Thereafter

PDF Fitting package

HERAFitter developers

March 22, 2013

Abstract

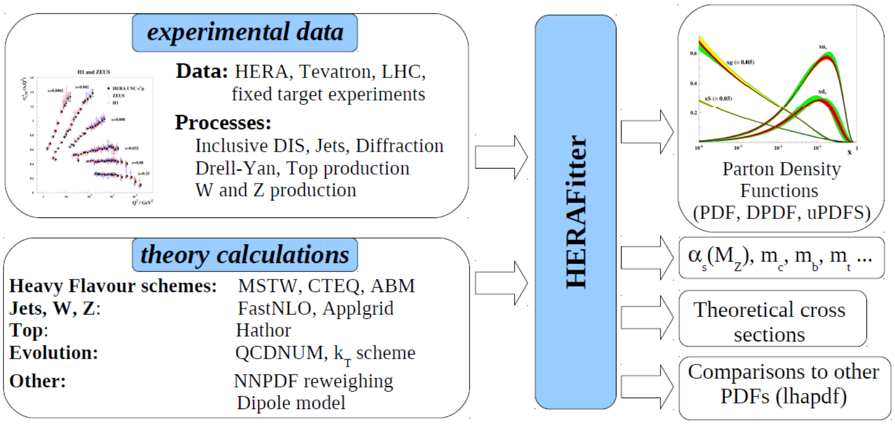
The determination of the proton patron distribution functions is a complex endeavor involving several physics process. The main process is the deep-inelastic scattering and the central data set covering most of the proton structure phase space is provided at HERA ep collider. Further processes (fixed target DIS, ppbar collisions etc.) provide further constrains for particular aspects: flavor separation, very high Bjorken-x etc. In particular, the precise measurements obtained or to come from LHC will continue to improve the knowledge of the PDF. HERAFitter project aim at providing a framework for QCD analyses related to the proton structure in the context of multi-process and multi-experiment. The framework includes modules or interfaces enabling a large number of theoretical and methodological options, as well as a large number of relevant data sets from HERA, Tevatron and LHC. This manual explains the theoretical input used in the QCD analysis, the fit methodology as well as the the installation procedure of the program. More information and the package downloads can be found on the web site `http://herafitter.org`.

Contents

1 Introduction

This manual provides a short description of the `Thereafter` program which can be used to determine unpolarised proton parton density functions (PDFs). The parton density functions are needed to calculate cross sections for ep, pp, and $p\overline{p}$ colliders and thus they are required for interpretation of the data collected at the LHC.

A schematic structure of the `Thereafter` is illustrated in Fig. 1 which encapsulates all the current functionality of the platform.



experimental data
Data: HERA, Tevatron, LHC, fixed target experiments
Processes:
Inclusive DIS, Jets, Diffraction
Drell-Yan, Top production
W and Z production
theory calculations
Heavy Flavour schemes: MSTW, CTEQ, ABM
Jets, W, Z: FastNLO, Applgrid
Top: Hathor
Evolution: QCDNUM, k_T scheme
Other: NNPDF reweighing
Dipole model
HERAFitter
Parton Density Functions
(PDF, DPDF, uPDFS)
α_s(M_Z), m_c, m_b, m_t ...
Theoretical cross sections
Comparisons to other PDFs (lhapdf)

Figure 1: Schematic structure of the `Thereafter` program.

The manual is structured such that it first describes briefly the theoretical input (section 2), followed by a description of the PDF parameterisation (section 3.1) and various χ^2 functions used in the minimisation (section 3.2). The minimisation is based on the standard MINUIT program [1] which is not discussed here. Section 5 is dedicated to program installation instructions for different fit scenarios (section 5.1) and provides a description of the program steering cards with the output options given in section 5.3.

2 Theoretical Input

The main features of the QCD theory are the confinement (at short ranges the quarks are strongly bound inside protons) and the asymptotic freedom (at large scales the coupling constant of strong force decreases and quarks become quasi-free partons). The factorisation theorem exploits these features by separating the short and long distances processes, such that structure functions can be written as a convolution between calculable parts (hard scattering coefficients) and non-calculable parts parton distribution functions (PDFs), which are therefore parametrised and determined from data.

Factorisation is most rigorously established for deep inelastic lepton-hadron scattering. For hadronic processes in which a colorless electroweak final state is produced (i.e. Higgs, a real or virtual W, Z or γ) factorisation is also well established and differential cross section calculations are currently available up to next-to-next-to leading order (NNLO) in perturbation theory.

Factorisation is also proved to work for sufficiently inclusive colored final states, such as the one-jet and dijet cross section. In this case no fully rigorous all-order demonstration is available, but no counterexample to factorisation has been found so far.

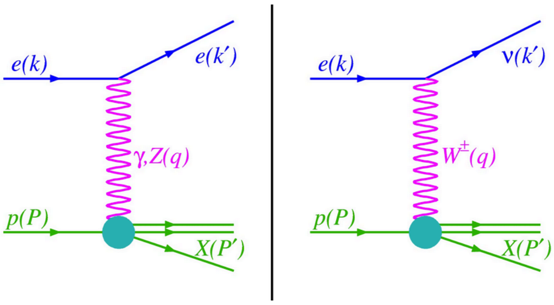


Figure 2: Diagram for a neutral and charged current DIS scattering.

For the diffractive case, two factorisations are used here: the Regge factorisation and the collinear factorisation for the hard scattering.

The proton PDFs are classically extracted from the QCD fits by a measure of agreement between data and theory models. The fit procedure used in `Thereafter` framework is common to all the processes and it consists first in parametrising PDFs at a starting scale Q_0^2, chosen to be below the charm mass threshold. The PDFs are then evolved using coupled, integro-differential Dokshitzer-Gribov-Lipatov-Altarelli-Parisi (DGLAP) [2, 3, 4, 5, 6] evolution equations as implemented in the QCDNUM [7] program, at NLO [8, 9] in the $\overline{\text{MS}}$ scheme (LO and NNLO evolutions are also available). The PDFs calculated at a scale corresponding to an measured cross section are convoluted with the partonic cross sections to calculate the predicted cross section. For all measurement points, the predicted and measured cross sections together with the corresponding errors are used to build a global χ^2, which is minimized to deduce the initial PDF parameters. This generic procedure includes a number of subtitles and assumptions, depending on the measurements, scale and available calculations.

In the following sections, the theoretical input for various processes is described, such as electron-proton deep inelastic scattering (DIS) process in section 2.1, *pp* Drell-Yan process in section 2.2. For the jet cross sections calculations, `Thereafter` uses APPLGRID or FastNLO such as in section 2.4. In section 2.3 the $t\bar{t}$ cross sections are calculated based on the HATHOR package. Alternative approaches to collinear factorisation are also discussed, see sections 2.5 for Dipole models and 2.6 for unintegrated PDFs. The diffractive PDFs are discussed as well in section 2.7, i.e. PDFs in the context of the resolved Pomeron model to predict such processes as seminclusive diffractive scattering in DIS.

2.1 Deep Inelastic Scattering Formalism and Schemes

The DIS experiments provide the cleanest approach to measure the proton structure. The DIS experiments are carried either on fixed targets or at the collider facilities using electrons, muons or neutrinos to probe the proton. In the DIS process a lepton is scattered off the constituents of the proton by a virtual exchange of neutral (NC) or charged (CC) boson producing a hadronic shower and a scattered lepton in the final state. The schematic of the DIS kinematical variables is illustrated in figure 2: the negative squared four-momentum of the exchange boson, Q^2, the scaling variable x, which can be related in the parton model to the fraction of momentum carried by the struck quark, and the inelasticity parameter y, which is the fraction of the energy transferred to the hadronic vertex. The NC (and similarly CC) cross section can be expressed in terms of generalised structure functions:

$$\frac{d^2\sigma_{NC}^{e^\pm p}}{dxdQ^2} = \frac{2\pi\alpha^2}{xQ^4}[Y_+\tilde{F}_2^\pm \mp Y_- x\tilde{F}_3^\pm - y^2\tilde{F}_L^\pm], \tag{1}$$

where $Y_\pm = 1 \pm (1-y)^2$ with y being the inelasticity. The structure function $\tilde{F}_2$ is the dominant contribution to the cross section, $x\tilde{F}_3$ is important at high Q^2 and $\tilde{F}_L$ is sizable only at high y. $\tilde{F}_2^\pm$ and $\tilde{F}_3^\pm$ can be expressed in terms of five structure functions describing the contributions from pure photon exchange, γZ interference and pure Z exchange:

$$\tilde{F}_2^\pm = F_2 - v_e\left(\frac{\kappa_W Q^2}{Q^2 + M_Z^2}\right)F_2^{\gamma Z} + (v_e^2 + a_e^2)\left(\frac{\kappa_W Q^2}{Q^2 + M_Z^2}\right)^2 F_2^Z \tag{2}$$

$$x\tilde{F}_3^\pm = \pm a_e\left(\frac{\kappa_W Q^2}{Q^2 + M_Z^2}\right)xF_3^{\gamma Z} \mp 2a_e v_e\left(\frac{\kappa_W Q^2}{Q^2 + M_Z^2}\right)^2 xF_3^Z \tag{3}$$

Here, the pure photon exchange is described by F_2, pure Z exchange by F_2^Z and xF_3^Z, and γZ interference by $F_2^{\gamma Z}$ and $xF_3^{\gamma Z}$. v_e is the weak vector and a_e the weak axial-vector coupling of the electron to the Z. The Weinberg angle θ_W enters the quantity κ_W in the following way: $\kappa_W = \frac{1}{4\sin^2\theta_W \cos^2\theta_W}$.

In the framework of perturbative QCD the structure functions are directly related to the parton distribution functions, i.e. in leading order (LO) F_2 is the momentum sum of quark and anti-quark distributions:

$$[F_2, F_2^{\gamma Z}, F_2^Z] = x\sum_q [e_q^2, 2e_q v_q, v_q^2 + a_q^2]\{q(x,Q^2) + \bar{q}(x,Q^2)\} \tag{4}$$

$$[xF_3^{\gamma Z}, xF_3^Z] = x\sum_q [e_q^2 a_q, 2v_q a_q]\{q(x,Q^2) - \bar{q}(x,Q^2)\} \tag{5}$$

In analogy to neutral currents, the inclusive CC ep cross section can be expressed in terms of structure functions:

$$\frac{d^2\sigma_{CC}^{e^\pm p}}{dxdQ^2} = \frac{G_F^2}{4\pi x}\left(\frac{M_W^2}{Q^2 + M_W^2}\right)^2 [Y_+ W_2^\pm + y^2 W_L^\pm \mp Y_- x W_3^\pm], \tag{6}$$

Here, G_F is the Fermi constant which is related to the weak coupling g and electromagnetic coupling e, i.e. $G_F = \frac{g^2}{4\sqrt{2}M_W^2}$. In LO the e^+p and e^-p cross sections are sensitive to different quark densities:

$$\begin{aligned} \tilde{\sigma}_{CC}^{e^+p} &= x[\bar{u} + \bar{c}] + (1-y)^2 x[d + s] \\ \tilde{\sigma}_{CC}^{e^-p} &= x[u + c] + (1-y)^2 x[\bar{d} + \bar{s}]. \end{aligned} \tag{7}$$

The cross-section predictions are obtain by convoluting the PDFs with the hard scattering coefficient functions. There are various approaches of dividing the structure functions into calculable processes and PDFs. For the DIS processes, those are calculated using the Fixed-Flavour number (FFN) [10, 11, 12] or the general mass Variable-Flavour number (GM-VFN) [13] schemes. In the FFN scheme, heavy quark contributions are explicitly included in the hard cross sections. In the VFN scheme, PDFs corresponding to heavy quarks are introduced and the number of active flavors changes by one unit when the scale crosses the threshold for heavy quark distribution ($Q^2 \geq m_Q^2$).

The evolution program QCDNUM [7] used in `Thereafter` provides the calculation of the deep inelastic structure functions in the zero-mass, generalised mass and the fixed flavour number schemes. The VFN schemes with various treatments for the heavy quark thresholds are considered in `Thereafter` :

- the Thorne Roberts (TR) scheme with its variants at NLO and NNLO [14, 15] as provided by the MSTW group,
- the ACOT scheme with its variants at LO and NLO as provided by the CTEQ group,
- BMSN scheme at NLO and NNLO [1].

The fixed-flavour number scheme is available via the QCDNUM implementation and via the `OPENQCDRAD` [16] interface. Each of these schemes is briefly discussed in further details.

2.1.1 Zero-Mass Variable Flavour Scheme

In the zero-mass variable flavour number scheme (ZM-VFNS) heavy quark densities are included in the proton for $Q^2 >> m_h^2$ but they are treated as massless in both, the initial and final states. This scheme is accurate in the region where Q^2 is much greater than m_h^2 but becomes unreliable for $Q^2 \sim m_h^2$.

2.1.2 General Mass Variable Flavour Scheme: Thorne-Roberts scheme

The Thorne-Roberts (TR) scheme, (referred as RT scheme in the `Thereafter`) is a general-mass variable flavour number scheme (GM-VFNS) used as default for the MTSW PDF sets. GM-VFNS smoothly connects the two regions: scales below the heavy quark mass scale ($Q^2 < m_h^2$) and scales much above the heavy quark scale threshold ($Q^2 >> m_h^2$). However, the definition is not unique. A GM-VFNS can be defined by demanding equivalence of the $n_f = n$ (FFNS) and $n_f = n + 1$ flavour (ZM-VFNS) descriptions above the transition point for the new parton distributions (they are by definition identical below this point), at all orders.

The TR scheme has two different variants depending on the treatment across the threshold of heavy quarks: TR standard (as used in MSTW PDF sets [15, 17]) and TR optimal [18], with a smoother transition across the heavy quark mass scales. Both of these variants are accessible within the `Thereafter` package. The calculations are available to NLO and NNLO order. In addition, a fast version of the scheme is available (i.e. RT FAST) by using the k-factor technique. The k-factors are applied at the first iteration of the minimisation process enabling the code to run fast, as the k-factors are defined as the ratio between massless and massive scheme, while the massless scheme accessed via QCDNUM is very fast.

2.1.3 General Mass Variable Flavour Scheme: ACOT scheme

The Aivazis-Collins-Olness-Tung scheme belongs to the group of VFN factorisation schemes that use the renormalization method of Collins-Wilczek-Zee (CWZ) [19]. This scheme involves a mixture of the $\overline{\text{MS}}$ scheme for light partons (and for heavy partons when the factorisation scale is larger than the heavy quark mass) and the zero-momentum subtraction renormalisation scheme for graphs with heavy quark lines (if factorisation scale is smaller than the mass of the heavy quark threshold). The DGLAP kernels and PDF evolution are pure $\overline{\text{MS}}$. Therefore, the ACOT scheme is considered to be a minimal extension of $\overline{\text{MS}}$ scheme.

Within the ACOT package, different variants of the ACOT scheme are available: ACOT Full, S-ACOT Chi, ACOT ZM, $\overline{\text{MS}}$ at LO and NLO. For the longitudinal structure function higher order calculations are also available. The ACOT Full implementation fully takes into account the quark masses and

[1] The BMSN scheme as provided by ABM group currently is not yet fully implemented in `Thereafter` .

it reduces to $\overline{\text{MS}}$ in the limit of masses going to zero, but it has the disadvantage of being quite slow. Therefore the k-factor technique has been adopted within the `Thereafter` machinery to be able to perform QCD fits. The k-factor can be defined in two different ways: on one hand as the ratio between same order calculations but massless vs massive (i.e. NLO (ZM-VFNS)/NLO (ACOT), on the other hand one could speed up the calculations by defining the k-factors as the ratio between LO (massless)/NLO (massive). Both options are available in the `Thereafter` package and give similar results. For convergence of the k-factors usually $2-3$ iterations are needed. These different variants of this scheme are all integrated in the `Thereafter` framework and can be selected via namelist `HF_SCHEME` in the `steering.txt` (ACOT ZM, ACOT FULL, S-ACOT Chi).

The differences between TR and ACOT scheme types are summarised in the figure 3. One major issue in a complete GM-VFNS, is that of the ordering of the perturbative expansion. The equivalency of swapping the $O(m_H^2/Q^2)$ terms between Wilson coefficients (or hard-scattering amplitudes) without violating the definition of a GM-VFNS is what mainly distinguish the ACOT from TR schemes.

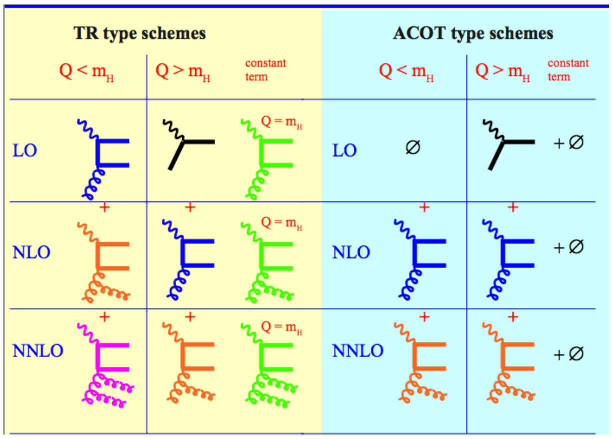


Figure 3: Schematic summary of ACOT and TR schemes.

2.1.4 Fixed -Flavour Number Scheme

As mentioned before, in the FFN scheme only gluon and the light quarks are considered as partons within the proton, massive quarks are produced perturbatively in the final state. `Thereafter` includes FFN scheme from ABM [16] and can use as well the QCDNUM implementation.

In addition, the recent variant of the fixed-flavour number scheme in which the running mass definition is used in the $\overline{\text{MS}}$ scheme [20] is implemented in `Thereafter` .This variant is realised via the interface to the open-source code OPENQCDRAD [16]. This scheme has the advantage of reducing the sensitivity of the DIS cross sections to higher order corrections, and improving the theoretical precision of the mass definition. In QCDNUM, the calculation of the heavy quark contributions to DIS structure functions are available at NLO and only electromagnetic exchange contributions are taken into account. In the ABM implementation, the QCD corrections to the massive Wilson coefficients up to the currently best known approximate NNLO for the neutral-current (NC) heavy-quark production [21] and up to NLO for the charged-current (CC) case are available.

2.1.5 Electroweak corrections for ep scattering

To properly compare the experimental data with theoretical predictions, QED corrections are necessary. In the `Thereafter` the electroweak corrections for the DIS process are based on the EPRC package [22].

The calculations of higher-order electroweak corrections to DIS scattering at HERA are performed in the on-shell scheme where the gauge bosons masses M_W abd M_Z are treated symmetrically as basic parameters together with the top and Higgs masses, besides the fine structure constant α and other fermion masses.

The code provides the running of α using the most recent parametrisation of the hadronic contribution to Δ_α [23], as well as an older one from Burkhard [24]. For the Drell Yan processes there are independent treatments applied as k-factors (such as from SANC, FEWZ packages).

2.2 Drell Yan processes

This section presents calculations of Drell Yan processes that can be used to predict lepton pair production at the LHC or Tevatron. A schematic of the Drell Yan process is shown in figure 4. The calculations of the Drell Yan processes are known for many observables up to the NNLO order. For example, there are packages such as FEWZ [25], DYNNLO [26] for NNLO, or MCFM [27] for NLO calculations. However, due to the complicated nature of these calculation involving an increased number of diagrams with additional higher order, these calculations are too slow to be used iteratively in a fit. There are various methods to overcome this shortage: using the k-factors approximation from lower to higher order, or using the so-called grid technique (storing the matrix elements on grids such that the cross sections later may be calculated convoluting these grids with the input PDFs) when available.

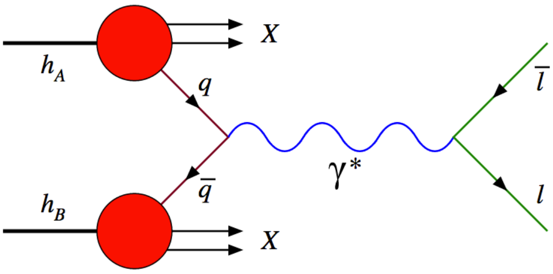


Figure 4: Diagram for a generic DY scattering.

`Thereafter` provides two implementations for pp Drell Yan processes. The first implementation uses calculations at LO which can be extended to NLO using k-factors, the second uses the APPLGRID interface. Short description of both implementations is given below while for details of the theoretical modules we direct the user to read the references of these packages provided in the description.

The leading order Drell-Yan [28, 29] cross section for the neutral current, triple differential in invariant mass M, boson rapidity y and CMS lepton scattering angle $\cos\theta$, can be written as

$$\frac{d^3\sigma}{dM dy d\cos\theta} = \frac{\pi\alpha^2}{3MS}\sum_q P_q\left[F_q(x_1,Q^2)F_{\bar{q}}(x_2,Q^2) + (q \leftrightarrow \bar{q})\right], \tag{8}$$

where S is the squared CMS beam energy, $x_{1,2} = \frac{M}{\sqrt{S}} \exp(\pm y)$, $F_q(x_1, Q^2)$ parton distribution functions, and

$$\begin{aligned} P_q &= e_l^2 e_q^2 (1 + \cos^2\theta) \\ &+ e_l e_q \frac{2M^2(M^2 - M_Z^2)}{\sin^2\theta_W \cos^2\theta_W [(M^2 - M_Z^2)^2 + \Gamma_Z^2 M_Z^2]} [a A_q (1 + \cos^2\theta) + 2 b B_q \cos\theta] \\ &+ \frac{M^4}{\sin^4\theta_W \cos^4\theta_W [(M^2 - M_Z^2)^2 + \Gamma_Z^2 M_Z^2]} [(a^2 + b^2)(A_q^2 + B_q^2)(1 + \cos^2\theta) + 8 a b A_q B_q \cos\theta]. \end{aligned} \tag{9}$$

Here θ_W is the Weinberg angle, M_Z and Γ_Z are Z boson mass and width, and

$$\begin{aligned} a &= -\frac{1}{4} + \sin^2\theta_W, \\ b &= -\frac{1}{4}, \\ A_q &= \frac{1}{2} I_q^3 - e_q \sin^2\theta_W, \\ B_q &= \frac{1}{2} I_q^3, \\ I_u^3 &= -I_d^3 = \frac{1}{2}, \\ e_l &= -1, e_u = \frac{2}{3}, e_d = -\frac{1}{3}. \end{aligned} \tag{10}$$

The expression for charged current has a simpler form:

$$\frac{\mathrm{d}^3\sigma}{\mathrm{d}M \mathrm{d}y \mathrm{d}\cos\theta} = \frac{\pi\alpha^2}{48 S \sin^4\theta_W} \frac{M^3(1 - \cos\theta)^2}{(M^2 - M_W^2) + \Gamma_W^2 M_W^2} \sum_{q_1, q_2} V_{q_1 q_2}^2 F_{q_1}(x_1, Q^2) F_{q_2}(x_2, Q^2), \tag{11}$$

where $V_{q_1 q_2}$ is the CKM quark mixing matrix and M_W and Γ_W are W boson mass and decay width.

The simple form of these expressions allows to calculate integrated cross sections without utilization of Monte-Carlo techniques. This is particularly useful for PDF fitting purposes because the statistical fluctuations are avoided in this case. In both neutral and charged current expressions the parton density functions factorise as a function dependent only on boson rapidity y and invariant mass M. The integral in $\cos\theta$ can be computed analytically and integrations in y and M can be performed with the Simpson method. The $\cos\theta$ parts are kept in the equation explicitly because their integration is asymmetric for data in lepton η bins and also is being performed when applying the lepton $p_\perp$ cuts.

The fact that PDF functions factorise, allows to significantly boost calculations when performing parameter fits over lepton rapidity data. In this case the factorised part of the expression which is independent on PDFs can be calculated only once for all minimisation iterations. The leading order code in `Thereafter` package implements this optimisation and uses fast convolution routines provided by QCDNUM. Currently the full width LO calculations are optimised for lepton pseudorapidity and boson rapidity distributions with the possibility to apply lepton $p_\perp$ cuts. This flexibility allows to perform the calculations within the phase space corresponding to the available measurement.

The calculated leading order cross sections are multiplied by k-factors to obtain predictions at NLO or NNLO precision.

On the other hand, one can obtain directly the NLO predictions by using APPLGRID or FASTNLO techniques, which rely on factorisation theorem by decoupling the hard scattering coefficients from PDFs. The hard scattering coefficients are calculated once and stored into a grid for a given kinematic bin, speeding up the convolution process with the PDFs allowing to be used for QCD fits. These method are described in more details in section 2.4

2.3 Cross Sections for $t\bar{t}$ production in pp or $p\bar{p}$ collisions

Top-quark pairs ($t\bar{t}$) are mainly produced at hadron colliders via gg fusion and $q\bar{q}$ annihilation. Furthermore, there are the qq' and qg production modes. The program HATHOR [30] allows calculating the expected total $t\bar{t}$ cross section at $p\bar{p}$ and pp colliders up to approximate NNLO accuracy. Version 1.3 of HATHOR includes the exact NNLO for $q\bar{q} \rightarrow t\bar{t}$ [31] as well as a new high-energy constraint on the approximate NNLO obtained from soft-gluon resummation [32]. The default choice for renormalization and factorization scale in $t\bar{t}$ production is the top-quark mass, m_t. The pole mass scheme is typically employed for m_t but HATHOR also supports calculations in the $\overline{\mathrm{MS}}$ scheme.

2.4 Jets

This sections presents various fast calculations techniques for jet production based on the factorization formalism.

The calculation of higher order jet cross sections is very demanding in means of computing power. The reasons are the large number of contributing Feynman diagrams and also the large number of infrared divergencies. For an accurate cancellation of these singularities, the dipole subtraction method is often applied in such calculations. During the necessary Monte Carlo integration a very fine phase space sampling has to be performed in order to account for the accurate cancellation of the counter terms.

In order to enable the inclusion of jet-cross section measurements in PDF and α_s fits, the perturbative coefficients have to be pre-computed in a PDF and α_s independent way. For this purpose, two similar tools are interfaced to the `Thereafter` .

2.4.1 FastNLO

The fastNLO project [33, 34, 35] enables the inclusion of jet data in PDF and α_s fits. This tool uses multi-dimensional interpolation techniques to convert the convolutions of perturbative coefficients with parton distribution functions and the strong coupling into simple products. The perturbative coefficients are calculated by the `NLOJET++` program [36] where calculations for jet-production in DIS [37] as well as in hadron-hadron collisions [38, 39] with threshold-corrections of O(NNLO) for inclusive jet cross sections [40] are available.

The fastNLO libraries are included in the `Thereafter` package and no further requirements or compilation options are needed. In order to include a new measurement into the PDF-fit, the fastNLO table have to be specified. These tables include all necessary information of the perturbative coefficients and the calculated process for all bins of a certain dataset. Tables for almost all published jet measurements are available through the project website `http://fastnlo.hepforge.org`.

Features of the fastNLO concept are the very quick convolution of the perturbative coefficients with the PDFs of $O(100ms)$ and the very high accuracy of the interpolation procedure. The fastNLO tables are conventionally calculated for multiple factors of the factorization scale, and the renormalization scale factor can be chosen freely. Some of the fastNLO tables already involve a scale-independent concept [35], which allows for the free choice of the renormalization and the factorization scale as a function of two pre-defined observables. The evaluation of the strong coupling constant, which enters the cross section calculation, is taken consistently from the QCDNUM evolution code.

2.4.2 APPLGRID

The APPLGRID [41] package allows to compute in a fast way an estimate of NLO cross sections for particular processes for arbitrary set of proton parton density functions. The package implements calculations of Drell Yan cross sections of electroweak boson (Z, W) production as well as jet production in proton-(anti)proton collisions and DIS processes.

The approach is based on storing perturbative coefficients of NLO QCD calculations of final-state observables measured in hadron colliders in look-up tables. The PDFs and the strong couplings are included during the final calculations, e.g. during the PDF fits procedure. The method allows variation of factorization and renormalization scales in calculations.

The look-up tables (grids) can be generated with modified versions of of MCFM [42, 43] or NLO-jet++ [39] software distributed with the full version of APPLGRID package.

APPLGRID supports the interface to the MCFM parton level generators, hence the model input parameters such as electroweak parameters are in fact pre-set following the MCFM input steering card, while binning and definitions of the observables for which the differential cross sections are needed are set in the APPLGRID code. The grid parameters, Q^2 binning and interpolation orders are also defined in the code.

APPLGRID performs construction of the grid tables in two steps: *(i)* exploration of the phase space in order to optimize the memory storage and *(ii)* actual grid construction in the phase space corresponding to the requested observables.

Afterwards the NLO cross sections are restored from the grids with providing PDFs, α_S, factorization and renormalization scales and with QCD NNLO k-factors applied if stated.

In order to use APPLGRID tables in `Thereafter` , APPLGRID package has to be downloaded and installed first. In addition, `Thereafter` code has to be configured with a special option (for details see section 5.1).

2.5 DIPOLE models

The dipole picture provides an attractive approach to the virtual photon-proton scattering in the low x region because it allows to describe both, inclusive and diffractive processes. In this approach the virtual photon fluctuates into a $q\bar{q}$ (or $q\bar{q}g$) dipole which interacts with the proton [44]. The dipoles can be viewed as quasi-stable quantum mechanical states, which have very long life time $\propto 1/m_p x$ and a size which is not changed by scattering. A schematic view of dipole factorisation at small x in DIS is illustrated in figure 5. The virtual photon fluctuates into a quark-antiquark pair and subsequently interacts with the target, and the dynamics of the interaction are embedded in the dipole scattering amplitude.

Several dipole models have been developed to describe various DIS reactions. They vary due to different assumption made about the behavior of the dipole-proton cross sections. In the `Thereafter` three representative models are implemented: the original Golec-Biernat-Wüsthoff (GBW) [45] dipole saturation model, the colour glass condensate approach to the high parton density regime Iancu-Itakura-Munier (IIM) model [46], a modified GBW model which takes into account the effects of DGLAP evolution Bartels-Golec-Kowalski(BGK) [47].

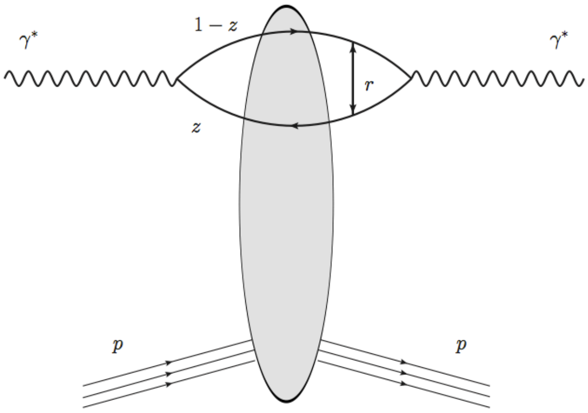


Figure 5: Schematic diagram of dipole factorisation for the inclusive cross section in DIS.

2.5.1 GBW model

In the GBW model the dipole-proton cross section $\sigma_{\rm dip}$ is given by

$$\sigma_{\rm dip}(x, r^2) = \sigma_0 \left(1 - \exp\left[-\frac{r^2}{4R_0^2(x)}\right]\right), \tag{12}$$

where r corresponds to the transverse separation between the quark and the antiquark, and R_0^2 is an x dependent scale parameter which has a meaning of saturation radius, $R_0^2(x) = (x/x_0)^{\lambda}$. The free fitted parameters are the cross-section normalisation σ_0 as well as x_0 and λ.

2.5.2 IIM model

The IIM model assumes an improved expression for the dipole cross section which is based on the Balitsky-Kovchegov equation [48]. The explicit formula for $\sigma_{\rm dip}$ can be found in [46]. The free fitted parameters are the alternative scale parameter $\tilde{R}$, x_0 and λ.

2.5.3 BGK model

The BGK model modifies the GBW model by taking into account the DGLAP evolution of the gluon density. The dipole cross section is given by

$$\sigma_{\rm dip}(x, r^2) = \sigma_0 \left(1 - \exp\left[-\frac{\pi^2 r^2 \alpha_s(\mu^2) x g(x, \mu^2)}{3\sigma_0}\right]\right).$$

The factorization scale μ^2 has the form $\mu^2 = C_{bgk}/r^2 + \mu_0^2$. In this model the gluon density, which is parametrized at some starting scale Q_0^2 by

$$xg(x, Q_0^2) = A_g x^{-\lambda_g}(1-x)^{C_g}.$$

is evolved to larger Q^2's using LO and NLO DGLAP evolution. The free fitted parameters for this model are σ_0, μ_0^2 and three parameters for the gluon density: A_g, λ_g, C_g. The parameter C_{bgk} is kept fixed: $C_{bgk} = 4.0$.

2.5.4 BGK model with valence quarks

The dipole models are valid in the low-x region only, where the valence quark contribution is small, of the order of 5%. The new HERA F_2 data have a precision which is better than 2 %. Therefore in the `Thereafter` the contribution of the valence quarks is taken from the PDF fits and added to the original BGK model, this is uniquely possible within the `Thereafter` framework. The quality of the fits of the BGK dipole model with valence quarks and without valence quarks are the same. The sample input steering and output fits are discussed in section 5.3.

2.6 Transverse Momentum Dependent (unintegrated PDF) with CCFM

In this subsection another alternative approach to the collinear DGLAP evolution is presented. In high energy factorization [49] generally the measured cross section is written as a convolution of the partonic cross section $\hat{\sigma}(k_t)$ which depends on the transverse momentum k_t of the incoming parton with the k_t-dependent parton density function $\tilde{\mathcal{A}}(x, k_t, p)$ (transverse momentum dependent (TMD) or unintegrated uPDF):

$$\sigma = \int \frac{dz}{z} d^2k_t \hat{\sigma}(\frac{x}{z}, k_t) \tilde{\mathcal{A}}(x, k_t, p) \tag{13}$$

The evolution of $\tilde{\mathcal{A}}(x, k_t, p)$ can proceed via the BFKL, DGLAP or via the CCFM evolution equations. Here, an extension of the CCFM [50, 51, 52, 53] evolution is applied. Since the evolution cannot be easily obtained in a closed form, first a kernel $\tilde{\mathcal{A}}(x'', k_t, p)$ is determined from the MC solution of the CCFM evolution equation, and then is then folded with the non-perturbative starting distribution $\mathcal{A}_0(x)$ [54]:

$$\begin{aligned} x\mathcal{A}(x, k_t, p) &= x \int dx' \int dx'' \mathcal{A}_0(x) \tilde{\mathcal{A}}(x'', k_t, p)\, \delta(x' \cdot x'' - x) && (14) \\ &= \int dx' \int dx'' \mathcal{A}_0(x) \tilde{\mathcal{A}}(x'', k_t, p)\, \frac{x}{x'} \delta(x'' - \frac{x}{x'}) && (15) \\ &= \int dx' \mathcal{A}_0(x') \cdot \frac{x}{x'} \tilde{\mathcal{A}}\left(\frac{x}{x'}, k_t, p\right) && (16) \end{aligned}$$

The kernel $\tilde{\mathcal{A}}$ includes all the dynamics of the evolution, Sudakov form factors and splitting functions and is determined in a grid of $50 \otimes 50 \otimes 50$ bins in x, k_t, p.

The calculation of the cross section according to Eq.(13) involves a multidimensional Monte Carlo integration which is time consuming and suffers from numerical fluctuations, and therefore cannot be used directly in a fit procedure involving the calculation of numerical derivatives in the search for the minimum. Instead the following procedure is applied:

$$\begin{aligned} \sigma_r(x, Q^2) &= \int_x^1 dx_g \mathcal{A}(x_g, k_t, p) \hat{\sigma}(x, x_g, Q^2) && (17) \\ &= \int_x^1 dx' \mathcal{A}_0(x') \cdot \tilde{\sigma}(x/x', Q^2) && (18) \end{aligned}$$

The kernel $\tilde{\mathcal{A}}$ has to be provided separately and is not calculable within this program. The starting distribution $\mathcal{A}_0$ at the starting scale Q_0 of the following form is used:

$$x\mathcal{A}_0(x, k_t) = N x^{-B_g} \cdot (1 - x)^{C_g} \left(1 - D_g x\right) \tag{19}$$

with free parameters N, B_g, C_g, D_g.

The calculation of the *ep* cross section follows eq.(13), with the off-shell matrix element including quarks masses taken from [49] in its implementation in `CASCADE` [55]. In addition to the boson gluon fusion process, also valence quark initiated $\gamma q \to q$ processes are included, with the valence quarks taken from [56].

2.7 Diffractive PDFs

In this section the diffractive process is briefly described. It was observed at HERA that about 10% of deep inelastic interactions are diffractive leading to events in which the interacting proton stays intact ($ep \to eXp$). In the diffractive process the proton appears well separated from the rest of the hadronic final state by a large rapidity gap, otherwise the events look similar to normal deep inelastic events. This process is usually interpreted as the diffractive dissociation of the exchanged virtual photon to produce any hadronic final state system X with mass much smaller than W and the same net quantum numbers as the exchanged photon. Figure 6 illustrates the kinematic variables used to describe the inclusive diffractive DIS process. For this, the proton vertex factorisation approach is assumed such that the diffractive DIS is mediated by the exchange of hard Pomeron and a secondary Reggeon. The factorisable pomeron picture has proved remarkably successful for the description of most of these data.

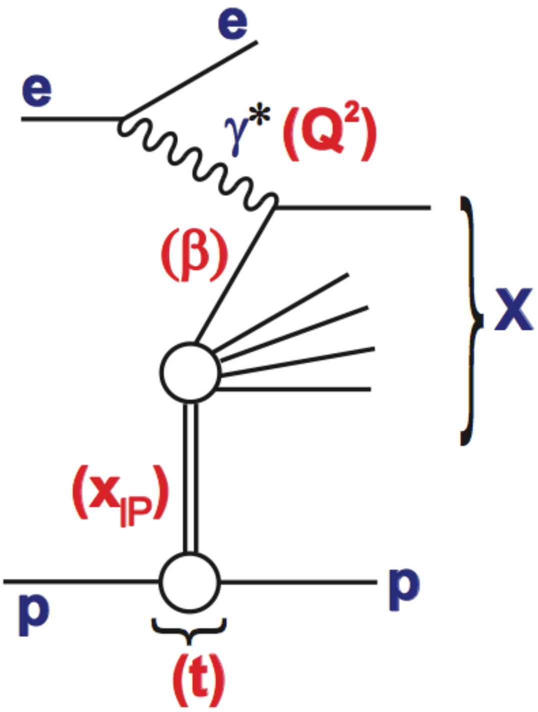


Figure 6: Schematic diagram of the kinematic variables used to describe the inclusive diffractive DIS process.

In addition to x, Q^2 and the squared four-momentum transfer t (the undetected momentum transfer to the proton system), the mass M_X of the diffractively produced final state provides a further degree of freedom. In practice, the variable M_X is often replaced by β,

$$\beta = \frac{Q^2}{M_X^2 + Q^2 - t}. \tag{20}$$

In models based on a factorisable pomeron, β may be viewed as the fraction of the pomeron longitudinal momentum which is carried by the struck parton, $x = \beta IP$.

2.7.1 Cross-section

As for the inclusive case, the diffractive cross-section can be expressed as:

$$\frac{d\sigma}{d\beta\, dQ^2 d\xi\, dt} = \frac{2\pi\alpha^2}{\beta Q^4}\left(1+(1-y)^2\right)\overline{\sigma}^{D(4)}(\beta, Q^2, \xi, t) \tag{21}$$

where the “reduced cross-section” , $\overline{\sigma}$, is defined as

$$\overline{\sigma}^{D(4)} = F_2^{D(4)} - \frac{y^2}{1+(1-y)^2} F_L^{D(4)} = F_T^{D(4)} + \frac{2(1-y)}{1+(1-y)^2} F_L^{D(4)} \tag{22}$$

The dimension of $F_k^{D(4)}(\beta, Q^2, \xi, t)$ is GeV^{-2} and thus the quantities integrated over t.

$$F_k^{D(3)}(\beta, Q^2, \xi) \equiv \int_{t_{\min}}^{t_{\max}} dt F_k^{D(4)}(\beta, Q^2, \xi, t) \tag{23}$$

are dimensionless. Maximum kinematically allowed value of t reads as

$$t_{\mathrm{MAX}} = -\frac{\xi^2 m_p^2 + p_\perp^2}{1-\xi} \approx -\frac{\xi^2}{1-\xi} m_p^2 \tag{24}$$

where m_p is the proton mass. As $x = \xi\beta$ we can normalize to the standard DIS formula

$$\frac{d\sigma}{d\beta\, dQ^2\, d\xi\, dt} = \frac{2\pi\alpha^2}{x\, Q^4}\left(1+(1-y)^2\right)\xi\overline{\sigma}^{D(4)}(\beta, Q^2, \xi, t) \tag{25}$$

which upon integration over t reads

$$\frac{d\sigma}{d\beta\, dQ^2\, d\xi} = \frac{2\pi\alpha^2}{xQ^4}\left(1+(1-y)^2\right)\xi\overline{\sigma}^{D(3)}(\beta, Q^2, \xi). \tag{26}$$

The diffractive structure functions can be expressed as convolutions of the calculable coefficient functions with diffractive quark and gluon distribution functions, which in general depend on all of x, Q^2, β, t.

2.7.2 Regge factorization

For a better description of data, a contribution from a secondary Reggeon, $I\!R$, is included, hence

$$F_k^{D(4)}(\beta, Q^2, \xi, t) = \sum_{X=I\!P, I\!R} \phi_X(\xi, t)\, F_k^X(\beta, Q^2) \tag{27}$$

or

$$F_k^{D(3)}(\beta, Q^2, \xi) = \sum_{X=I\!P, I\!R} \Phi_X(\xi)\, F_k^X(\beta, Q^2) \tag{28}$$

where

$$\Phi_X(\xi) = \int\limits_{t_{\min}}^{t_{\max}} dt\, \phi_X(\xi, t)\,. \tag{29}$$

Parametrization of the fluxes follows

$$\phi_X(\xi, t) = \frac{A_X\, e^{b_X t}}{\xi^{2\alpha_X(t)-1}} \tag{30a}$$

where

$$\alpha_X(t) = \alpha_X(0) + \alpha'_X t\,. \tag{30b}$$

$F_k^{I\!R}(\beta, Q^2)$ are taken as those of the pion.

3 Methodology for PDF fits

In this section the main fit formalism is presented in detail with various options as implemented in `Thereafter` İt consists from different functional forms used to parametrise the PDFs at the starting scale (section 3.1, various χ^2 definitions and representations (section 3.2, which is passed to the `MINUIT` package. The treatment of the experimental uncertainties is described in section 3.3. In addition a different approach to PDF studies based on the reweighing techniques is described in section 4.

3.1 PDF Parameterisation

Different parametrisation for PDFs at the starting scale present in `Thereafter` are described in this section. It starts from various standard functional forms (3.1.1), it follows with the bi-log-normal functional form (3.1.2), and it extends to more exotic forms based on generalised polynomial such as Chebyshev (3.1.3).

3.1.1 Standard Functional form

Through standard functional form it is understood a simple polynomial that interpolates between the low and high x regions:

$$xf(x) = Ax^B(1-x)^C P_i(x), \tag{31}$$

We identify few standard forms commonly used by PDF groups.

CTEQ style

The notation used throughout this text reflects the notation used in the code.

$$xf(x) = a_0 x^{(a_1+n)}(1-x)^{a_2} e^{a_3 x}(1+e^{a_4 x}+e^{a_5 x^2}), \tag{32}$$

HERAPDF style

The parametrised PDFs at HERA are the valence distributions xu_v and xd_v, the gluon distribution xg, and the u-type and d-type $x\bar{U}$, $x\bar{D}$, where $x\bar{U} = x\bar{u}$, $x\bar{D} = x\bar{d} + x\bar{s}$. The following standard functional form is used to parametrise them

$$xf(x) = Ax^B(1-x)^C(1+Dx+Ex^2), \tag{33}$$

where the normalisation parameters, A_{uv}, A_{dv}, A_g, are constrained by the QCD sum-rules, such that the counting and momentum conservation are preserved. The B parameters $B_{\bar{U}}$ and $B_{\bar{D}}$ are set equal, $B_{\bar{U}} = B_{\bar{D}}$, such that there is a single B parameter for the sea distributions. The strange quark distribution is already present at the starting scale and it is assumed here that $x\bar{s} = f_s x\bar{D}$ at Q_0^2. The strange fraction is chosen to be $f_s = 0.31$ which is consistent with determinations of this fraction using neutrino induced di-muon production. In addition, to ensure that $x\bar{u} \to x\bar{d}$ as $x \to 0$, $A_{\bar{U}} = A_{\bar{D}}(1-f_s)$. The D and E are introduced one by one until no further improvement in χ^2 is found. For the case when adding more precision data in the fit, as when adding HERA II data, this allows then for use of a more flexible parametrisation for the gluon and valence especially. The best fit results in a total of 10 free parameters when performing fits to solely HERA I data (fits are referred then ro as HERAPDF1.0), and of 13 free parameters when adding preliminary HERA II data on top (fits are referred then to as HERAPDF1.5).

An assumption that the strange sea quark density follows the same shape as the down quark density is used in HERAPDF fits. However, inspired by the HERMES analysis [57] which measure semi-inclusive production of the strange mesons, the following functional form is provided in the `Thereafter` to parametrise the strange density at the starting scale:

$$xs(x) = f_s \frac{1}{1 + \tanh(-(x - x_{hs}h_{hr}))}, \tag{34}$$

with $x_{hs} = 0.07$ and $h_{hr} = 20$, corresponding to a sharp turn on of the strange density at $x \sim 0.07$.

Strange style

For the studies of the strange quark sea density, the parametrisation of $x\bar{D}$ is replaced by a sum of $x\bar{d}(x) + x\bar{s}(x)$ parametrised densities:

$$xs(x) = \frac{fs}{1 - fs} A_{\bar{d}} x_s^B (1 - x)_s^C. \tag{35}$$

Flexible style

Flexible style is the extension of the "HERAPDF style" by allowing extra 2 free parameters for every PDF distribution, namely the D and E parameters for the medium x region. It can be used to study the data sensitivity to PDFs. The total number of free parameters is therefore 22.

Strange quark PDF is the least determined PDF from the l

3.1.2 Bi-Log-Normal Functional Form

A bi-log-normal distribution is proposed by [58] to parametrise the x-dependence of the parton density function of the proton. This new parametrisation is motivated by arguments of multi-particle statistics. This function can be regarded as a generalisation of parametrisation commonly used by global fit groups. The following parametrisation as general ansatz is proposed:

$$xf(x) = x^{p - b\log(x)}(1 - x)^{q - \log(1-x)}. \tag{36}$$

In order to satisfy the QCD sum rules this parametric form requires numerical integration.

3.1.3 Chebyshev Polynomial Functional Form

A flexible Chebyshev polynomials based parametrisation is used for the gluon and sea densities. The polynomials use $\log x$ as an argument to emphasize the low x behavior. The parametrisation is valid for $x > x_{min} = 1.7 \times 10^{-5}$. The PDFs are multiplied by $1 - x$ to ensure that they vanish as $x \to 1$. The resulting parametrisation form is

$$xg(x) = A_g (1 - x) \sum_{i=0}^{N_g - 1} A_{g_i} T_i \left(-\frac{2 \log x - \log x_{min}}{\log x_{min}} \right), \tag{37}$$

$$xS(x) = (1 - x) \sum_{i=0}^{N_S - 1} A_{S_i} T_i \left(-\frac{2 \log x - \log x_{min}}{\log x_{min}} \right). \tag{38}$$

Here the sum over i runs up to $N_{g,S} = 15$ order Chebyshev polynomials of the first type T_i for the gluon, g, and sea-quark, S, density, respectively. The normalisation A_g is given by the momentum sum rule. The advantages of the parametrisation given by equations 37,38 is that momentum sum rule can be evaluated analytically and already for $N \geq 5$ the fit quality is similar to a standard Regge-inspired parametrisation with a similar number of parameters.

3.1.4 Diffractive parametrisation Functional Form

Pomeron parametrisation

The Pomeron is parametrised at the initial Q_0^2 in terms of two singlet distributions, f_g and f_+.

$$\frac{d}{dt} f_+ = \frac{\alpha_s}{2\pi}\left[\mathcal{P}_{\mathrm{FF}} f_+ + \mathcal{P}_{\mathrm{FG}} f_g\right] \tag{39a}$$

$$\frac{d}{dt} f_g = \frac{\alpha_s}{2\pi}\left[\mathcal{P}_{\mathrm{GF}} f_+ + \mathcal{P}_{\mathrm{GG}} f_g\right] \tag{39b}$$

As $I\!P$ is neutral, $f_q = f_{\bar{q}}$ for each flavour q. Assuming that all light quark PDFs are equal

$$f_d = f_u = f_s\,, \tag{40}$$

we have

$$f_{q-} \equiv 0 \tag{41a}$$

$$f_{q+} \equiv 2 f_q \tag{41b}$$

At $n_f = 3$

$$f_{q+} = f_+/3,\;\; q = d, u, s\,. \tag{42}$$

i.e.

$$\tilde{f}_{q+} \equiv f_{q+} - \frac{1}{n_f} f_+ = 0,\;\; \text{for } q = d, u, s\,. \tag{43}$$

This gives all PDFs for the FFNS, while for VFNS f_{h+} for $h = c, b, t$ are generated dynamically above the respective transition scales Q_h^2. Hence at $n_f > 3$ the singlet has contributions from the heavy quarks and we get non-trivial nonsinglet distributions $\tilde{f}_{h+}$ satisfying

$$\frac{d}{dt}\tilde{f}_{h+} = \frac{\alpha_s}{2\pi}\mathcal{P}_{(+)}\tilde{f}_{h+} \tag{44}$$

Parametrisation at Q_0^2

Full PDFs are given in analogy to Eq. 28

$$f_k^{D(3)}(\beta, Q^2, \xi) = \hat{\Phi}_{I\!P}(\xi)\, f_k^{I\!P}(\beta, Q^2) + \Phi_{I\!R}(\xi)\, f_k^{I\!R}(\beta, Q^2) \tag{45}$$

where $\hat{\Phi}_{I\!P} \equiv \Phi_{I\!P}/A_{I\!P}$, with the fluxes given by Eq. 29 and Eq. 30.

The Pomeron PDFs are parametrised as

$$f_N^{I\!P} = A_1^{(N)} x^{A_2^{(N)}} (1-x)^{A_3^{(N)}}\, \exp\left(-\frac{d}{1.00001 - x}\right), \tag{46}$$

where the ‘dumping factor’ d is taken as 0.01 or 0.001. N = G for gluon and N = S for ‘singlet’: $f_{\mathrm{S}} \equiv f_+(n_f = 3)$, cf. Eq. 42.

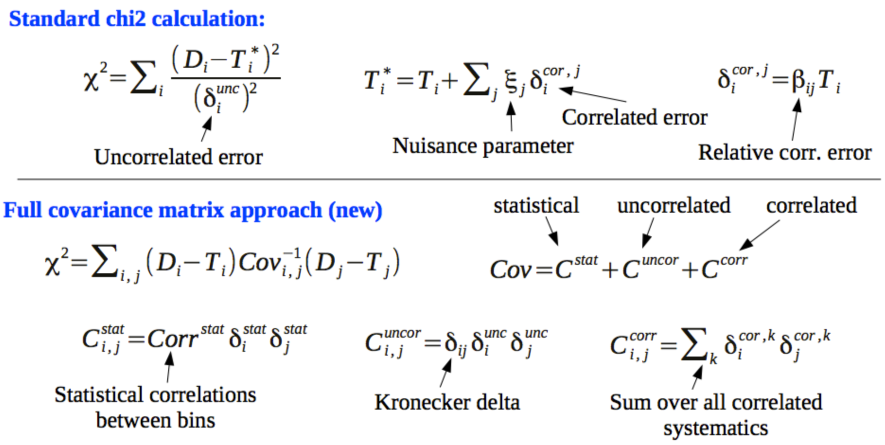

Standard chi2 calculation:
$\chi^2=\sum_i \frac{(D_i-T_i^*)^2}{(\delta_i^{unc})^2}$
Uncorrelated error
$T_i^*=T_i+\sum_j \xi_j \delta_i^{cor,j}$
Correlated error
Nuisance parameter
$\delta_i^{cor,j}=\beta_{ij} T_i$
Relative corr. error
Full covariance matrix approach (new)
$\chi^2=\sum_{i,j}(D_i-T_i)Cov^{-1}_{i,j}(D_j-T_j)$
statistical
uncorrelated
correlated
$Cov=C^{stat}+C^{uncor}+C^{corr}$
$C^{stat}_{i,j}=Corr^{stat}\delta_i^{stat}\delta_j^{stat}$
Statistical correlations between bins
$C^{uncor}_{i,j}=\delta_{ij}\delta_i^{unc}\delta_j^{unc}$
Kronecker delta
$C^{corr}_{i,j}=\sum_k \delta_i^{cor,k}\delta_j^{cor,k}$
Sum over all correlated systematics

Figure 7: Various χ^2 representations in Thereafter .

3.2 Chisquare Definition

In this section various forms of χ^2 are described, based on nuisance parameters or covariance matrix. A schematic picture of χ^2 definitions is displayed in Fig. 7 The description starts with most simple cases and it extends to more evolved form that takes into account the possible biases arising from low statistics data.

3.2.1 Using Nuisance Parameters

In this subsection the focus is on the χ^2 using nuisance parameters. Different variants are discussed.

Simple and Scaled Form

For a single data set, the χ^2 function can be defined in a simple form

$$\chi^2_{\rm exp}(\boldsymbol{m},\boldsymbol{b}) = \sum_i \frac{\left[m^i - \sum_j \gamma^i_j m^i b_j - \mu^i\right]^2}{\left(\delta_{i,\rm stat} m^i\right)^2 + \left(\delta_{i,\rm uncor}\, m^i\right)^2} + \sum_j b_j^2. \tag{47}$$

or more evolved as [59],

$$\chi^2_{\rm exp}(\boldsymbol{m},\boldsymbol{b}) = \sum_i \frac{\left[m^i - \sum_j \gamma^i_j m^i b_j - \mu^i\right]^2}{\delta^2_{i,\rm stat}\mu^i\left(m^i - \sum_j \gamma^i_j m^i b_j\right) + \left(\delta_{i,\rm uncor}\, m^i\right)^2} + \sum_j b_j^2. \tag{48}$$

Here μ^i is the measured central value at a point i with relative statistical $\delta_{i,stat}$ and relative uncorrelated systematic uncertainty $\delta_{i,unc}$. Further, β_j denotes a nuisance parameter for a correlated systematic error source of type j with an uncertainty while γ^i_j quantifies the sensitivity of the measurement μ^i at the point i to the systematic source j. The function $\chi^2_{\rm exp}$ depends on the set of underlying physical quantities m^i (denoted as the vector $\boldsymbol{m}$) and the set of systematic uncertainties b_j ($\boldsymbol{b}$). This definition of the χ^2 function takes into account that systematic uncertainties are proportional to the central values (multiplicative errors), whereas the statistical errors scale with the square roots of the expected number of events. Other scaling properties for the statistical and uncorrelated systematic uncertainties are discussed later.

In the case of off-diagonal statistical uncertainties, the χ^2 function is

$$\chi^2_{\mathrm{exp}}(\boldsymbol{m},\boldsymbol{b}) = \sum_{ij}\left(m^i - \sum_l \Gamma^i_l(m^i) b_l - \mu^i\right) C^{-1}_{\mathrm{stat.}\ ij}(m^i, m^j)\left(m^j - \sum_l \Gamma^j_l(m^j) b_l - \mu^j\right) + \sum_l b_l^2 \,. \quad (49)$$

Here the scaling properties of the correlated systematic uncertainties Γ^i_j and of the covariance matrix $C_{\mathrm{stat.}\ ij}$ are expresses as a dependence on m_i and the dependence of Δ_{stat} on b_j is ignored.

Eq. 49 allows for two methods for fast determination of the minimum, without need to include the formal nuisance parameters corresponding to the systematic error sources into the minuit minimisation. In the first method, the minimisation vs. b_j is used to define covariance matrix for the systematic uncertainties which is determined as

$$C_{\mathrm{syst}\ ij} = \sum_l \Gamma^i_l \Gamma^j_l \,. \quad (50)$$

The total covariance matrix is given by the sum of the statistical and systematic covariance matrices

$$C_{\mathrm{tot}\ ij} = C_{\mathrm{stat}\ ij} + C_{\mathrm{syst}\ ij} \,, \quad (51)$$

and the χ^2 function takes a form

$$\chi^2(\boldsymbol{m}) = \sum_{ij} (m^i - \mu^i) C^{-1}_{\mathrm{tot}\ ij} (m^j - \mu^j) \,. \quad (52)$$

The second methods is used to determine optimal shifts of the nuisance parameters at each iteration. The shifts are given by minimising Eq. 49 vs. b_l which leads to a system of linear equations

$$\sum_k \sum_{ij} C^{-1}_{\mathrm{stat}\ ij} \Gamma^i_l \Gamma^j_k \cdot b_k = \sum_{ij} C^{-1}_{\mathrm{stat}\ ij} \Gamma^i_l (m^j - \mu^j) \,, \quad (53)$$

where $1 \leq l \leq N_{\mathrm{syst}}$, the total number of correlated systematic uncertainties.

Finally the nuisance parameters $\boldsymbol{b}$ can be excluded from the χ^2 minimisation. In this case, which is referred to as an Offset method, the minimum is determined for their values set to zero while uncertainties on the parameters $\boldsymbol{p}$ are determined by shifting each nuisance parameter b_l by ± 1. The total covariance matrix for parameters p^i is determined as

$$C^{\mathrm{offset}}_{\mathrm{par}\ ij} = \sum_{l=1}^{N_{syst}} \Delta p^i_l \Delta p^j_l \,, \quad (54)$$

where $\Delta p^i_l = 0.5(p^i(b_l = +1) - p^i(b_l = -1))$ and the quality of the fit is estimated by fixing $\boldsymbol{p}$ to the value at the minimum and minimising with respect to $\boldsymbol{b}$

Finally, all three approaches can be combined together. For example, only some of the systematic uncertainties can be treated using the matrix method while others can be treated using the hessian method. In this case, the covariance matrix C_{syst} is build using the corresponding sub-set of systematic sources and C_{stat} is replaced by $C_{\mathrm{stat}} + C_{\mathrm{syst}}$ in Eq. 49. Similarly, some of the systematic uncertainties can be treated using offset method and then $C^{\mathrm{total}}_{\mathrm{par}} = C^{\mathrm{hessian}}_{\mathrm{par}} + C^{\mathrm{offset}}_{\mathrm{par}}$ where offset and hessian covariance matrices are calculated using corresponding systematic error sources.

Bias corrections

The correlated and uncorrelated systematic uncertainties can be treated as additive, $\Gamma^i_l(m^i) = \gamma^i_l \mu^i$ or multiplicative, $\Gamma^i_l(m^i) = \gamma^i_l m^i$. The LogNormal treatment in which $\mu^i + \sum_l \Gamma^i_j b_l$ is replaced by $\mu^i \prod_l \exp(\gamma^i_j b_l)$ is foreseen for the next release of the `HERAFitter`.

The statistical uncertainties can be treated as additive, $\Delta^i(m^i) = \delta^i \mu^i$ and as Poisson, $\Delta^i(m^i) = \delta^i \sqrt{\mu^i m^i}$. More complex scaling from Eq. 48, which depends on shifts of b_j, is implemented using an iterative approach: for the first iteration $b_l = 0$ is used to determine values of b_l which are then applied in the second iteration. Statistical covariance matrix is scaled in a similar manner. In this case the correlation matrix is assumed to be fixed, the diagonal elements are updated using the prescription describe above and the covariance matrix is rescaled accordingly.

The modifications of the covariance matrix at each iteration of the minuit minimisation may lead to systematic biases. There are two approaches to avoid these biases. In the first approach the covariance matrix is calculated using the expected values at the first iteration of the minimisation and kept fixed to these values for further iterations. This method requires several repetitions of the minimisation, to ensure that values close to optimal are obtained already at the first iteration. The second method modifies the χ^2 function by adding a term corresponding to non-constant value of the covariance matrix:

$$\chi^2_{\log} = 2 \log \frac{\Delta^i(m^i)}{\Delta^i(\mu^i)} \tag{55}$$

3.2.2 HERAFitter implementation

The form of the χ^2 function and the scaling properties of the uncertainties are controlled globally by the CHI2SettingsName and Chi2Settings variables and individually using ":" modifiers. The global scaling properties of the uncertainties are described in Table 1. The global form of the χ^2 function is defined by the CorChi2Type parameter, see Table 2.

The default behavior can be changed for each correlated systematic source by ":" modifiers. They are described in Table 3. The modifiers should appear at the end of the systematic source name, e.g. 'H3:M'. Several modifiers can be used, e.g. 'H3:M:C'.

The names of systematic error sources are read first from the ListOfSources variable of the &Systematics namelist, located in the steering.txt file. Next the names are read from the data files following the sequence given by the InputFileNames list. The properties of each systematic error source are defined by its first occurrence. That means that if, for example, 'H3:M:C' is defined in the ListOfSources variable, the source 'H3' is treated as multiplicative and using covariance matrix approach regardless definitions in the data files. If, however, ListOfSources defines a source without any modifiers, e.g. 'H3', the default treatment, following the Chi2Settings variable is enforced for this source. Thus the ListOfSources variable is a convenient way to modify behavior of the correlated systematic sources.

The shifts of the systematic sources are reported in the Results.txt file. The uncertainty on the shift is however estimated only approximately, neglecting the correlation with the theory parameters. An accurate determination of the uncertainty can be achieved by using the toy MC method (see section 3.3.3) or by using ':E' modifier. In the later case the systematic source is treated using the `Minuit` minimisation. Note, however, that this approach can slow the minimisation convergence considerably.

3.3 Treatment of the Experimental Uncertainties

3.3.1 Correlated errors and χ^2

Results of a measurement can be modelled as (see eg. [60, 61])

$$m_n = t_n(a) + r_n\sigma_n + \sum_{\mu=1}^{K} s_\mu b_{n\mu}\ , \quad n = 1, \ldots, N \tag{56}$$

Option	StatScale	UncorSysScale	CorSysScale	Scaling rule
Poisson	+	+	−	$\sqrt{m^i\mu^i}$
Linear	−	+	+	m^i
NoRescale	+	+	+	μ^i
LogNorm	Reserved, not implemented			

Table 1: Global scaling rules for statistical, uncorrelated and correlated systematic uncertainties. The scaling rule is given with respect to corresponding relative uncertainty. E.g. for the Poisson statistical uncertainty the absolute statistical uncertainty is $\Delta_i = \delta_{i,\mathrm{stat}}\sqrt{m^i\mu^i}$

CorChi2Type value	Description
Hessian	Use nuisance parameters.
Matrix	Use covariance matrix.
Offset	Use offset method

Table 2: Possible values of the CorChi2Type parameter which defines treatment of the correlated systematic uncertainties.

Modifier	Description
Scaling properties	
:M	Multiplicative scaling, m^i
:A	Additive scaling, μ^i
:P	Poisson scaling, $\sqrt{m^i\mu^i}$
χ^2 treatment	
:N	Nuisance parameter treatment
:C	Covariance matrix treatment
:O	Offset method treatment
:E	Nuisance parameter, included in Minuit (“External”)

Table 3: Modifiers for correlated systematic uncertainty sources.

where
m_n is the value measured for the n-th data point,
$t_n(\boldsymbol{a})$ is true (theoretical) value depending on parameters $\boldsymbol{a} = (a_1, \ldots, a_M)$,
σ_n is the uncorrelated error,
$b_{n\mu}$ are the errors from the μ-th correlated error source,
r_n and s_μ are random variables fluctuating around 0 with unit dispersion.

First, we assume that all r_n are uncorrelated with s_μ, mutually independent and normally distributed,

$$\rho(r) = \frac{e^{-r^2/2}}{\sqrt{2\pi}} . \tag{57}$$

In the following we will use scaled variables

$$x_i \equiv \frac{m_i - t_i}{\sigma_i} , \tag{58a}$$

$$\beta_{i\mu} \equiv \frac{b_{i\mu}}{\sigma_i} . \tag{58b}$$

Keeping $\boldsymbol{s}$ fixed we get the probability density of measurements,

$$dp(\boldsymbol{m}|\boldsymbol{s}) = (2\pi)^{-N/2} \, e^{-\chi_1^2(\boldsymbol{s})/2} \, d^N x , \tag{59}$$

where

$$\chi_1^2(\boldsymbol{s}) = \sum_{n=1}^{N} \left(x_n - \sum_\mu \beta_{n\mu} s_\mu \right)^2 \equiv (\boldsymbol{x} - \boldsymbol{\beta s})^2 , \tag{60}$$

Further, taking into account the probability distribution of the correlated error sources, $p(\boldsymbol{s}) \, d^K s$, we have

$$p(\boldsymbol{m}, \boldsymbol{s}) = p(\boldsymbol{s}) \, p(\boldsymbol{m}|\boldsymbol{s}) . \tag{61}$$

Assuming again the uncorrelated normal distribution,

$$p(\boldsymbol{s}) = \prod_{\mu=1}^{K} \frac{e^{-s_\mu^2/2}}{\sqrt{2\pi}} , \tag{62}$$

we get

$$dp(\boldsymbol{m}, \boldsymbol{s}) = (2\pi)^{-(N+K)/2} \, e^{-\chi_c^2(\boldsymbol{s})/2} \, d^N x \, d^K s , \tag{63}$$

with

$$\chi_c^2(\boldsymbol{s}) = (\boldsymbol{x} - \boldsymbol{\beta s})^2 + \boldsymbol{s}^2 . \tag{64}$$

This quadratic form in $\boldsymbol{s}$ allows for analytical integration of Eq 63 resulting in

$$p(\boldsymbol{m}) \propto e^{-\chi^2/2} , \tag{65}$$

where

$$\chi^2 = \boldsymbol{x}^{\mathrm{T}} \boldsymbol{A} \, \boldsymbol{x} \tag{66}$$

with $\boldsymbol{A}$ depending on $\boldsymbol{\beta}$ only (see eg. [60] Appendix B).

This is e.g. the CTEQ approach described in [60]. It is worth noting that the solution Eq. 66 for χ^2 can be obtained by minimizing $\chi_c^2(\boldsymbol{s})$ of Eq. 64 wrt. $\boldsymbol{s}$.

3.3.2 The Offset method

In the Offset method presented here we assume that $\boldsymbol{s}$ is fixed, and we find the best theoretical model by minimizing χ_1^2 wrt. to $\boldsymbol{a}$. Hence the fitted parameters become functions of $\boldsymbol{s}$. We do not impose any particular statistical properties on $\boldsymbol{s}$ and we take $\boldsymbol{a}(\boldsymbol{s} = 0)$ as the ultimate fit result for the theory parameters. The dependence on $\boldsymbol{s}$ is, however, used to determine the full error matrix of $\boldsymbol{a}$ (cf. [62]).

The full covariance matrix V reads

$$V = V^{(\mathrm{unc})} + V^{(\mathrm{cor})} \tag{67}$$

For each $\boldsymbol{s}$ we find the parameters $\boldsymbol{a}(\boldsymbol{s})$ by minimising $\chi_1^2(\boldsymbol{s})$, which results in $V^{(\mathrm{unc})}(\boldsymbol{s}) = M^{-1}(\boldsymbol{s})$ where

$$M_{jk}(\boldsymbol{s}) = \frac{1}{2} \frac{\partial^2 \chi_1(\boldsymbol{s})^2}{\partial a_j \partial a_k}\bigg|_{\boldsymbol{a}=\boldsymbol{a}(\boldsymbol{s})} . \tag{68}$$

The dependence of M on $\boldsymbol{s}$ is considered to be a higher order correction and we take $V^{(\mathrm{unc})} = M^{-1}(0)$.

Within linear approximation to the error propagation

$$V_{jk}^{(\mathrm{cor})} = \sum_{\mu} \frac{da_j}{ds_\mu} \frac{da_k}{ds_\mu} \tag{69}$$

and we calculate the derivatives as

$$\frac{da_j}{ds_\mu} \approx \frac{a_j(s_\mu = \epsilon) - a_j(s_\mu = -\epsilon)}{2\epsilon} \tag{70}$$

with $\boldsymbol{a}(s_\mu = \epsilon)$ resulting from fits to the data shifted by $\epsilon b_{n\mu}$.

In the code we use $\epsilon = 1$, i.e. one standard deviation of the correlated error source which, in the ideal statistical limit, corresponds to $\Delta\chi_1^2 = 1$. On the other hand, within the leading approximation, the value of ϵ is irrelevant.

If another error definition, $\Delta\chi_1^2 = \lambda$, is adopted[2] then the full covariance matrix, V, must be scaled by λ.

Implementation via `Thereafter`

The Offset method is turned on by setting `CHI2Style = 'Offset'` By default all fits are run in a single job, each fit driven by initial parameters and Minuit commands read from `minuit.in.txt`. Two optional parameters can be set in the `CSOffset` NAMELIST, e.g.

```
&CSOffset
 CorSysIndex  =  0
 UsePrevFit = 1
&End
```

Defaults are set in `read_steer.f` and the `CSOffset NAMELIST` is read only when the Offset method is active. Setting `CorSysIndex` to any value $\in [-K, K]$ restricts the job to a single fit to data shifted (down or up) by a corresponding correlated error source. `CorSysIndex =` 0 corresponds to the central fit. If `CorSysIndex > NSYSMAX` then all the fits are performed. The best way to perform all fits in a single run is to not specify `CorSysIndex` at all (Default: `CorSysIndex = NSYSMAX+1`)

The parameter `UsePrevFit` determines how to use results of previous fits, if such results are present in the `output` folder.

[2] E.g. Jon Pumplin uses $\lambda = 5$, cf. `minuit/iterate.F`

0 — Do not use any previous fit results (Default)

1 — Use previously obtained parameters as starting values for the current fit. Read initial parameters from minuit.save_<CSI>.txt — e.g. minuit.save_001m.txt for CorSysIndex = −1. If the file does not exist and CorSysIndex ≠ 0 try to read minuit.save_0.txt.

2 — Do not perform the fit if a corresponding Results_<CSI>.txt file exists, otherwise switch to mode 1.

3.3.3 Monte Carlo Method

The PDF uncertainties can be estimated using a Monte Carlo technique [63, 64]. The method consists in preparing replicas of data sets by allowing the central values of the cross sections to fluctuate within their systematic and statistical uncertainties taking into account all point-to-point correlations. The preparation of the data is repeated for a large N (> 100 times) and for each of these replicas a NLO QCD fit is performed to extract the PDF set. The PDF central values and uncertainties are estimated using the means values and RMS over the replicas.

3.3.4 Implementation in Thereafter

The steering flags to activate the MC method are located in the steering.txt via:

```
&MCErrors

  lRAND    = False
  lRANDDATA = True
  ISeedMC = 123456
  ! --- Choose what distribution for the random number generator
  ! STATYPE (SYS_TYPE)  =   1  gauss
  ! STATYPE (SYS_TYPE)  =   2  uniform
  ! STATYPE (SYS_TYPE)  =   3  lognormal
  ! STATYPE (SYS_TYPE)  =   4  poisson (only for lRANDDATA = False !)
  STATYPE =  1
  SYSTYPE =  1
&End
```

To activate MC method for error estimation this is done by setting lRand = True . To use data (true, default) or theory (false) for the central values of the MC replica the the flag lRANDDATA is used. The seed for random number generator is selected via ISeedMC . The smearing of the uncertainties is treated differently for correlated or uncorrelated source and three distributions are supported for random number generators: Gauss, uniform, lognormal, and Poisson. If the flags are set to 0 then no smearing is produced.

3.3.5 Regularisation methods

Regularisation methods are aiming to study the parametrisation assumptions on PDFs. When more flexible parametrisation styles is used the shape of the PDFs must be constrained and various methods could be used. HERAFitter framework provides the means to study and compare various methods.

Data Driven Regularisation

This method was first applied by NNPDF group which uses redundant parametrisation and introduces a stopping criteria based on data. The data driven regularisation method splits data randomly into “fit” and “control” samples. The “fit” sample is used to determine PDF parameters. The χ^2 of this sample is observed to semi-monotonically decrease. The “control” sample is used to protect against over-fitting and for this sample the χ^2 will first decrease and then will start to increase due to fluctuation of the sample.

External Regularisation based on penalty term in χ^2

Another method to constrain the PDF shape is to simply apply a penalty term to the χ^2 function. One method is so called “length penalty” which selects PDF solutions with smoother shape in $W \approx Q\sqrt{\frac{1-x}{x}}$:

$$L = \int_{W_{min}}^{W_{max}} \sqrt{1 + \left(\frac{dxf(W)}{dW}\right)^2} dW \tag{71}$$

This method can be applied when using Chebyshev polynomials to parametrise PDFs. For more details, the reader is invited to look up the [65] reference. This is implemented in `Thereafter` via steering flags under Namelist `&Cheb` .

```
&Cheb
  ! Set following > 0 to turn on:
   NCHEBGLU = 0   ! number of parameters for the gluon (max 15)
   NCHEBSEA = 0   ! number of parameters for the sea   (max 15)

  ! Cheb. polynomial type: multiply by (1-x) (1) or not (0)
   ichebtypeGlu = 1
   ichebtypeSea = 1

  ! Starting point in x:
   chebxmin = 1.E-5

   ILENPDF  = 0   ! use pdf length constraint

  ! PDF length constraint strength for different PDFs:
   PDFLenWeight = 1., 1., 1., 1., 1.

  ! Range in W where length constraint is applied:
   WMNLen =  20.
   WMXLen = 320.
&end
```

For the case when using flexible parametrisation style, a χ^2 penalty term is applied to account for a deviation from a simple PDF parametrisation form such as:

$$\chi^2_{reg} = T \sum_f \left(\left(\frac{D_f}{\Delta_D}\right)^2 + \left(\frac{E_f}{\Delta_E}\right)^2 \right), \tag{72}$$

with $\Delta D = \Delta E = 100$, such that for large D and E the ratio will approach 1. The T is the regularisation parameters, such that for $T = 0$ there is no penalty term and for large T there is strong penalty. It is easily access via Namelist: `&ExtraMinimisationParameters` with the name for the name `'Temperature'` , which is the T from above description.

4 Bayesian Reweighting Technique

Bayesian reweighting of PDF sets is a way to include new data into an existing PDF set without actual carrying out a full-blown fitting procedure. It has been first suggested by Giele and Keller [63] and first pursued in practice by the NNPDF Collaboration [66, 67]. Watt and Thorne [68] proposed a scheme of how to implement the Bayesian reweighting technique also for PDF predictions based on central values with errors determined using the Hessian Eigenvector Method.

The `Thereafter` package allows to update any PDF that is either available as probability distribution (i.e. a lhapdf .LHgrid file in NNPDF format) or as PDF eigenvector set (i.e. any PDF set in lhapdf .LHgrid file format with errors determined using the Hessian Eigenvector Method). This enables the user to assess the impact of new data not only for the `HERAPDF` using the full-blown fit procedure but also for the other standard global PDF sets and allows to compare how the data impacts different PDFs.

The Bayesian Reweighting technique essentially uses PDF probability distributions as input, applies weights to these distributions based on how well the new data is described and outputs an updated PDF probability distribution. In the following paragraphs, firstly the construction of these PDF probability distributions is described, then the calculation of the weights to update the PDF probability distribution is introduced and lastly, the configuration of the module within the `Thereafter` framework is explained.

4.1 PDF probability distributions

PDF probability distributions are constructed as finite ensembles of N_{rep} parton distribution functions PDF_k, $\mathcal{E} = \{PDF_k, k = 1, ..., N_{\text{rep}}\}$. Observables $\mathcal{O}(\text{PDF})$ are conventionally calculated from the average of the predictions obtained from the ensemble:

$$\langle \mathcal{O}(\text{PDF}) \rangle = \frac{1}{N_{\text{rep}}} \sum_{k=1}^{N_{\text{rep}}} \mathcal{O}(\text{PDF}_k) \tag{73}$$

Their uncertainties are calculated as the standard deviation, defined as:

$$\sigma_{\mathcal{O}(\text{PDF})} = \sqrt{\frac{1}{N_{\text{rep}} - 1} \sum_{k=1}^{N_{\text{rep}}} (\mathcal{O}(\text{PDF}_k) - \langle \mathcal{O}(\text{PDF}) \rangle)^2} \tag{74}$$

While the standard PDF sets from the NNPDF collaboration are already available as ensembles of parton distribution functions, the PDF predictions of other PDF fitting groups need to be converted to PDF probability distributions. This is possible provided that the PDF sets have associated uncertainties that can be used to create replicas of the central PDF set with random variations that lie within the uncertainties.

In the case of uncertainties provided by standard Hessian eigenvectors error sets, this can be easily achieved by creating the k-th random replica by introducing to the central PDF set, PDF_0, random fluctuations.

If the PDF eigenvectors are asymmetric, that is they come in pairs of negative and positive PDF error sets, corresponding to negative and positive deviations from the central value, these random fluctuations are created by drawing a random number R_{jk} and adding, depending on the sign of the random number, the difference of the positive or respectively negative PDF of the j-th PDF eigenvector pair from the central value, scaled by the absolute value of the random number:

$$\mathrm{PDF}_k = \mathrm{PDF}_0 + \sum_{j=0}^{n} \left[\mathrm{PDF}_j^{\pm} - \mathrm{PDF}_0\right] |R_{jk}| \tag{75}$$

Here, k denotes the number of the random replica and runs from $k = 1, ..., N_{\mathrm{rep}}$; j denotes the eigenvector pair and runs from $j = 1, ..., n$, where n is the number of eigenvectors, e.g. $n = 20$ for MSTW08.

In case, the Hessian eigenvectors are symmetrised and only one error set is given per eigenvector, the above prescription simplifies to:

$$\mathrm{PDF}_k = \mathrm{PDF}_0 + \sum_{j=0}^{n} \left[\mathrm{PDF}_j - \mathrm{PDF}_0\right] R_{jk} \tag{76}$$

4.2 Bayesian Reweighting of PDF sets

Once PDF probability distributions are available as inputs, they can be updated to incorporate the new data. This is achieved by applying weights to the PDF probability distributions such that the prediction for observable $\langle O(\mathrm{PDF})\rangle$ from equation 73 changes to:

$$\langle O^{\mathrm{new}}(\mathrm{PDF})\rangle = \frac{1}{N_{\mathrm{rep}}} \sum_{k=1}^{N_{\mathrm{rep}}} w_k O(\mathrm{PDF}_k) \tag{77}$$

The weights w_k calculated are here according to:

$$w_k = \frac{(\chi_k^2)^{\frac{1}{2}(N_{\mathrm{data}}-1)} \exp^{-\frac{1}{2}\chi_k^2}}{\frac{1}{N_{\mathrm{rep}}} \sum_{k=1}^{N_{\mathrm{rep}}} (\chi_k^2)^{\frac{1}{2}(N_{\mathrm{data}}-1)} \exp^{-\frac{1}{2}\chi_k^2}}, \tag{78}$$

where N_{data} is the number of new data points, k denotes the specific replica for which the weights is calculated and χ_k^2 is between a given data point y_i and its theoretical prediction obtained with the k-th PDF replica:

$$\chi^2(y, \mathrm{PDF}_k) = \sum_{i,j=0}^{N_{\mathrm{data}}} (y_i - y_i(\mathrm{PDF}_k))\sigma_{ij}^{-1}(y_j - y_j(\mathrm{PDF}_k)) \tag{79}$$

The weighted PDF probability distribution can be turned into a new ensemble of PDF replicas, based on which predictions for any observable can be calculated. This new, reweighted PDF probability distribution commonly is chosen to be based upon a smaller number of PDF sets compared to the input PDF probability distribution, in order throw away those replicas that are incompatible with the data and to create a more light-weight PDF set.

4.3 Usage of the PDF reweighting in the `Thereafter` framework

The `Thereafter` allows to perform PDF reweighting for NNPDF-style PDF probability distributions as well as for PDF sets with Hessian PDF error Eigenvector sets.

This requires the `NNPDF reweight` and the `LHAPDF` modules to be installed, see sections 5.2 and 5.1.4. In the `Thereafter` steering files, the PDF reweighting needs to be switched on and the relevant parameters have to be set:

- **FLAGRW**: En/disable reweighting
- **RWPDFSET**: Name of the PDF set to be reweighted
- **RWDATA**: Arbitrary name for the data to be updated, used to create the names of the output PDF set and the directory
- **RWMETHOD**: Do the reweighting based on chi2 (method 1, where you read in `Thereafter` data files and theory predictions and calculate the chi2 based on them) or on data (method 2, where you have to provide an input text file with theoretical predictions – the input format of these will be explained below)
- **DORWONLY**: Disable the usual PDF fit, such that only the reweighting is done
- **RWREPLICAS**: Number of input replicas used for the PDF probability distributions (not applicable for NNPDF sets, since they come with a fixed number of replicas)
- **RWOUTREPLICAS**: Number of replicas in the output PDF set.

The setup of the module is such, that it is parsing the `Thereafter` steering file and from the specified settings creates a special reweighting steering file in the directory `input_steering` with the pattern `<RWPDFSET>_<RWDATA>_<RWMETHOD: chi2 or data>.in`.

In the output directory, a sub-directory is created for the output of the reweighting procedure. Its name pattern is: `output/<RWPDFSET>_<RWDATA>_<RWMETHOD: chi2 or data>/` and it will contain the following files:

- `<RWPDFSET>_<RWDATA>_<RWMETHOD: chi2 or data>_nRep<RWOUTREPLICAS>.LHgrid`: The output PDF probability distribution in form of an .LHgrid file, which allows easy usage.
- `whist-rw.eps`: Plot with the distributions of weights calculated for each replica. The meaning of this variable is further described in [66, 67].
- `palpha-rw.eps`: Plot with the probability for each replica to describe the data. The meaning of this variable is further described in [66, 67].
- `<RWPDFSET>_<RWREPLICAS>InputReplicas.LHgrid`: This is the PDF probability function that has been produced from the eigenvector PDF sets produced by the Hessian method (not applicable for NNPDF sets).

5 Program Manual

This section presents first the program installation instructions for various scenarios supported by the Thereafter platform. It follows with a basic user manual which is meant to guide the user in his analysis.

5.1 Program Installation Instructions

The Installation Instructions are dependent on which modules are activated via the configuration option.

5.1.1 Pre-requirements

The following packages are needed in order to build Thereafter package:

- QCDNUM [7] version at least `qcdnum-17-00-04`, can be found at
  `http://mbotje.web.cern.ch/mbotje/qcdnum/Site/QCDNUM17.html`
- `CERNLIB` libraries. Note that for `CERNLIB` one can use `/afs/` installation from CERN:
  `/afs/cern.ch/sw/lcg/external/cernlib/`

The Thereafter program has been tested on various platforms:
SL4, SL5 (32 and 64 bit), SL6 (64 bit), Ubuntu 10.10 (gcc and gfortran version 4.6.3), Mac OS (gcc and gfortran version 4.7.2)

5.1.2 Default Installation

- Specify `CERN_ROOT` and `QCDNUM_ROOT` variables such that
  `$CERN_ROOT/lib` and `$QCDNUM_ROOT/lib` point to the corresponding libraries
- Run:

  ```
  ./configure
  make
  make install
  ```

  After these commands are finished, the executable `bin/FitPDF` file should be installed
- Run a check:

  ```
  bin/FitPDF
  ```

5.1.3 Installation with APPLGRID

- Specify CERN_ROOT and QCDNUM_ROOT variables such that `$CERN_ROOT/lib` and `$QCDNUM_ROOT/lib` point to the corresponding libraries
- Make sure that `$PATH` and `$LD_LIBRARY_PATH` variables point to the APPLGRID environment.
- Run:

  ```
  ./configure --enable-applgrid
  make
  make install
  ```

  After these commands are finished, the executable `bin/FitPDF` file should be installed
- Run a check:

  ```
  bin/FitPDF
  ```

5.1.4 Installation with LHAPDF

Installation with LHAPDF requires the LHAPDF package, available online at: `http://lhapdf.hepforge.org/install`. Then

```
tar -xvzf lhapdf-v.r.p.tar.gz
cd lhapdf-v.r.p
./configure --prefix=/path/to/directory (and/or --enable-low-memory)
make
make install
cd /path/to/directory/share/lhapdf
mkdir PDFsets
```

Once installed, create the path that will be linked to the HERAFitter package. Specify LD_LIBRARY_PATH and LHAPATH variables such that they point to the corresponding libraries, and PDF sets location (where lhapdf tables are stored)

```
export LD_LIBRARY_PATH=/path/to/directory/lhapdf-v.r.p/lib:$LD_LIBRARY_PATH
export LHAPATH=/path/to/directory/share/lhapdf/PDFsets
```

5.2 Installation with PDF reweighting

Note: For installation allowing for PDF reweighting, the latest version of LHAPDF, lhapdf-5.8.7b2, should be installed.

- Make sure that `$LD_LIBRARY_PATH` includes the LHAPDF libraries.
- Run:

```
    ./configure --enable-lhapdf  --enable-nnpdfWeight
    make
    make install
```

After these commands are finished, the executable `bin/FitPDF` file should be installed

- Set `FLAGRW = True` in the steering file and change also the other parameters of the `&reweighting` namelist if needed.

- Run a check:

```
    bin/FitPDF
```

5.2.1 Installation with HATHOR

- Download Hathor from

  `http://www-zeuthen.desy.de/~moch/hathor/`

  and install it according to the instructions given there (requires `LHAPDF` library)

- Define a variable HATHOR_ROOT such that HATHOR_ROOT points to the directory of your Hathor installation

- Install the HERAFitter as described above but configuring it with the option ”–enable-hathor” before building it

5.2.2 Installation for TMD (uPDF) in high-energy factorisation (using CASCADE)

- Installation with TMD requires Cascade and Pythia generators, they can be downloaded from `http://cascade.hepforge.org/` and `https://pythia6.hepforge.org/` respectively.

  After installation of generator packages, the `CASCADE_ROOT` and `PYTHIA_ROOT` environment variables have be specified and point to the corresponding libraries. In DESY afs environment the Pr-installed versions of Cascade and Pythia can be used:

```
export CASCADE\_ROOT=/afs/desy.de/group/alliance/mcg/public/MCGenerators/cascade/2.2.04/\$SYSNAME
export PYTHIA\_ROOT=/afs/desy.de/group/alliance/mcg/public/MCGenerators/pythia6/425/\$SYSNAME}
```

  where `SYSNAME` = i586_rhel50 or similar.

- Run:

```
    ./configure --enable-uPDF
    make
    make install
```

- use steering and minuit input files from ”input_steering”:

```
   cp input-steering/steering.txt.kt-factorisation steering.txt
   cp input-steering/minuit.in.txt.kt-factorisation minuit.in.txt
   cp input-steering/steer-ep-CASCADE steer-ep
```

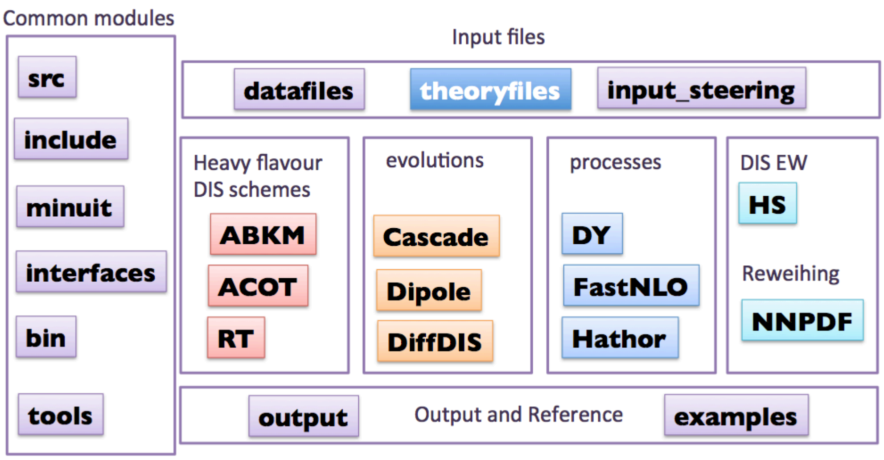


Figure 8: Schematic structure of the `Thereafter` program organisation in different modules.

- edit steering.txt:

```
\&CCFMFiles: give name for output grid file for uPDF
\&HERAFitter
TheoryType = 'uPDF3' ! 'DGLAP'  -- collinear evolution
                     ! 'DIPOLE' -- dipole model
                     ! 'uPDF'   -- un-integrated PDFs:
                     !uPDF1 fit with kernel ccfm-grid.dat file
                     !uPDF2 fit evolved uPDF, fit just normalisation
                     !uPDF3 fit using precalculated grid of sigma_hat
```

- run the program: bin/FitPDF
- plotting F_2 fit results:
  DrawResults will draw F_2 results.
  The uPDFs need to be plotted with an external package (currently not available).

5.3 User Manual

In this section a user manual is presented. The section starts with a general overview of the code organisation and it follows by a more detailed explanation of most used functions.

5.3.1 Code Organisation

A general diagram of available modules is illustrated in figure 8. The flow is depicted such that it follows the structure of the `Thereafter` .

In addition, an inventory list with short description of existing subroutines is presented in Table 4. Here we choose to enlist only the routines from the common target module to guide the user of available functionalities.

steerings	• steering.txt:	free PDF parameters to be varied by MINUIT
	• minui.in.txt:	main steering card
	• ewparam.txt:	setting ElectroWeak parameters, as well as masses
src	• main.f:	main program
	• read_steer.f:	access steer parameters from steering card
	• read_data.f:	reading the datatables and storing data information
	• init_theory.f:	initialising theory modules
	• dataset_tools.f:	allocating bin indices
	• error_logging.f:	error logging information
	• minuit_ini.f:	initialise minuit module
	• fcn.f:	passes to minuit the χ^2 to be minimize
	• pdf_param.f:	parametrisation of the PDFs at starting scale
	• sumrules.f:	PDF constraints at starting scale, such as QCD sum rules.
	• evolution.f:	evolution of PDFs
	• theory_dispatcher.f:	distribute calculation of theory prediction for a dataset
	• dis_sigma.f	calculate the DIS cross sections
	• GetChisquare.f	calculates the χ^2
	• GetCovChisquare.f	calculates the χ^2 using covariance matrix
	• GetPointScaledErrors.f	calculates the rescaled statistical, uncorrelated and constant errors
	• prep_corr.f	prepare systematic correlation matrix
	• systematics.f	build the matrix for systematic uncertainties and invert it
	• error_bands_pumplin.f	Hessian error calculations
	• mc_errors.f	MC method for creating replicas of data through smearing.
	• GetDiffDisXsection.f	calculate the diffractive DIS cross sections
	• FixModelParams.f	used for diffractive DIS cross sections
	• lhapdf_dum.f	(used only with ENABLE_LHPDF)
	• reweighting.f	main subroutine for PDF rewighting (used only with ENABLE_NNPDF)
	• nnpdfreweighting.f	main subroutine for NNPDF rewighting (used only with ENABLE_NNPDF)
	• dy_cc_sigma.f	calculate the DY cross sections (LO)
	• applgrids_dum.f	protective file against miss use of flags in steering
	• fappl_grid.cxx	(used only with ENABLE_APPLGRID)
	• applgrids.f	passing PDFs to APPLGRID (used only with ENABLE_APPLGRID)
	• pp_jets_applgrid.f	For DY process convolution for APPLGRID
	• ep_jets_fastnlo.f	Calculate *ep* jets cross sections
	• getncxskt.f	Access the NC cross sections grids for uPDFs
	• Getgridkt.f	Acess the grids for uPDFs
	• ttbar_hathor_dum.f	protective file against miss use of flags in steering
	• ttbar_hathor.f	(used only with ENABLE_HATHOR)
	• offset_fns.f	collects results from Offset method and stores them
	• g_offset.cc	file used for Offest method
	• matrix.cc	inversion of matrix as used for Offset method
	• FitPars_base.cc	file used for Offest method
	• FTNFitPars.cc	file used for Offest method
	• Xstring.cc	file used for Offest method
	• decor.cc	file used for Offest method
	• store_output.f	write the output
	• store_h1qcdfunc.f	store structure functions

Table 4: A list of main subroutines are listed with a short description of their function.

5.3.2 Steering files

The software behavior is controlled by three files with steering commands. These files have predefined names:

- `steering.txt` – controls main ”stable” (un-modified during minimisation) parameters. The file also contains names of data files to be fitted to, definition of kinematic cuts
- `minuit.in.txt` – controls minimisation parameters and minimisation strategy. Standard Minuit commands can be provided in this file
- `ewparam.txt` – controls electroweak parameters such as W and Z boson masses and CKM matrix parameters.

Steering.txt

Different options are activated via steering flags in the main steering file.

The format of the steering file follows standard ”namelist” conventions. Comments start with exclamation mark (similarly used for data file format). The following namelist blocks are encountered:

- `InFiles`: Namelist to control input data
- `InCorr`: Namelist to control statistical correlation files
- `Scales` (Optional): Namelist to modify renormalisation/factorisation scale
- `HeraFitter`: Main steering cards.
- `ExtraMinimisationParameters`: Namelist to add extra to minuit parameters.
- `Output`: Namelist that outputs steering cards
- `Cuts`: Namelist for process dependent cuts
- `MCErrors` (Optional):Namelist for MC errors steering cards
- `Cheb` (Optional): Chebyshev study namelist
- `Poly` (Optional): pure polynomial parameterisation for valence quarks
- `HQScale` (Optional): choose the factorisation scale for HQs
- `lhapdf` (Optional):LHAPDF steering card
- `reweighting` (Optional): reweighting steering cards

These namelist blocks are described in greater details in the User’s example 5.4.

Theory type:

here is a steering flag which defines the theory type via the chosen evolution. The following types are supported:

- `TheoryType = 'DGLAP'` as used for collinear evolution theories. For this type, it follows with another flag which sets the order of perturbative series in α_S: `Order`which can be leading order (LO), next-to-leading order (NLO) and when available NNLO.
- `TheoryType = 'DIPOLE'` as used for the dipole models;
- `TheoryType = 'uPDF'` as used for the un-integrated PDFs (with 3 variants)

Starting scale:

The evolution starting scale is set via flag `Q02`, commonly used below charm mass threshold, as imposed by QCDNUM.

Scheme type:

For the DIS process, Several schemes are available for heavy quark treatments via `HF_SCHEME`flag.

- VFNS (Variable Flavour Number Schemes):
  - RT-VFS schemes [from Robert Thorne], `HF_SCHEME = RT, RT OPT`, as well as the fast variants based on k-factors `RT FAST, RT OPT FAST`
  - `Zero Mass VFNS [qcdnum], ZM-VFNS`
  - ACOT (ACOT-Full, ACOT-ZM, S-ACOT-Chi) schemes [from Fred Olness], `HF_SCHEME = ACOT Full, ACOT Chi, ACOT ZM`, they are all based on k-factors.
- FFNS (Fixed Flavour Number Scheme)
  - via QCDNUM, `HF_SCHEME = FF`
  - via ABM (openqcdrad-1.6) [from Sergey Alekhin], `HF_SCHEME = FF ABM`

IMPORTANT to note if running with FFNS (nf=3):

- only neutral current DIS data should be used in FF scheme due to missing NLO coefficient functions in charged current (W+c) process, valence quarks in this case should to be fixed in minuit.in.txt file.
- In FF ABM implementation the charged current coefficients are available therefore valence parameters do not need to be fixed.
- $\alpha_s(Q^2)$ in FFNS is 3-flavour and recommended to be set to value of 0.105 such that is not too high at low energies
- the scale in FFNS is defined as $\mu^2 = Q^2 + 4m_h^2$ by default, can be changed in HQScale in `steering.txt`(scale variation in ABM not yet implemented)
- the pole mass definition for heavy quarks is set in ABM by default, the running mass definition [20] can be switched in by setting `HF_SCHEME = FF ABM RUNM` in `steering.txt`
  .

PDF parameterisation style:

There are various types of parametric functional form supported by `Thereafter` Ṫhey are accessed via the steering flag called `PDFStyle`. Available styles are summarised as following:

`'10p HERAPDF'`	– HERAPDF-like with extra assumption Buv = Bdv
`'13p HERAPDF'`	– HERAPDF-like with Buv and Bdv floated independently
`'10p H12000'`	– H12000-like (D,U,Dbar,Ubar+g)
`'CTEQ'`	– CTEQ-like parameterisation
`'CHEB'`	– CHEBYSHEV parameterisation based on glu,sea, uval,dval evolved pdfs
`'LHAPDFQ0'`	– use lhapdf library to define pdfs at starting scale and evolve with
local qcdnum parameters	
`'LHAPDF'`	– use lhapdf library to define pdfs at all scales
`' DDIS'`	– use Diffractive DIS
`'BiLog'`	– bi-lognormal parametrisation

These styles were described in details in section 3.1. The LHAPDF style can be used only with proper configuration settings, as explained in the section 5.1.

Definition of Chisquares:

This section is explained in the section 3.2. Currently two different styles are supported for a smoother transition to the new style which is more flexible. The old format corresponds to

`CHI2Style` — (string) choice of the χ^2 function:

`'H12000'`	– Pascaud-like, systematic shifts to theory, no scaling of statistical, uncorrelated err
`'HERAPDF'`	– Pascaud-like + ”mixed error scaling”
`'HERAPDF Sqrt'`	– Pascaud-like + ”sqrt error scaling”
`'HERAPDF Linear'`	– Pascaud-like + ”linear error scaling”
`'Offset'`	– offset method activated

(logical) debug flag: The debug flag will be turned on for more print outs via `LDEBUG.`

Selection of the data:

The namelist &Cuts, located inside the `steering.txt` file can be used to apply simple process dependent cuts. The cuts are limited to bin variables. Simple low and high limits are allowed. For example, a cut on $Q^2 > 3.5$ GeV^2 for NC ep scattering is specified as

```
! Rule #1: Q2 cuts
 ProcessName(1)     = 'NC e+-p'
 Variable(1)        = 'Q2'
 CutValueMin(1)     = 3.5
 CutValueMax(1)     = 1000000.0
```

Maximum 100 cuts can be used by default.

The specific input files are stored in the *input_steerings* directory and it contains the following ready to use inputs (with corresponding minuit files):

- `steering.txt.ALLdata`: all data files
- `steering.txt.DIFFRACTION`: diffraction specific settings
- `steering.txt.kt-factorisation`: kt factorisation specific settings
- `steering.txt.dipole`: dipole model specific settings

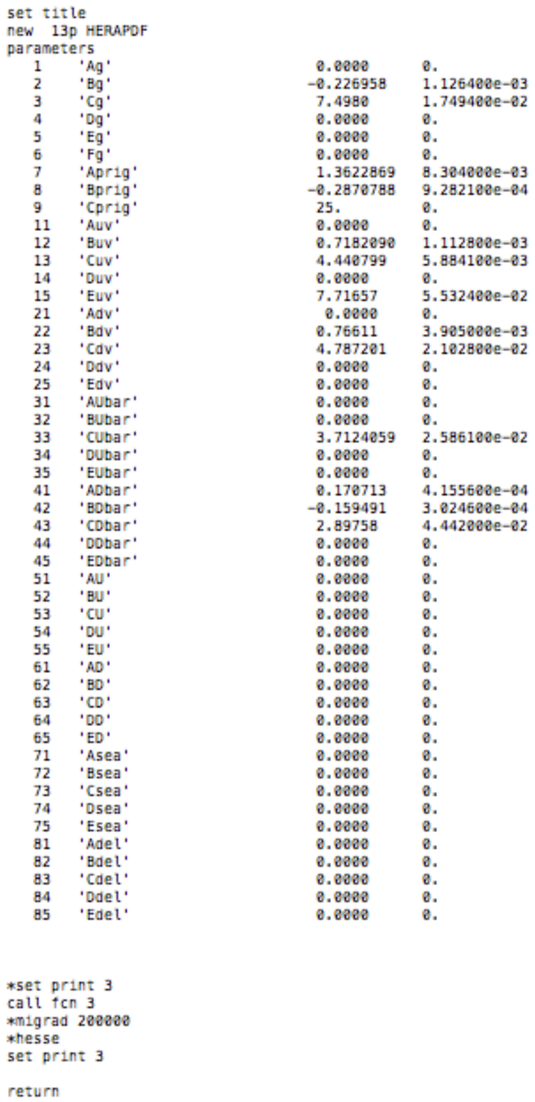
`Minuit` steering cards

The minuit steering card is described as it follows, a sample file is presented in Fig. 9

The first 3 lines set title and announces MINUIT the list of parameters. The index of parameters is the first column and it is hardwired to the source code:

1 -10	gluon parameters
11-20	uval parameters
21-30	dval parameters
31-40	Ubar parameters
41-50	Dbar parameters
51-60	U parameters
61-70	D parameters
71-80	Sea parameters
81-90	Delta parameters
91-100	other parameters: alphas (95), fs=Dbar/str (96), fc=Ubar/ch (97)

The second column represents just user defined names, the third column is input value for the parameter. The forth column sets the step size (usually chosen of the same order as of the error). To note that if step size value is 0. then this parameter is FIXED. The fifth column sets the lower boundary of the fit parameter, The sixth column set the upper boundary of the fit parameter if boundaries are not mentioned then there are no boundaries.



```
set title
new  13p HERAPDF
parameters
   1    'Ag'                           0.0000      0.
   2    'Bg'                          -0.226958    1.126400e-03
   3    'Cg'                           7.4980      1.749400e-02
   4    'Dg'                           0.0000      0.
   5    'Eg'                           0.0000      0.
   6    'Fg'                           0.0000      0.
   7    'Aprig'                        1.3622869   8.304000e-03
   8    'Bprig'                       -0.2870788   9.282100e-04
   9    'Cprig'                        25.         0.
   11   'Auv'                          0.0000      0.
   12   'Buv'                          0.7182090   1.112800e-03
   13   'Cuv'                          4.440799    5.884100e-03
   14   'Duv'                          0.0000      0.
   15   'Euv'                          7.71657     5.532400e-02
   21   'Adv'                           0.0000     0.
   22   'Bdv'                          0.76611     3.905000e-03
   23   'Cdv'                          4.787201    2.102800e-02
   24   'Ddv'                          0.0000      0.
   25   'Edv'                          0.0000      0.
   31   'AUbar'                        0.0000      0.
   32   'BUbar'                        0.0000      0.
   33   'CUbar'                        3.7124059   2.586100e-02
   34   'DUbar'                        0.0000      0.
   35   'EUbar'                        0.0000      0.
   41   'ADbar'                        0.170713    4.155600e-04
   42   'BDbar'                       -0.159491    3.024600e-04
   43   'CDbar'                        2.89758     4.442000e-02
   44   'DDbar'                        0.0000      0.
   45   'EDbar'                        0.0000      0.
   51   'AU'                           0.0000      0.
   52   'BU'                           0.0000      0.
   53   'CU'                           0.0000      0.
   54   'DU'                           0.0000      0.
   55   'EU'                           0.0000      0.
   61   'AD'                           0.0000      0.
   62   'BD'                           0.0000      0.
   63   'CD'                           0.0000      0.
   64   'DD'                           0.0000      0.
   65   'ED'                           0.0000      0.
   71   'Asea'                         0.0000      0.
   72   'Bsea'                         0.0000      0.
   73   'Csea'                         0.0000      0.
   74   'Dsea'                         0.0000      0.
   75   'Esea'                         0.0000      0.
   81   'Adel'                         0.0000      0.
   82   'Bdel'                         0.0000      0.
   83   'Cdel'                         0.0000      0.
   84   'Ddel'                         0.0000      0.
   85   'Edel'                         0.0000      0.


*set print 3
call fcn 3
*migrad 200000
*hesse
set print 3

return
```

Figure 9: An example of a minuit steering card.

Only parameters that have the step size non-zero are let to vary in the fit (free parameters). Another way to fix the parameters is simply by typing at the end of the list of parameters “FIX parameter number”. (make sure there is one line free before the minuit list). Examples of commands taken by minuit:

call fcn 3	fit is not performed, only 1 iteration, useful for testing Minuit parameters ARE NOT minimized.
migrad	fit is performed (default number of calls 2000).
migrad 20000	fit is performed up to 20000 calls, then terminates.
hesse	hessian estimate of the `MINUIT`parameters (more reliable than `MINUIT`)

The output of the fit is stored in the output/ directory as `minuit.out.txt`. Statements to watch in minuit.out.txt:

- `FCN=575.16`this is total chisquare
- `FROM MIGRAD STATUS=CONVERGED`this is desirable for a fit that converged
- `FROM HESSE STATUS=OK`this is desirable for a fit that converged
- `ERROR MATRIX ACCURATE` errors estimated with HESSE method

HERAFitter parameters for diffractive fits

Parameter	HERAFitter name	input file
$A_1^{(G)}$	Ag	minuit.in.txt
$A_2^{(G)}$	Bg	minuit.in.txt
$A_3^{(G)}$	Cg	minuit.in.txt
$A_1^{(S)}$	Auv	minuit.in.txt
$A_2^{(S)}$	Buv	minuit.in.txt
$A_3^{(S)}$	Cuv	minuit.in.txt
$\alpha_{I\!P}(0)$	Pomeron_a0	steering.txt
$A_{I\!R}$	Reggeon_factor	steering.txt
$\alpha_{I\!R}(0)$	Reggeon_a0	steering.txt

HERAFitter parameters for dipole fits

The default initial parameters for the fit without valence quarks are :

σ_0	A_g	λ_g	C_g	$cBGK$	$eBGK$
37.490	3.3446	0.0298	2.6302	4.0	15.362

For the BGK dipole model fits with valence quarks the initial parameters and the obtained χ^2 are:

No	Q^2		σ_0	A_g	λ_g	C_g	$cBGK$	$eBGK$	Np	χ^2	χ^2/Np
1	$Q^2 \geq 3.5$	NLO	35.980	1.964	-0.147	3.068	4.0	15.171	196	245.74	1.254
2	$Q^2 \geq 8.5$	NLO	27.820	3.660	-0.076	8.405	4.0	18.188	157	128.92	0.821

5.3.3 Data file format

Experimental data are provided by the standard `ASCII` text files. The files contain a ”header” which describes the data format and the ”data” in terms of a 2-dimensional table. Each line of the data table corresponds to a data point, the meaning of the columns is specified in the file header.

For example, a header for HERA-I combined H1-ZEUS data for e+p neutral current scattering cross section is given in the file

```
datafiles/H1ZEUS_NC_e-p_HERA1.0.dat
```

The format of the file follows standard ”namelist” conventions. Comments start with exclamation mark. Pre-defined variables are:

- `Name` — (string) provides a name of the data set
- `Reaction` — (string) reaction type of the data set. Reaction type is used to trigger corresponding theory calculation. The following reaction types are currently supported by the HERAFitter:
  - `'NC e+-p'` – double differential NC ep scattering (ZMVFS and RT-VFS schemes)
  - `'CC e+-p'` – double differential CC ep scattering (ZMVFS scheme)
  - `'CC pp'` – single differential $d\sigma_{W^\pm}/deta_{\ell^\pm}$ production and W asymmetry at pp and $p\bar{p}$ colliders (LO+kfactors and APPLGRID interface)
  - `'NC pp'` – single differential $d\sigma_Z/dy_Z$ at pp and $p\bar{p}$ colliders (`LO` with k-factors and `APPLGRID` interface)
  - ’pp jets APPLGRID’ – $pp \rightarrow$ inclusive jet production, using `APPLGRID`
  - ’FastNLO jets’ – jet cross sections using `FastNLO` interface. All ep, pp and $p\bar{p}$ colliders are supported.
  - ’FastNLO ep jets normalised’ – jet cross sections in the ep collisions using `FastNLO` interface and normalised to the inclusive DIS cross sections.
- `NData` — (integer) specifies number of data points in the file. This corresponds to the number of table rows which follow after the header.
- `NColumn` — (integer) number of columns in the data table.
- `ColumnType` — (array of strings) Defines layout of the data table. The following column types are pre-defined: ’Bin’, ’Sigma’, ’Error’ and ’Dummy’ The keywords are case sensitive. ’Bin’ correspond to an abstract bin definition, ’Sigma’ corresponds to the data measurement, ’Error’ - to various type of uncertainties and ’Dummy’ indicates that the column should be ignored.
- `ColumnName` — (array of strings) Defines names of the columns. The meaning of the name depends on the ColumnType. For ColumnType ’Bin’, ColumnName gives a name of the abstract bin. The abstract bins can contain any variable names, but some of them must be present for correct cross section calculation. For example, ’x’, ’Q2’ and ’y’ are required for DIS NC cross-section calculation.

  For ColumnType ’Sigma’, ColumnName provides a label for the observable, which can be any string.

  For ColumnType ’Error’, the following names have special meaning:
  - ’stat’ – specifies column with statistical uncertainties, request Poisson re-scaling;
  - ’stat const’ – specifies column with statistical uncertainties, request no re-scaling of the errors;
  - ’uncor’ – specifies column with uncorrelated uncertainties. Any name containing keyword “uncor” is treated as an uncorrelated error source, e.g. “h1 uncor”;

- 'uncor const' – specifies column with uncorrelated uncertainties, request no re-scaling of the errors;
- 'total' – specifies column with total uncertainties. Total uncertainties are not used in the fit, however there is an additional check is performed if 'total' column is specified: sum in quadrature of statistical, uncorrelated and correlated systematic uncertainties is compared to the total and a warning is issued if they differ significantly.
- 'ignore' - specifies column to be ignored (for special studies).
- Other names specifies columns of correlated systematic uncertainty. For a given data file, each column of the correlated uncertainty must have unique name. To specify correlation across data files, same name must be used for different files.

- `SystScales` — (array of float) For special studies, systematic uncertainties can be scaled The numbering of uncertainties starts from the first column with the ColumnType 'Error'. For example, setting

  ```
  SystScale(1) = 2.
  ```

  in `datafiles/H1ZEUS_NC_e-p_HERA1.0.dat` would scale stat. uncertainty by factor of two.

- `Percent` — (array of bool) For each uncertainty specify if it is given in absolute ("false") or in percent ("true"). The numbering of uncertainties starts from the first column with the `ColumnType` 'Error' (see example above).

- `NInfo` — (integer) Calculation of the cross-section predictions may require additional information about the data set. The number of information strings is given by NInfo

- `CInfo` — (array of strings) Names of the information strings. Several of them are predefined for different cross-section calculations.

- `DataInfo` — (array of float) Values, corresponding to `CInfo` names.

- `IndexDataset` – (integer) Internal H1 Fitter index of the data set. Provide unique numbers to get extra info for χ^2/dof for each data set.

- `TheoryInfoFile` — (string) Optional additional theory file with extra information for cross-section calculation. This could be k-factors, `APPLGRID` file or `FastNLO` table.

- `TheoryType` — (string) Theory file type ('kfactor', 'applgrid' or 'fastnlo').

- `NKFactor` — (integer) For kfactor files, number of columns in `TheoryInfoFile`.

- `KFactorNames` — (array of strings) For kfactor files, names of columns in `TheoryInfoFile`.

Depending on the chosen process specific requirements for the header might be present. Dataset-wise options are provided by a `CInfo` / `DataInfo` variable set. In case the information varies between data points (e.g. bin borders, hadronisation corrections etc.) it is provided within data table and recognised by the program using reserved column names. In the following all these requirements are listed and shortly explained.

Data format requirements for DIS

In this subsection we describe specific requirements for files using 'NC e+-p' and 'CC e+-p' reaction types. Examples of such input files are:

`datafiles/H1ZEUS_NC_e-p_HERA1.0.dat`

`datafiles/H1ZEUS_CC_e-p_HERA1.0.dat`.

The properly formatted DIS input files will have the following fields available in the `CInfo` variable list:

- `'sqrt(S)'` — the ep collision centre-of-mass energy in GeV. In particular, for HERA based results the the corresponding `DataInfo` value should be 300. for measurements based on data collected prior to 1997 (inclusive) and 318. afterwards.
- `'reduced'` — a field indicating whether calculated cross section should be reduced (1.) or not (0.) (reference to proper equation somewhere in this manual).
- `'e charge'` — electric charge of the collided lepton beam. Supported `DataInfo` values are '1.' for electron and '-1.' for positron.
- `'e polarity'` — polarity of the lepton beam. The corresponding `DataInfo` value should be between -1.0 and 1.0 (is this true?) with abs(1.0) indicating fully polarised beam and 0.0 fully unpolarised one.

  In case of non-vanishing polarity following additional fields are required:
- `'pol err unc'` — explain
- `'pol err corLpol'` — explain
- `'pol err corTpol'` — explain

The inclusive DIS cross sections are calculated on the x-Q2-y grid. Correspondingly, the following columns need to present in the correctly formatted input file: `'x'`, `'Q2'` and `'y'`.

Data format requirements for FastNLO

In this subsection we describe data format specific for the FastNLO implementation accessed by choosing 'FastNLO jets' and 'FastNLO ep jets normalised' reaction types. Examples of properly formatted files are:

`datafiles/HERA/ZEUS_InclJets_HighQ2_98-00.dat`

`datafiles/HERA/H1_NormInclJets_HighQ2_99-07.dat`.

`TheoryType = 'FastNLO'` indicates usage of the FastNLO. The variable `ThoryInfoFile` should contain the proper path to the FastNLO table in version 2.0 and higher. HERAFITTER supports both flexible and inflexible scales. Older FastNLO tables can be still accessed through the APPLGRID interface.

The following fields are required to be present in the `CInfo` list:

- `'PublicationUnits'` — The desired units in which the cross sections are calculated by the FastNLO code. If the corresponding `DataInfo` field is set to '1.' the cross sections will be given in the same units as used in the relevant publication. In the case it is set to '0.', absolute cross section units will be used.
- `'MurDef', 'MufDef'` — The renormalisation and factorisation scale definitions used with variable scale FastNLO tables. If the chosen FastNLO table does not support variable scales, these fields will be ignored and the scale embedded within the table will be used instead. The

values of the corresponding DataInfo fields set the renormalisation scale μ_r and factorisation scale μ_f following the FastNLO standard:

$$\begin{aligned}
\text{value} &: \quad \text{definition} \\
0 &: \quad \mu^2_{r/f} = \mu^2_1 \\
1 &: \quad \mu^2_{r/f} = \mu^2_2 \\
2 &: \quad \mu^2_{r/f} = (\mu^2_1 + \mu^2_2) \\
3 &: \quad \mu^2_{r/f} = (\mu^2_1 + \mu^2_2)/2 \\
4 &: \quad \mu^2_{r/f} = (\mu^2_1 + \mu^2_2)/4 \\
5 &: \quad \mu^2_{r/f} = ((\mu_1 + \mu_2)/2)^2 \\
6 &: \quad \mu^2_{r/f} = ((\mu_1 + \mu_2))^2 \\
7 &: \quad \mu^2_{r/f} = \max(\mu^2_1, \mu^2_2) \\
8 &: \quad \mu^2_{r/f} = \min(\mu^2_1, \mu^2_2) \\
9 &: \quad \mu^2_{r/f} = (\mu_1 * exp(0.3 * \mu_2))^2
\end{aligned}$$

where μ_1 and μ_2 are specific scales chosen during production of the table. In particular for jet production at HERA traditionally

$$\mu^2_1 = Q^2 \qquad \mu^2_2 = p^2_T$$

- `sqrt(S)` — Should be defined only for 'FastNLO ep jets normalised' reaction type. The ep collision centre-of-mass energy in GeV. In particular, for HERA based results the the corresponding DataInfo value should be 300. for measurements based on data collected prior to 1997 (inclusive) and 318. afterwards.
- `'lumi(e-)/lumi(tot)'` — Should be defined only for 'FastNLO ep jets normalised' reaction type. The normalisation depends on the ratio of the positron and electron data used for the cross section measurement. This ratio should be given in a format (lumi(e^-) / (lumi(e^-) + lumi(e^+)) and assume values between [0., 1.].
- `'UseZMVFNS'` — Should be defined for 'FastNLO ep jets normalised' reaction type. The calculation of the integrated inclusive DIS cross sections could be time consuming. This option provides an opportunity to use a "Zero Mass Variable Flavour Number Scheme" approximation which is very fast and possibly provides enough precision for the normalisation purposes. ZMVNS is used if the corresponding DataInfo field is set to 1. Otherwise, the same scheme is used as defined globally with the variable 'HF_SCHEME' defined in steering.txt file.

In addition there are some specific values within the ColumnName field which allow passing the information specific to each data point. They are listed below:

- `'Z0Corr'` — (optional) The correction due to the Z_0 boson exchange. If it is given, each point calculated by the FastNLO code will be multiplied by the Z0Corr value.
- `'NPCorr'` — (optional) The non-perturbative correction. If it is given, each point calculated by the FastNLO code will be multiplied by the NPCorr value. Z0Corr and NPCorr can be added simultaneously, and in this case the calculated cross sections will be multiplied by the product Z0Corr * NPCorr.

- 'q2min', 'q2max', 'ymin', 'ymax', 'xmin', 'xmax' — Should be defined for 'FastNLO ep jets normalised' reaction type and are used to define DIS phase space for the normalisation. Since these three (q2, y, x) are connected by the relation

  $$Q^2 = x \cdot y \cdot s \tag{80}$$

  only two are required to be present to unambiguously define the DIS phase space for each data point.

5.3.4 Understanding the output

The results of the minimization are printed to the standard output and written to the files in the output/ directory.

The quality of the fit can be judged based on total χ^2 per degrees of freedom. It is printed for each iteration as

```
                         Iteration   Chi2   NDF      Chi2/NDF
   FitPDF f,ndf,f/ndf       3      588.64 579         1.02
```

The resulting χ^2 is reported at the end of minimisation for each data set and for correlated systematic uncertainties separately. This information is printed and written to the output/Results.txt file. The Results.txt file contains additional information about shifts of the correlated systematic uncertainties.

The minimization information from the minuit program is stored using the standard minuit in the output/minuit.out.txt file. The level of verbosity for this information can be changed by minuit commands in the minuit.in.txt file. Make sure that minuit does not report any errors or warnings at the end of minimisation.

Point by point comparison of the data and predictions after the minimization is provided in the file stored in output/fittedresults.txt. The file reports three columns corresponding to the three first bins of the input tables, data value, sum in quadrature of statistical and uncorrelated systematic uncertainty, total uncertainty, the predicted value, before and after applying correlated systematic shifts, pull between the data and theory and data set index. The pull p is calculated as

$$p = \frac{\mu - m}{\sigma_{\text{uncor}}} \tag{81}$$

where μ is the data value, m is the prediction and σ_{uncor} is the total uncorrelated uncertainty. Similar information is stored in the pulls.first.txt and pulls.last.txt files (dataset index, first bin, second bin, third bin, theory, data, pull). Theory is adjusted for systematic error shifts in this case.

The output PDFs are stored in output/pdfs_q2val_XX.txt files. Each of the files reports values of gluon, and quark PDFs as a function of x for fixed Q^2 points. The Q^2 values and x grid are specified by &Output namelist in the steering.txt file.

The PDF information and data to theory comparisons can be plotted using the bin/DrawResults program. Calling it without arguments plots results from output/ directory. Given the program one argument specifies sub-directory where the information is read. Calling the bin/DrawResults program with two arguments provides comparison of the PDFs obtained in the two fits.

Finally, the Thereafter package provides PDFs in the LHAPDF format. To obtain the LHAPDF grid file, run the tools/tolhapdf.cmd script. The script produces the PDFs.LHgrid file which can be read by the lhapdf version lhapdf-5.8.6.tar.gz or later.

5.4 User Example

Two examples are available in Thereafter for benchmarking purposes. First example describes the default fit with HERA DIS inclusive data, second is a fit where all available data in Thereafter are fitted simultaneously.

5.4.1 DIS inclusive only

By default in Thereafter steering files (steering.txt and minuit.in.txt) are set to fit the DIS inclusive cross section data. For this fit no any additional configuration options are required. The result of this example fit obtained by user (total and partial $\chi 2$, systematic shifts, data to theory comparison and histograms, see section 8.4 ”Understanding the output” for more details) can be compared to result provided for benchmarking in ”examples”.

5.4.2 All processes

In order to run the second example with all data, the corresponding steering.txt from ”input_steering” has to be copied to the main directory:

```
cp input-steering/steering.txt.ALLdata steering.txt
```

User must make sure that all data sets as given in steering.txt.ALLdata together with corresponding theory file have been downloaded before running this example.
Since the included sets contain various ep, pp, $p\overline{p}$ and fix target data, it is necessary to have Thereafter configured with applgrid, hathor and lhapdf options (corresponding shared library linking as explained in section 5.1 is required before configuration):

```
./configure –enable-applgrid –enable-lhapdf –enable-hathor
```

It is recommended to do ”make clean” before each configuration.
As in previous case, the fit result can be compared to the one provided in ”examples”.

A How to add new data

Inclusion of the data files is controlled by &InFiles namelist in the steering.txt file. For example, by default the following four HERA-I files are included:

```
&InFiles
    NInputFiles = 4
    InputFileNames(1) = 'datafiles/H1ZEUS_NC_e-p_HERA1.0.dat'
    InputFileNames(2) = 'datafiles/H1ZEUS_NC_e+p_HERA1.0.dat'
    InputFileNames(3) = 'datafiles/H1ZEUS_CC_e-p_HERA1.0.dat'
    InputFileNames(4) = 'datafiles/H1ZEUS_CC_e+p_HERA1.0.dat'
&End
```

To include more files:

- Increase the NInputFiles variable.
- Specify the additional file by providing corresponding InputFileNames() variable.

Details about data file format can be found in section 5.3.3.

References

[1] F. James and M. Roos, Comput. Phys. Commun. **10**, 343 (1975).

[2] V. N. Gribov and L. N. Lipatov, Sov. J. Nucl. Phys. **15**, 438 (1972).

[3] V. N. Gribov and L. N. Lipatov, Sov. J. Nucl. Phys. **15**, 675 (1972).

[4] L. N. Lipatov, Sov. J. Nucl. Phys. **20**, 94 (1975).

[5] Y. L. Dokshitzer, Sov. Phys. JETP **46**, 641 (1977).

[6] G. Altarelli and G. Parisi, Nucl. Phys. B **126**, 298 (1977).

[7] M. Botje (2010), http://www.nikef.nl/h24/qcdnum/index.html, [arXiv:1005.1481].

[8] G. Curci, W. Furmanski, and R. Petronzio, Nucl.Phys. **B175**, 27 (1980).

[9] W. Furmanski and R. Petronzio, Phys.Lett. **B97**, 437 (1980).

[10] E. L. *et al.*, Phys. Lett. **B291**, 325 (1992).

[11] E. L. *et al.*, Nucl. Phys. **B392**, 162, 229 (1993).

[12] S. Riemersma, J. Smith, and van Neerven. W.L., Phys. Lett. **B347**, 143 (1995), [hep-ph/9411431].

[13] R. Demina, S. Keller, M. Kramer, S. Kretzer, R. Martin, *et al.* (1999), [hep-ph/0005112].

[14] R. S. Thorne and R. G. Roberts, Phys. Rev. D **57**, 6871 (1998), [hep-ph/9709442].

[15] R. S. Thorne, Phys. Rev. **D73**, 054019 (2006), [hep-ph/0601245].

[16] S. Alekhin, *OPENQCDRAD*, a program description and the code are available via: http://www-zeuthen.desy.de/~alekhin/OPENQCDRAD.

[17] A. D. Martin, Eur. Phys. J. C **63**, 189 (2009).

[18] R. S. Thorne (2012), [arXiv:1201.6180].

[19] J. C. Collins, Phys.Rev. **D58**, 094002 (1998), [hep-ph/9806259].

[20] S. Alekhin and S. Moch, Phys. Lett. **B699**, 345 (2011), [arXiv:1011.5790].

[21] K. H., N. Lo Presti, S. Moch, and A. Vogt, Nucl.Phys. **B864**, 399 (2012).

[22] H. Spiesberger, Private communication.

[23] Jegerlehner, Proceedings, LC10 Workshop **DESY 11-117** (2011).

[24] . Burkhard, (input from Voica needed).

[25] Y. Li and F. Petriello, Phys.Rev. **D86**, 094034 (2012), [arXiv:1208.5967].

[26] G. Bozzi, J. Rojo, and A. Vicini, Phys.Rev. **D83**, 113008 (2011), [arXiv:1104.2056].

[27] A. Falkowski, M. L. Mangano, A. Martin, G. Perez, and J. Winter (2012), [arXiv:1212.4003].

[28] S. D. Drell and T.-M. Yan, Phys. Rev. Lett. **25**, 316 (1970).

[29] M. Yamada and M. Hayashi, Nuovo Cim. **A70**, 273 (1982).

[30] M. Aliev, H. Lacker, U. Langenfeld, S. Moch, P. Uwer, *et al.*, Comput.Phys.Commun. **182**, 1034 (2011), [arXiv:1007.1327].

[31] P. Bärnreuther, M. Czakon, and A. Mitov (2012), [arXiv:1204.5201].

[32] S. Moch, P. Uwer, and A. Vogt, Phys.Lett. **B714**, 48 (2012), [hep-ph/1203.6282].

[33] T. Kluge, K. Rabbertz, and M. Wobisch, pp. 483–486 (2006), [hep-ph/0609285].

[34] M. Wobisch, D. Britzger, T. Kluge, K. Rabbertz, and F. Stober [fastNLO Collaboration] (2011), [arXiv:1109.1310].

[35] D. Britzger, K. Rabbertz, F. Stober, and M. Wobisch [fastNLO Collaboration] (2012), [arXiv:1208.3641].

[36] Z. Nagy and Z. Trocsanyi, Phys.Rev. **D59**, 014020 (1999), [hep-ph/9806317].

[37] Z. Nagy and Z. Trocsanyi, Phys.Rev.Lett. **87**, 082001 (2001), [hep-ph/0104315].

[38] Z. Nagy, Phys.Rev. **D68**, 094002 (2003), [hep-ph/0307268].

[39] Z. Nagy, Phys.Rev.Lett. **88**, 122003 (2002), [hep-ph/0110315].

[40] N. Kidonakis and J. Owens, Phys.Rev. **D63**, 054019 (2001), [hep-ph/0007268].

[41] T. Carli *et al.*, Eur. Phys. J. **C66**, 503 (2010), [arXiv:0911.2985].

[42] J. M. Campbell and R. K. Ellis, Phys. Rev. **D60**, 113006 (1999), [arXiv:9905386].

[43] J. M. Campbell and R. K. Ellis, Nucl. Phys. Proc. Suppl. **205-206**, 10 (2010), [arXiv:1007.3492].

[44] N. N. Nikolaev and B. Zakharov, Z.Phys. **C49**, 607 (1991).

[45] K. Golec-Biernat and M. Wüsthoff, Phys. Rev. D **59**, 014017 (1999), [hep-ph/9807513].

[46] E. Iancu, K. Itakura, and S. Munier, Phys. Lett. **B590**, 199 (2004), [hep-ph/0310338].

[47] J. Bartels, K. Golec-Biernat, and H. Kowalski, Phys. Rev. D **66**, 014001 (2002), [hep-ph/0203258].

[48] I. Balitsky, Nucl. Phys. B **463**, 99 (1996), [hep-ph/9509348].

[49] S. Catani, M. Ciafaloni, and F. Hautmann, Nucl. Phys. B **366**, 135 (1991).

[50] M. Ciafaloni, Nucl. Phys. B **296**, 49 (1988).

[51] S. Catani, F. Fiorani, and G. Marchesini, Phys. Lett. B **234**, 339 (1990).

[52] S. Catani, F. Fiorani, and G. Marchesini, Nucl. Phys. B **336**, 18 (1990).

[53] G. Marchesini, Nucl. Phys. B **445**, 49 (1995).

[54] H. Jung and F. Hautmann (2012), [arXiv:1206.1796].

[55] H. Jung, S. Baranov, M. Deak, A. Grebenyuk, F. Hautmann, *et al.*, Eur.Phys.J. **C70**, 1237 (2010), [arXiv:1008.0152].

[56] M. Deak, F. Hautmann, H. Jung, and K. Kutak, *Forward-Central Jet Correlations at the Large Hadron Collider* (2010), [arXiv:1012.6037].

[57] A. Airapetian *et al.* [HERMES Collaboration], Phys.Lett. **B666**, 446 (2008), [arXiv:0803.2993].

[58] A. Schöening (2011), Private communication.

[59] F. Aaron *et al.* [H1 Collaboration], Eur. Phys. J. **C63**, 625 (2009), [arXiv:0904.0929].

[60] D. Stump *et al.*, Phys. Rev. **D65**, 014012 (2002), [hep-ph/0101051].

[61] M. Botje, J.Phys. **G28**, 779 (2002), [hep-ph/0110123].

[62] C. Pascaud and F. Zomer (1995), lAL-95-05.

[63] W. T. Giele and S. Keller, Phys.Rev. **D58**, 094023 (1998), [hep-ph/9803393].

[64] W. T. Giele, S. Keller, and D. Kosower (2001), [hep-ph/0104052].

[65] A. Glazov, S. Moch, and V. Radescu, Phys. Lett. B **695**, 238 (2011), [arXiv:1009.6170].

[66] R. D. Ball, V. Bertone, F. Cerutti, L. Del Debbio, S. Forte, *et al.*, Nucl.Phys. **B855**, 608 (2012), [arXiv:1108.1758].

[67] R. D. Ball *et al.* [NNPDF Collaboration], Nucl.Phys. **B849**, 112 (2011), [arXiv:1012.0836].

[68] G. Watt and R. Thorne, JHEP **1208**, 052 (2012), [arXiv:1205.4024].