, when $A_{UT}^{\text{Collins|Sivers}}|_{\text{RC}}$ are compared to $A_{UT}^{\text{Collins|Sivers}}$:

$$\text{RC Ratio} = \frac{A_{UT}^{\text{Collins|Sivers}}|_{\text{RC}}}{A_{UT}^{\text{Collins|Sivers}}}. \quad (66)$$

In these latter figures, it is apparent that the discussed observables can amount up to about 5% of the Born-level results. In the context of upcoming comparisons with experimental data and/or more precise predictions, it will be important that SIDIS-RC EvGen also includes the missing higher-order RC effects in the future. One way as a first approximation could be employing the so-called (simplest) exponentiation procedure that had been used in [78] for calculations of RCs beyond the ultra-relativistic limit in unpolarized elastic $e + p$ scattering.

The exponentiation procedure is based on simply accounting for multiple soft photons and respective loops for canceling the infrared divergence. In our

case, we need to consider $\sigma_{\text{SIDIS}}^B + (\text{term proportional of } \delta_{VR})$ as the first two terms in expansion of the exponent $\exp[(\alpha/\pi)\delta_{VR}]\sigma_{\text{SIDIS}}^B$ in a series over α . Here, δ_{VR} is the sum of the infrared divergent terms in Eq. (41). This is not only because one can find a straightforward way to account for multiple soft-photon radiation but also it is a regularization of the lowest-order RCs, since δ_{VR} will be infinitely close to the pion threshold at $P_X^2 \rightarrow M_{th}^2$. The term δ_{VR} in Eq. (40) can be later substituted by $\delta_{VR} \rightarrow (\pi/\alpha) (\exp[(\alpha/\pi)\delta_{VR}] - 1)$ in SIDIS-RC EvGen. Note that Eq. (54) should be modified accordingly. Such a new implementation in the event generator, even given its simple nature, may increase the precision of the pseudo-data in all Figs. 9, 10, and 11.

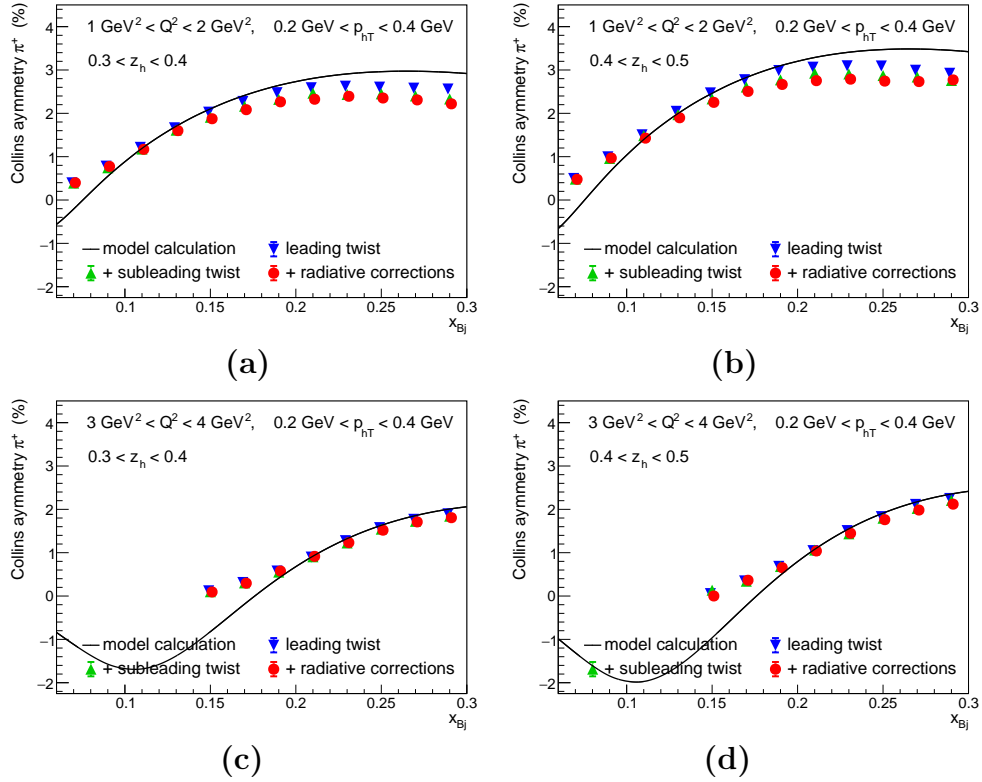


Figure 9: (Color online) The Collins SSA pseudo-data on π^+ as a function of x_{Bj} , obtained from SIDIS-RC EvGen in specific Q^2 , z_h , and P_{hT} bins that are shown in the plots. The model calculation is at the bin center while the pseudo-data is integrated over the entire bin, leading to some additional difference between them. For better visualization, the points have been slightly offset horizontally. For producing these pseudo-data sets, the electron beam energy of 10.6 GeV has been used (for the other kinematic parameters, see Listing 2 in Appendix B.)

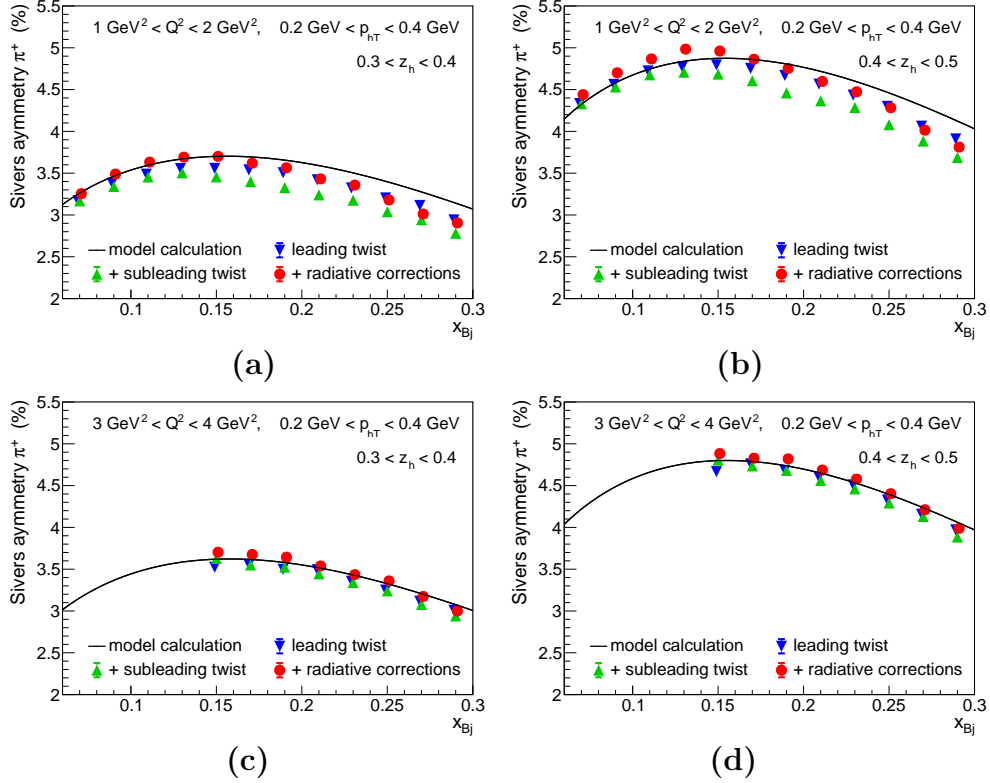


Figure 10: (Color online) The Siverts SSA pseudo-data on π^+ as a function of x_{Bj} , obtained from SIDIS-RC EvGen in the same Q^2 , z_h , and P_{hT} binning, as in Fig. 9. The model calculation is at the bin center while the pseudo-data is integrated over the entire bin, leading to some additional difference between them. For better visualization, the points have been slightly offset horizontally.

With the A_{UT}^{Collins} , A_{UT}^{Siverts} , $A_{UT}^{\text{Pretzelosity}}$ SSAs and $A_{UT}^{\text{sl-t1}}$, $A_{UT}^{\text{sl-t2}}$ terms, it is possible to address essential questions in modern nuclear physics such as (i) whether one can provide a high precision test of lattice QCD predictions via the tensor charge [79] (a fundamental quantity related to the nucleon spin); (ii) whether there are clear signatures of relativistic effects inside the nucleon, (iii) how one can extract quantitative information about a contribution of the quark orbital angular momentum to the proton spin; (iv) how to quantify the quark transverse motion inside the nucleon and observe spin-orbit correlations.

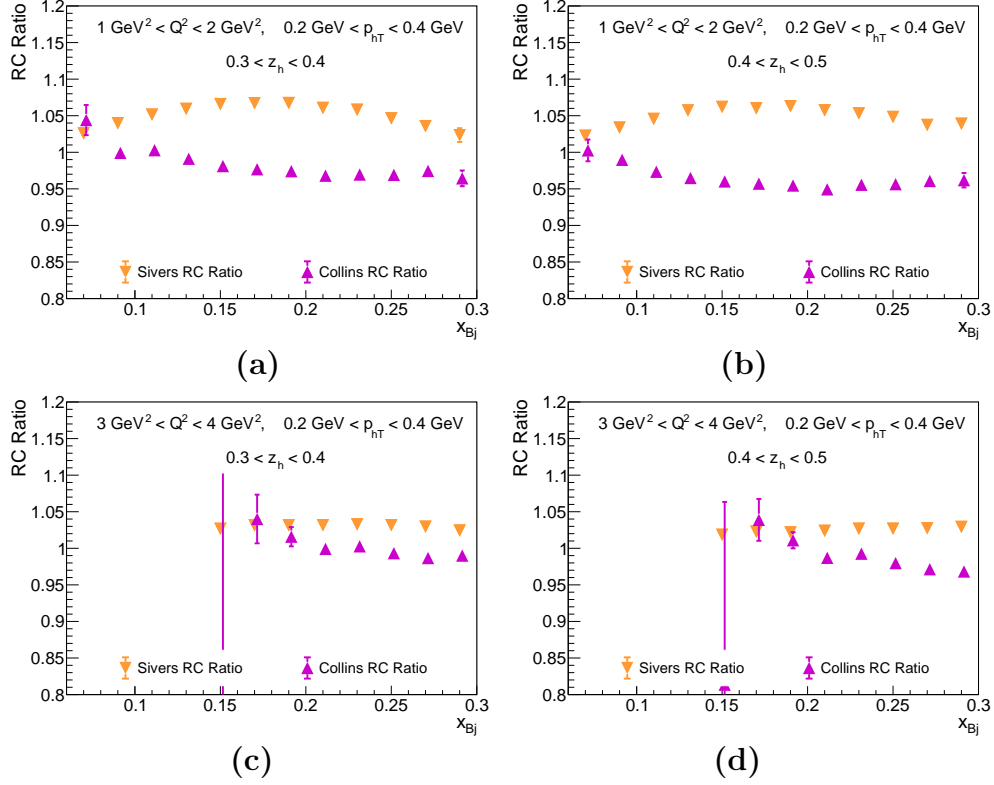


Figure 11: (Color online) Ratios of the red circle pseudo-data to the green upward triangle pseudo-data shown in Fig. 9 and Fig. 10. The ratios demonstrate the lowest-order RC effects on the Collins and Sivers SSAs when the subleading-twist effects are also taken into account. The large error bars of the Collins RC ratio in the lowest bin of $x_{Bj} \sim 0.15$ appear because the Collins asymmetry drops to near zero at this point. This figure shows one of the principal results we gain from SIDIS-RC EvGen.

4.2. Prospects on charged-hadron multiplicity and asymmetry data comparisons

In the partonic description, the azimuthal asymmetries in Eq. (61) are applicable for studying spin-independent and spin-dependent TMDs and FFs (see Eqs. (23) and (26)). In this regard, another relevant experimental observable to investigate TMDs and FFs is the differential hadron multiplicity [22, 27, 29], defined as

$$n_{\text{SIDIS}}^h(x_{Bj}, Q^2, z_h, P_{hT}^2) \equiv \left(\frac{d\sigma_{\text{SIDIS}}^B}{dx_{Bj} dQ^2 dz_h dP_{hT}^2} \right) / \left(\frac{d\sigma_{\text{DIS}}}{dx_{Bj} dQ^2} \right), \quad (67)$$

where the denominator is the DIS cross section.

Comparisons with experimental data on the multiplicity observable from Eq. (67) and on other asymmetries from Eq. (61), based on using pseudo-data from SIDIS-RC EvGen, will be topics for investigation in the future, after exclusive structure functions are also implemented in SIDIS-RC EvGen.

5. Summary and outlook

MC event generators are important constituents of a variety of experimental analyses. They are extensively used to make predictions and preparations for designed experiments. Physics models and frameworks used for underlying event generation and their reliability are essential for understanding of various systematic effects in studies of different observables. One source of such effects are induced by radiative corrections in semi-inclusive deep inelastic scattering.

In this paper, we presented a C++ coded standalone MC event generator called SIDIS-RC EvGen, for a purpose of studying the partonic structure of the nucleon in 3D momentum space, by using SIDIS that also includes the lowest-order RCs. We discussed and simultaneously provided various details on

- (1) the twist-2 and twist-3 SIDIS structure functions, which are thoroughly and systematically investigated in the Wandzura-Wilczek-type and Gaussian approximations [10, 52]¹¹ on the basis of available data;
- (2) the inelastic tail to the SIDIS six-fold differential cross section, including the lowest-order RC components, for which explicit exact analytical formulas (with a longitudinally polarized lepton and an arbitrarily polarized target) are computed in compact and covariant form convenient for numerical analyses [30];
- (3) our created event generator's structure and functionality focusing on its library component for computing cross sections and binary (generator) component for event generation [45], along with showing numerical results on the Collins and Sivers SSAs in the SIDIS process.

It is relevant to outline the following potential prospects related to our work with SIDIS-RC EvGen; such as

¹¹These references generally render very useful inputs for MC event generators.

- (i) a) increasing the efficiency of SIDIS-RC EvGen by improving the foam efficiency, or even replacing the FOAM algorithm by the VEGAS algorithm;
- b) providing with other options for including state-of-the-art parameterizations of TMDs, used in most recent and upcoming phenomenological studies; such as of Refs. [80–90].
- (ii) a) incorporating the exclusive structure functions into our current framework;
- b) including Q^2 evolution of the SIDIS structure functions as an optional feature;
- c) improving the parameterization that describes the contribution of vacuum polarization by hadrons (δ_{vac}^h), making use of the most recent hadronic data for fitting [91, 92], the advanced calculations from [93–95], and the software package `alphaQED` of F. Jegerlehner [96];
- d) implementing the exponentiation procedure for σ_{SIDIS}^n already discussed in Sec. 4.1;
- e) planning by some of us to calculate the second-order SIDIS RCs in a more robust way (as a follow-up to Ref. [30]), and subsequently incorporating the updated RC framework in the event generator.
- (iii) comparing the event generator’s output with the HERMES, COMPASS, and JLab various SSA data sets [19–21, 24–26, 28], with the charged-hadron multiplicity data sets [22, 27, 29], as well as making predictions for the future EIC [49, 97].
- (iv) developing eventually a versatile framework for nucleon’s 3D momentum structure extraction from our standalone generator: an example of how a more flexible MC generator can be built by extending existing generators is like the multi-purpose particle physics event generator Herwig++ 3.0, built on the basis of the ThePEG substructure that underlies the Herwig++ versions [98].

Acknowledgements

This work has been supported in part by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under contracts No. DE-FG02-

03ER41231 (D.B., V.K., H.G., and Z.Z.), No. DE-AC02-06CH11357 (C.P.), No. DE-AC05-06OR23177 (A.P.) under which Jefferson Science Associates, LLC, manages and operates Jefferson Lab; and within the framework of the TMD Topical Collaboration (A.P.), and by the National Science Foundation Grant No. PHY-2012002 (A.P.).

Appendix A. Program listings of the flexible/streamlined modes

Listing 1: Demonstration of differences between the libsidis flexible mode (Sec. 3.2.1) and streamlined mode (Sec. 3.2.2).

```

1  using namespace sidis;
2  // Initial state.
3  part::Particles ps(
4      part::Nucleus::P, part::Lepton::E, part::Hadron::PI_P,
5      MASS_P + MASS_PI_0);
6  double S = 2 * MASS_P * 10;
7  // Phase space.
8  kin::PhaseSpace ph_space { 0.2, 0.9, 0.3, 2., 0.5*PI, 0.2*PI };
9  kin::Kinematics kin(ps, S, ph_space);
10 // Structure function parameterization.
11 sf::set::ProkudinSfSet sf_set;
12 // Polarization vectors.
13 double beam_pol = 0.5;
14 math::Vec3 targ_pol(0.1, 0.4, 0.2);
15 if (flexible) {
16     // Born coefficients.
17     xs::Born b(kin);
18     // Leptonic coefficients.
19     lep::LepBornUU lep_uu(kin);
20     lep::LepBornUP lep_up(kin);
21     lep::LepBornLU lep_lu(kin);
22     lep::LepBornLP lep_lp(kin);
23     // Hadronic coefficients.
24     had::HadUU had_uu(kin, sf_set);
25     had::HadUL had_ul(kin, sf_set);
26     had::HadUT had_ut(kin, sf_set);
27     had::HadLU had_lu(kin, sf_set);
28     had::HadLL had_ll(kin, sf_set);
29     had::HadLT had_lt(kin, sf_set);
30     // Polarized parts of Born cross section.
31     double xs_uu = xs::born_uu_base(b, lep_uu, had_uu);
32     double xs_ul = xs::born_ul_base(b, lep_up, had_ul);
33     double xs_ut1 = xs::born_ut1_base(b, lep_up, had_ut);
34     double xs_ut2 = xs::born_ut2_base(b, lep_uu, had_ut);
35     double xs_lu = xs::born_lu_base(b, lep_lu, had_lu);
36     double xs_ll = xs::born_ll_base(b, lep_lp, had_ll);
37     double xs_lt1 = xs::born_lt1_base(b, lep_lp, had_lt);
38     double xs_lt2 = xs::born_lt2_base(b, lep_lu, had_lt);
39     math::Vec3 xs_up(xs_ut1, xs_ut2, xs_ul);
40     math::Vec3 xs_lp(xs_lt1, xs_lt2, xs_ll);
41     // Assemble to get complete Born cross section.
42     double xs = (xs_uu + math::dot(xs_up, targ_pol))

```

```

43         + beam_pol*(xs_lu + math::dot(xs_lp, targ_pol));
44     }
45     if (streamlined) {
46         // Compute polarized cross section.
47         double xs = xs::born(kin, sf_set, beam_pol, targ_pol);
48     }

```

Appendix B. Quick start and Build of SIDIS-RC EvGen

This appendix describes the procedure of the quick start of the event generator (for its building see <https://github.com/duanebyer/sidis>). But first of all, let us list the following libraries that this software does utilize.

- WW-SIDIS: structure functions for the proton using the WW-type and Gaussian approximations [10, 52];
- GSL: multi-dimensional Monte-Carlo integration [74], including the Monte-Carlo integrator VEGAS [73];
- FOAM: MC event generation using spatial partitioning [67];
- MSTWPDF: parton distribution functions for the proton [60, 61];
- Cog: code generation with Python [70];
- ROOT: plotting and data analysis [68];
- Cubature: a backup package for multi-dimensional numerical integration [75].

Quick start (generator): An example of the event generator’s `sidisgen` component’s user command-line interface is demonstrated in Listing 2, which describes an input file with all parameters needed for event generation. The allowed options can be seen by `sidisgen --help`. The generator is run in two steps using the input file. The FOAM library must be initialized before the events can be generated themselves. With this initialization, one approximates the SIDIS differential cross section in a specified kinematic region. Thus, we first call

```
sidisgen --initialize <parameter file> (B1)
```

Listing 2: An example parameter file for interfacing with the `sidisgen` generator.

```
1 mc.num_events      10000
2 # structure function parameterization
3 phys.sf_set        prokudin
4 # which method of radiative corrections to use
5 phys.rc_method     approx
6 # soft threshold dividing non-radiative and radiative cross sections
7 phys.soft_threshold 0.01
8 # kinematic threshold for SIDIS process
9 phys.mass_threshold 1.073249081
10 # initial conditions
11 setup.beam        e
12 setup.target       p
13 setup.hadron       pi+
14 setup.beam_energy  10.6
15 setup.beam_pol     0
16 setup.target_pol   0 0 0
17 # cuts on kinematic variables
18 cut.x              0.1 0.6
19 cut.Q_sq           2.0 3.5
20 cut.z              0.3 0.7
21 cut.W_sq           5.29 1e10
```

to create some saved FOAM tree that can be used in between runs. Then one should use the FOAM library for MC event generation, by calling

```
sidisgen --generate <parameter file> (B2)
```

to generate variable-weighted events¹². Afterwards, the resulting events are provided in a ROOT file, and it can be converted into other formats if required. Any produced event ROOT file will store the parameters used to produce it internally, which can be double-checked with `sidisgen --inspect` command.

The events will be written to a ROOT file with the following entries:

params: A subdirectory storing a record of the parameters used to produce the event file (for later reference).

events: A TTree storing all of the events. It has the following branches:

¹²Both importance sampling and rejection sampling have been used to reduce the variance of the weights. Nevertheless, the scale for rejection sampling can be increased by the user, if uniform event weights are preferred in performing some experimental analyses.

type) Whether the event is non-radiative (1), radiative (2), or exclusive (3).

weight) The variable weight of the event.

x, y, z, ph_t_sq, phi, phi_h) The SIDIS phase-space variables of the event.

R, tau, phi_k) Additional phase-space variables for radiative/exclusive events.

p, k1, k2, q, ph, k) Optional particle four-momenta, which will be included only if the `file.write_momenta` is set in the parameter file.

mc_coords[9], jac) Optional internal MC coordinates for the event, which will be included only if the `file.write_mc_coords` is set in the parameter file.

stats: A subdirectory storing statistics about the generated events. It has the following entries:

num_events[4]) A number of events in an array that is indexed by the event type ([0] for all events, [1] for non-radiative events, [2] for radiative events, [3] for exclusive events). This number will be greater than the number of entries in `events` because some events are generated and then later rejected, either by cuts or by rejection sampling.

num_events_acc[4]) A number of accepted events by event type, which is equal to the number of entries in `events`.

norm[4]) A normalization needed to calculate the cross section by event type (see note on normalization below).

prime[4]) A low quality estimate of cross section by event type. There are no guarantees that this is accurate; if a good cross-section estimate is needed, the normalization process should be followed.

Normalization: The produced events have variable weights, although if the various variance-reduction methods are successful, most the weights should be close to 1. A normalization is needed to convert the event weights into a cross section. The normalization can be different for events of different types,

such as a non-radiative normalization $\mathcal{N}_{\text{nrad}}$ and a radiative normalization \mathcal{N}_{rad} . Then the cross section is calculated from the event weights w_i as

$$\sigma_{\text{SIDIS}}^{\text{in}} \approx \frac{1}{\mathcal{N}_{\text{nrad}}} \sum_i w_{i,\text{nrad}} + \frac{1}{\mathcal{N}_{\text{rad}}} \sum_i w_{i,\text{rad}} + \dots \quad (\text{B3})$$

The normalizations are calculated by `sidisgen` and included in the output ROOT file within the array `stats/norm[4]`. We have, $\mathcal{N}_{\text{nrad}} = \text{norm}[1]$, $\mathcal{N}_{\text{rad}} = \text{norm}[2]$, and $\mathcal{N}_{\text{excl}} = \text{norm}[3]$. However, `sidisgen` will attempt to generate events, so that all of the normalizations are nearly equal. The ‘‘average’’ normalization is provided as $\mathcal{N} = \text{norm}[0]$, and should be sufficient for most cases. Then the cross section is calculated without paying attention to event types:

$$\sigma_{\text{SIDIS}}^{\text{in}} \approx \frac{1}{\mathcal{N}} \sum_i w_i. \quad (\text{B4})$$

Inspecting output ROOT file: The `sidisgen --inspect` command will give a brief description of an output ROOT file that is produced using `sidisgen`. This description includes the parameters used to produce the output file, and some statistics of the events contained in the output file.

Merging ROOT files: The `sidisgen --merge-soft` command will merge multiple output files together into a single ROOT file. The merged file still refers to the other output files through filename references, such that the original output files must not be renamed or deleted. The merged file can be used with any analysis script that expects a `sidisgen` output file. Alternatively, the `sidisgen --merge-hard` command will perform a hard copy, such that the original output files can then be renamed or deleted. However, this is not recommended as some information from the original ROOT files will be lost during the merging process. It will take much longer to complete than a soft merge.

Quick start (library): Listing 3 shows how to compute the Born cross section. Several other demonstrations of different features of the `libsidis` library can be found in the `example` folder, which is located at

<https://github.com/duanebyer/sidis/tree/master/example>

Listing 3: An example program using `libsidis` to compute the Born cross section.

```

1 #include <iostream>
2 #include <sidis/sidis.hpp>
3 #include <sidis/sf_set/prokudin.hpp>
4
5 sidis::Real const PI = sidis::PI;
6 sidis::Real const M_TH = sidis::MASS_P + sidis::MASS_PI_0;
7
8 int main() {
9     sidis::part::Particles particles(
10         sidis::part::Nucleus::P, // Target nucleus.
11         sidis::part::Lepton::E, // Beam lepton.
12         sidis::part::Hadron::PI_P, // Leading hadron.
13         M_TH // Threshold mass of undetected part.
14     );
15     sidis::Real S = 2. * 10.6 * particles.M; // Kinematic variable 'S = 2 p k1'.
16     sidis::kin::PhaseSpace phase_space {
17         0.2, // Bjorken x.
18         0.9, // Bjorken y.
19         0.3, // Bjorken z.
20         2., // Transverse momentum of hadron, squared.
21         0.5 * PI, // Azimuthal angle of hadron.
22         0., // Azimuthal angle of transverse target polarization.
23     };
24     sidis::kin::Kinematics kin(particles, S, phase_space);
25     sidis::Real beam_pol = 0.;
26     sidis::math::Vec3 target_pol(0., 0., 0.);
27     // Compute structure functions with WW-type approximation.
28     sidis::sf::set::ProkudinSfSet sf;
29     sidis::Real born_xs = sidis::xs::born(kin, sf, beam_pol, target_pol);
30     std::cout << "Born_unpolarized_cross-section_is_" << born_xs << std::endl;
31     return 0;
32 }

```

References

- [1] J. J. Ethier and E. R. Nocera, *Ann. Rev. Nucl. Part. Sci.* **70**, 43-76 (2020) [arXiv:2001.07722 [hep-ph]].
- [2] A. Metz and A. Vossen, *Prog. Part. Nucl. Phys.* **91**, 136-202 (2016) [arXiv:1607.02521 [hep-ex]].
- [3] A. V. Belitsky, X. d. Ji and F. Yuan, *Phys. Rev. D* **69**, 074014 (2004) [arXiv:hep-ph/0307383 [hep-ph]].
- [4] D. Boer and P. J. Mulders, *Phys. Rev. D* **57**, 5780-5786 (1998) [arXiv:hep-ph/9711485 [hep-ph]].
- [5] K. Goeke, A. Metz and M. Schlegel, *Phys. Lett. B* **618**, 90-96 (2005) [arXiv:hep-ph/0504130 [hep-ph]].

- [6] A. Bacchetta, M. Diehl, K. Goeke, A. Metz, P. J. Mulders and M. Schlegel, JHEP **02**, 093 (2007) [arXiv:hep-ph/0611265 [hep-ph]].
- [7] S. M. Aybat and T. C. Rogers, Phys. Rev. D **83**, 114042 (2011) [arXiv:1101.5057 [hep-ph]].
- [8] R. Angeles-Martinez, A. Bacchetta, I. I. Balitsky, D. Boer, M. Boglione, R. Boussarie, F. A. Ceccopieri, I. O. Cherednikov, P. Connor and M. G. Echevarria, *et al.* Acta Phys. Polon. B **46**, no.12, 2501-2534 (2015) [arXiv:1507.05267 [hep-ph]].
- [9] V. Barone, M. Boglione, J. O. Gonzalez Hernandez and S. Melis, Phys. Rev. D **91**, no.7, 074019 (2015) [arXiv:1502.04214 [hep-ph]].
- [10] S. Bastami, H. Avakian, A. V. Efremov, A. Kotzinian, B. U. Musch, B. Parsamyan, A. Prokudin, M. Schlegel, G. Schnell and P. Schweitzer, *et al.*, JHEP **06**, 007 (2019) [arXiv:1807.10606 [hep-ph]].
- [11] J. C. Collins, D. E. Soper and G. F. Sterman, Adv. Ser. Direct. High Energy Phys. **5**, 1-91 (1989) [arXiv:hep-ph/0409313 [hep-ph]].
- [12] S. Arnold, A. Metz and M. Schlegel, Phys. Rev. D **79**, 034005 (2009) [arXiv:0809.2262 [hep-ph]].
- [13] A. Airapetian, *et al.* [HERMES], Nucl. Instrum. Meth. A **540**, 68-101 (2005) [arXiv:physics/0408137 [physics.ins-det]].
- [14] V. Andrieux, A. Berlin, N. Doshita, M. Finger, M. Finger, Jr., F. Gaucheron, N. Horikawa, S. Ishimoto, T. Iwata and Y. Kisselev, *et al.* Nucl. Instrum. Meth. A **1025**, 166069 (2022)
- [15] C. A. Aidala, A. Bacchetta, M. Boglione, G. Bozzi, V. Carassiti, M. Chiosso, R. Cimino, G. Ciullo, M. Contalbrigo and U. D'Alesio, *et al.* "The LHCSpin Project," [arXiv:1901.08002 [hep-ex]].
- [16] S. Goertz, W. Meyer and G. Reicherz, Prog. Part. Nucl. Phys. **49**, 403-489 (2002) [erratum: Prog. Part. Nucl. Phys. **51**, 309-312 (2003)].
- [17] D. Keller, D. Crabb and D. Day, Nucl. Instrum. Meth. A **981**, 164504 (2020) [arXiv:2008.09515 [physics.ins-det]].

- [18] A. Airapetian, *et al.* [HERMES], Phys. Rev. Lett. **94**, 012002 (2005) [arXiv:hep-ex/0408013 [hep-ex]].
- [19] A. Airapetian, *et al.* [HERMES], Phys. Rev. Lett. **103**, 152002 (2009) [arXiv:0906.3918 [hep-ex]].
- [20] A. Airapetian, *et al.* [HERMES], Phys. Lett. B **693**, 11-16 (2010) [arXiv:1006.4221 [hep-ex]].
- [21] A. Airapetian, *et al.* [HERMES], Phys. Rev. D **87**, no.1, 012010 (2013) [arXiv:1204.4161 [hep-ex]].
- [22] A. Airapetian *et al.* [HERMES], Phys. Rev. D **87**, 074029 (2013) [arXiv:1212.5407 [hep-ex]].
- [23] V. Y. Alexakhin, *et al.* [COMPASS], Phys. Rev. Lett. **94**, 202002 (2005) [arXiv:hep-ex/0503002 [hep-ex]].
- [24] C. Adolph, *et al.* [COMPASS], Phys. Lett. B **717**, 383-389 (2012) [arXiv:1205.5122 [hep-ex]].
- [25] C. Adolph, *et al.* [COMPASS], Phys. Lett. B **744**, 250-259 (2015) [arXiv:1408.4405 [hep-ex]].
- [26] C. Adolph, *et al.* [COMPASS], Nucl. Phys. B **886**, 1046-1077 (2014) [arXiv:1401.6284 [hep-ex]].
- [27] M. Aghasyan, *et al.* [COMPASS], Phys. Rev. D **97**, no.3, 032006 (2018) [arXiv:1709.07374 [hep-ex]].
- [28] X. Qian, *et al.* [Jefferson Lab Hall A], Phys. Rev. Lett. **107**, 072003 (2011) [arXiv:1106.0363 [nucl-ex]].
- [29] M. Osipenko, *et al.* [CLAS], Phys. Rev. D **80**, 032004 (2009) [arXiv:0809.1153 [hep-ex]].
- [30] I. Akushevich and A. Ilyichev, Phys. Rev. D **100**, no.3, 033005 (2019) [arXiv:1905.09232 [hep-ph]].
- [31] D. Y. Bardin and N. M. Shumeiko, Nucl. Phys. B **127**, 242-258 (1977).
- [32] N. M. Shumeiko, Sov. J. Nucl. Phys. **29**, 807 (1979).

- [33] L. W. Mo and Y. S. Tsai, Rev. Mod. Phys. **41**, 205-235 (1969).
- [34] A. Ilyichev, I. Akushevich and S. Srednyak, Radiative Corrections in polarized SiDIS; *APCTP Focus Program in Nuclear Physics 2022: Hadron Physics Opportunities with JLab Energy and Luminosity Upgrade*, (2022), <https://indico.knu.ac.kr/event/566/contributions/525/>
- [35] S. V. Goloskokov and P. Kroll, Eur. Phys. J. C **65**, 137-151 (2010) [arXiv:0906.0460 [hep-ph]].
- [36] T. Liu, W. Melnitchouk, J. W. Qiu and N. Sato, JHEP **11** (2021), 157 [arXiv:2108.13371 [hep-ph]].
- [37] B. Karki, D. Byer and H. Gao, Comparison of Radiative Correction Factor for Semi-Inclusive Deep Inelastic Scattering between Factorized and Traditional approaches: *the paper is in its final stage of preparation; the intended journal is JHEP*.
- [38] I. Akushevich, H. Bottcher and D. Ryckbosch, RADGEN 1.0: Monte Carlo generator for radiative events in DIS on polarized and unpolarized targets, [arXiv:hep-ph/9906408 [hep-ph]].
- [39] Radiative Corrections Helpdesk, <https://www.jlab.org/RC/>
- [40] I. Akushevich, A. Ilyichev, N. Shumeiko, A. Soroko and A. Tolkachev, Comput. Phys. Commun. **104**, 201-244 (1997) [arXiv:hep-ph/9706516 [hep-ph]].
- [41] Codes for Radiative Correction Calculations, <http://www.hep.by/rc/>
- [42] I. Akushevich, A. Ilyichev and M. Osipenko, Phys. Lett. B **672**, 35-44 (2009) [arXiv:0711.4789 [hep-ph]].
- [43] I. Akushevich, N. Shumeiko and A. Soroko, Eur. Phys. J. C **10**, 681-687 (1999) [arXiv:hep-ph/9903325 [hep-ph]].
- [44] N. Varnava, Radiative Corrections Using the HAPRAD Code, (2014) <https://cds.cern.ch/record/1751384/files/Radiative%20Corrections%20Using%20the%20Haprad%20Code.pdf>
- [45] D. Byer, SIDIS-RC EvGen (SIDIS radiative correction event generator), (2020) <https://github.com/duanebyer/sidis>

- [46] J. P. Chen, H. Gao, T. K. Hemmick, Z.-E. Meziani, P. A. Souder [SoLID], [arXiv:1409.7741 [nucl-ex]].
- [47] V. D. Burkert, “CLAS12 and its initial Science Program at the Jefferson Lab Upgrade, [arXiv:0810.4718 [hep-ph]].
- [48] V. D. Burkert, *Ann. Rev. Nucl. Part. Sci.* **68**, 405-428 (2018).
- [49] A. Accardi, J. L. Albacete, M. Anselmino, N. Armesto, E. C. Aschenauer, A. Bacchetta, D. Boer, W. K. Brooks, T. Burton and N. B. Chang, *et al.* *Eur. Phys. J. A* **52**, no.9, 268 (2016) [arXiv:1212.1701 [nucl-ex]].
- [50] H. Avakian, H. Matevosyan, B. Pasquini and P. Schweitzer, *J. Phys. G* **42**, 034015 (2015).
- [51] H. Avakian, *EPJ Web Conf.* **85**, 02023 (2015).
- [52] A. Prokudin and K. Tezgin, WW-SIDIS, (2018) <https://github.com/prokudin/WW-SIDIS>
- [53] A. Bacchetta, U. D’Alesio, M. Diehl and C. A. Miller, *Phys. Rev. D* **70**, 117504 (2004) [arXiv:hep-ph/0410050 [hep-ph]].
- [54] M. Diehl and S. Sapeta, *Eur. Phys. J. C* **41**, 515-533 (2005) [arXiv:hep-ph/0503023 [hep-ph]].
- [55] M. Anselmino, M. Boglione, U. D’Alesio, A. Kotzinian, F. Murgia and A. Prokudin, *Phys. Rev. D* **71**, 074006 (2005) [arXiv:hep-ph/0501196 [hep-ph]].
- [56] P. Schweitzer, T. Teckentrup and A. Metz, *Phys. Rev. D* **81**, 094019 (2010) [arXiv:1003.2190 [hep-ph]].
- [57] M. Anselmino, M. Boglione, U. D’Alesio, A. Kotzinian, F. Murgia, A. Prokudin and S. Melis, *Nucl. Phys. B Proc. Suppl.* **191**, 98-107 (2009) [arXiv:0812.4366 [hep-ph]].
- [58] M. Anselmino, M. Boglione, U. D’Alesio, A. Kotzinian, S. Melis, F. Murgia, A. Prokudin and C. Turk, *Eur. Phys. J. A* **39**, 89-100 (2009) [arXiv:0805.2677 [hep-ph]].

- [59] M. Anselmino, M. Boggione, J. O. Gonzalez Hernandez, S. Melis and A. Prokudin, JHEP **04**, 005 (2014) [arXiv:1312.6261 [hep-ph]].
- [60] A. D. Martin, W. J. Stirling, R. S. Thorne and G. Watt, Eur. Phys. J. C **63**, 189-285 (2009) [arXiv:0901.0002 [hep-ph]].
- [61] Martin-Stirling-Thorne-Watt Parton Distribution Functions, <https://mstwpdf.hepforge.org/>
- [62] LHAPDF 6.3.0 PDF sets, <https://lhpdf.hepforge.org/pdfsets.html>
- [63] T. Liu, Z. Zhao, W. Xiong, H. Gao, LiuSIDIS: A sidis event generator, <https://github.com/TianboLiu/LiuSIDIS>
- [64] S. Wandzura and F. Wilczek, Phys. Lett. B **72**, 195-198 (1977).
- [65] W. Mao and Z. Lu, Phys. Rev. D **87**, no.1, 014012 (2013) [arXiv:1210.4790 [hep-ph]].
- [66] H. Burkhardt and B. Pietrzyk, Phys. Lett. B **356**, 398-403 (1995).
- [67] S. Jadach, A general Purpose Cellular Monte-Carlo Event Generator; Version 2.x, <http://jadach.web.cern.ch/jadach/Foam/Index.html>
- [68] ROOT: data analysis framework, <https://root.cern/>
- [69] P.A. Zyla, *et al.* [Particle Data Group], PTEP **2020**, no.8, 083C01 (2020), <https://pdg.lbl.gov/>
- [70] N. Batchelder, Cog: a file/code generation tool with Python, <https://nedbatchelder.com/code/cog/>
- [71] L. Martino, V. Elvira and F. Louzada, Signal Processing, 131, 386 (2017) [arXiv:1602.03572 [stat.CO]].
- [72] A. Kong, A note on importance sampling using standardized weights. University of Chicago, Dept. of Statistics, Tech. Rep, 348, <https://victorelvira.github.io/papers/kong92.pdf>
- [73] vegas, <https://github.com/gplepage/vegas>, vegas Documentation, <https://vegas.readthedocs.io/en/latest/index.html#>

- [74] GSL - GNU Scientific Library, <https://www.gnu.org/software/gsl/>
- [75] S. G. Johnson, *et al.*, Cubature v1.0.3: a package for adaptive multidimensional integration of vector-valued integrands over hypercubes, <https://github.com/stevengj/cubature>
- [76] J. C. Collins, Nucl. Phys. B **396**, 161-182 (1993) [arXiv:hep-ph/9208213 [hep-ph]].
- [77] D. W. Sivers, Phys. Rev. D **41**, 83 (1990).
- [78] I. Akushevich, H. Gao, A. Ilyichev and M. Meziane, Eur. Phys. J. A **51**, no.1, 1 (2015).
- [79] U. D'Alesio, C. Flore and A. Prokudin, Phys. Lett. B **803**, 135347 (2020) [arXiv:2001.01573 [hep-ph]].
- [80] I. Scimemi and A. Vladimirov, JHEP **06** (2020), 137 [arXiv:1912.06532 [hep-ph]].
- [81] V. Bertone, I. Scimemi and A. Vladimirov, JHEP **06** (2019), 028 [arXiv:1902.08474 [hep-ph]].
- [82] A. Bacchetta *et al.* [MAP], JHEP **10** (2022), 127 [arXiv:2206.07598 [hep-ph]].
- [83] J. Cammarota *et al.* [Jefferson Lab Angular Momentum], Phys. Rev. D **102** (2020) no.5, 054002 [arXiv:2002.08384 [hep-ph]].
- [84] M. Cerutti, A. Bacchetta, V. Bertone, C. Bissolotti, G. Bozzi, F. Piacenza, M. Radici and A. Signori, JPS Conf. Proc. **37** (2022), 020123.
- [85] L. Gamberg *et al.* [Jefferson Lab Angular Momentum (JAM) and Jefferson Lab Angular Momentum], Phys. Rev. D **106** (2022) no.3, 034014 [arXiv:2205.00999 [hep-ph]].
- [86] M. G. Echevarria, Z. B. Kang and J. Terry, JHEP **01** (2021), 126 [arXiv:2009.10710 [hep-ph]].
- [87] A. Bacchetta, F. Delcarro, C. Pisano and M. Radici, Phys. Lett. B **827** (2022), 136961 [arXiv:2004.14278 [hep-ph]].

- [88] M. Bury, A. Prokudin and A. Vladimirov, Phys. Rev. Lett. **126** (2021) no.11, 112002 [arXiv:2012.05135 [hep-ph]].
- [89] M. Bury, A. Prokudin and A. Vladimirov, JHEP **05** (2021), 151 [arXiv:2103.03270 [hep-ph]].
- [90] P. C. Barry, L. Gamberg, W. Melnitchouk, E. Moffat, D. Pitonyak, A. Prokudin and N. Sato, [arXiv:2302.01192 [hep-ph]].
- [91] M. Davier, A. Hoecker, B. Malaescu and Z. Zhang, Eur. Phys. J. C **80**, no.3, 241 (2020) [erratum: Eur. Phys. J. C **80**, no.5, 410 (2020)] [arXiv:1908.00921 [hep-ph]].
- [92] M. Davier, A. Hoecker, B. Malaescu and Z. Zhang, Eur. Phys. J. C **77**, no.12, 827 (2017) [arXiv:1706.09436 [hep-ph]].
- [93] R. Bonciani, P. Mastrolia and E. Remiddi, Nucl. Phys. B **676**, 399-452 (2004) [arXiv:hep-ph/0307295 [hep-ph]].
- [94] F. Jegerlehner and A. Nyffeler, Phys. Rept. **477**, 1-110 (2009) [arXiv:0902.3360 [hep-ph]].
- [95] F. Jegerlehner, Springer Tracts Mod. Phys. **274**, pp.1-693, (2017) Springer.
- [96] F. Jegerlehner, Software package alphaQED, <http://www-com.physik.hu-berlin.de/~fjeger/software.html>
- [97] R. Abdul Khalek, A. Accardi, J. Adam, D. Adamiak, W. Akers, M. Albaladejo, A. Al-bataineh, M. G. Alexeev, F. Ameli and P. Antonioli, *et al.* Nucl. Phys. A **1026**, 122447 (2022) [arXiv:2103.05419 [physics.ins-det]].
- [98] J. Bellm, S. Gieseke, D. Grellscheid, S. Plätzer, M. Rauch, C. Reuschle, P. Richardson, P. Schichtel, M. H. Seymour and A. Siódmok, *et al.*, Eur. Phys. J. C **76**, no.4, 196 (2016) [arXiv:1512.01178 [hep-ph]], <https://herwig.hepforge.org/>