

The new theory prediction pipeline and intrinsic charm

Felix Hekhorn

Jyväskylä QCD Seminar April 2023

Acknowledgement: This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement number 740006.

1. The new theory pipeline [\[2302.12124\]](#)

1.1 Motivation

1.2 PineAPPL [\[JHEP12.108\]](#)

1.3 pinefarm

1.4 EKO [\[EPJC82.976\]](#)

1.5 yadism [\[in preparation\]](#)

1.6 Outlook

2. Intrinsic Charm [\[Nature608.483\]](#)

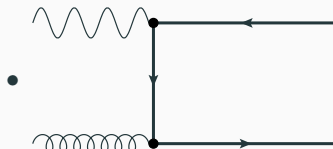
1. The new theory pipeline

[2302.12124]

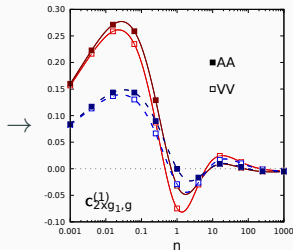
1.1. Motivation

Including New Computations

- Computing new observables is expensive both in runtime (days/weeks) and development time (month/years)
- E.g. NLO heavy quark production in polarized DIS [\[PRD98.014018\]](#) [\[1910.01536\]](#) [\[PRD104.016033\]](#)



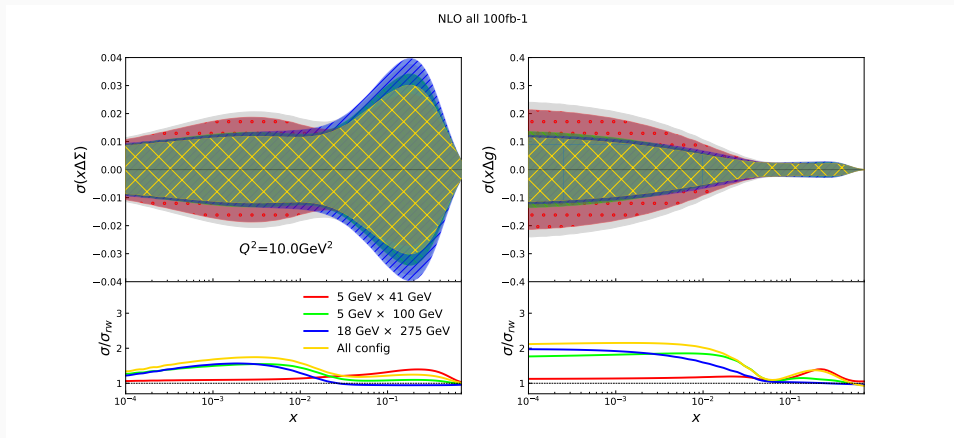
```
cdbl FullyDiff:ME:PRD_F2_V(cdbl m2, cdbl q2, cdbl sp, cdbl t1, cdbl u1, cdbl
tp, cdbl up) {
  int
  return tp*(-((q2*s5)/(q2+sp)) + up*((((8*m2)/(q2*s3) + (8*m2)/(q2*s4) + (32*Po
wer(m2,2))/(q2*s3*s4) - (8*s5)/(q2*s4) - (8*Power(15,2))/(q2*s3
*s4))/8. + ((8*m2)/(q2*t1) - (8*s5)/(q2*t3) + (8*m2)/(q2*u6) - (8*s5)/(q2*u6) +
(32*Power(m2,2))/(q2*t1*u6) - (8*Power(15,2))/(q2*t1*u6))/8. + ((-4*m2)/(q2*tp)
+ (8*s3)/(q2*tp) - (4*s5)/(q2*tp) - (4*t1)/(q2*tp) - (4*m2)/(q2*u6) - (4*s5)/(q2
*u6) - (8*m2*s3)/(q2*tp*u6) - (8*m2*s4)/(q2*tp*u6) - (8*s4*s5)/(q2*tp*u6) - (4*m
2*t1)/(q2*tp*u6) - (4*s4*t1)/(q2*tp*u6))/8. + ((4*m2)/(q2*tp) - (8*s3)/(q2*tp) +
(4*s5)/(q2*tp) + (4*t1)/(q2*tp) + (4*m2)/(q2*u6) + (4*s5)/(q2*u6) + (8*m2*s3)/(
q2*tp*u6) + (8*m2*s4)/(q2*tp*u6) + (8*s4*s5)/(q2*tp*u6) + (4*m2*t1)/(q2*tp*u6) +
(4*s4*t1)/(q2*tp*u6))/8. + ((8*s5)/(q2*s4) - (32*Power(m2,2))/(q2*s4*u6) + (8*
m2*s3)/(q2*s4*u6) - (8*s5)/(q2*u6) + (8*m2*s5)/(q2*s4*u6) + (8*m2*t1)/(q2*s4*u6)
+ (8*m2*tp)/(q2*s4*u6) - (8*s5*tp)/(q2*s4*u6))/8. + ((-8*s5)/(q2*s3) - (32*Pow
er(m2,2))/(q2*s3*t1) + (8*m2*s6)/(q2*s3*t1) - (8*s5)/(q2*t1) + (8*m2*s5)/(q2*s3*t
1) + (8*m2*tp)/(q2*s3*t1) - (8*s5*tp)/(q2*s3*t1) + (8*m2*u6)/(q2*s3*t1))/8. + (-
((s4*t1)/(q2*Power(tp,2))) + s5/(q2*tp) - (s3*u6)/(q2*Power(tp,2)))/2. + ((s4*t1
)/(q2*Power(tp,2)) - s5/(q2*tp) + (s3*u6)/(q2*Power(tp,2)))/2. + ((-4*s3*s4)/(q2
*Power(tp,2)) - (4*s4*t1)/(q2*Power(tp,2)) - (32*m2)/(q2*tp) - (2*s5)/(q2*tp) -
(34*s3*u6)/(q2*Power(tp,2)) - (4*t1*u6)/(q2*Power(tp,2)))/4. + ((4*s3*s4)/(q2*P
ower(tp,2)) + (34*s4*t1)/(q2*Power(tp,2)) + (32*m2)/(q2*tp) + (2*s5)/(q2*tp) + (
34*s3*u6)/(q2*Power(tp,2)) + (4*t1*u6)/(q2*Power(tp,2)))/4. + ((-4*m2)/(q2*s3) -
```



- How to measure the actual impact on PDFs?

Reweighting

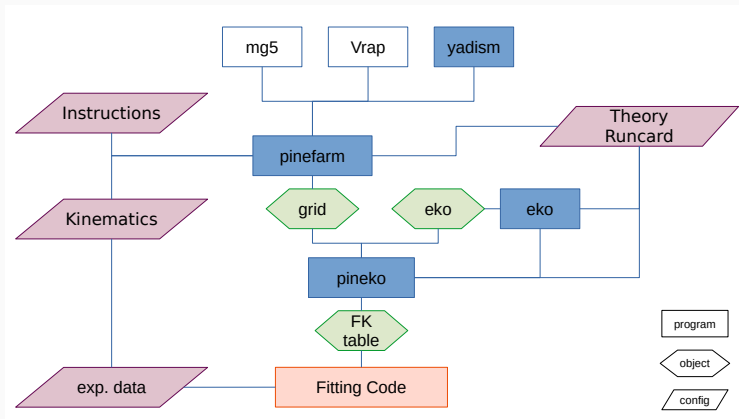
Reweighting is possible [PRD104.114039] - and even needed for the EIC



but a new PDF fit would be better!

New Theory Prediction Pipeline

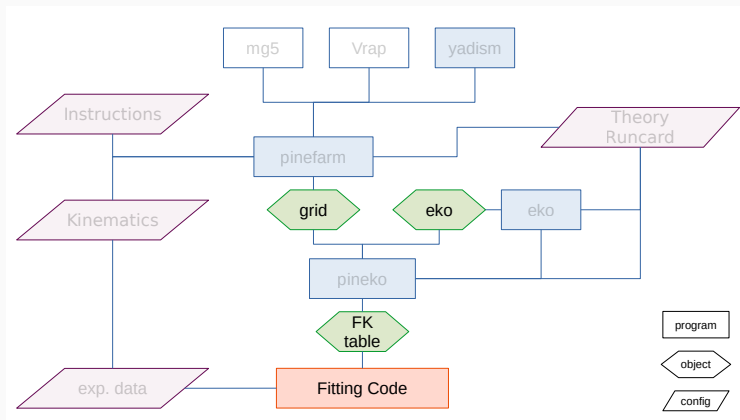
Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

New Theory Prediction Pipeline

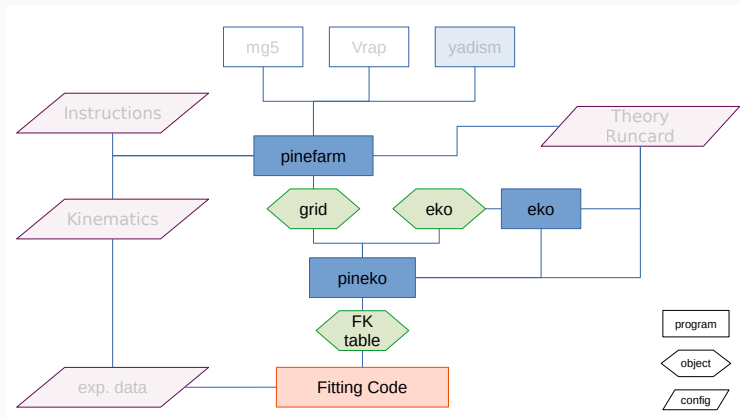
Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

New Theory Prediction Pipeline

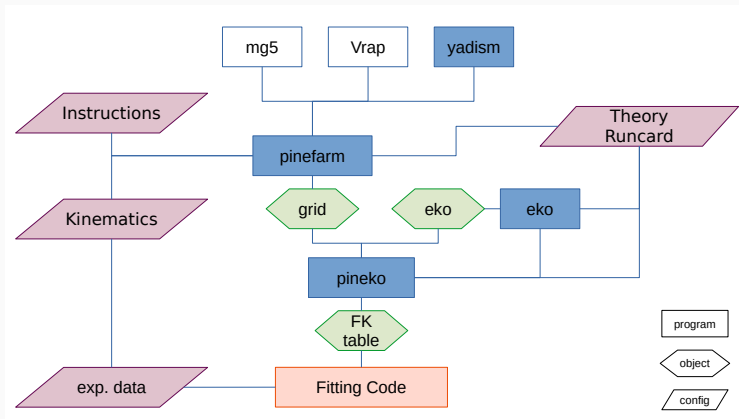
Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

New Theory Prediction Pipeline

Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL



<https://nnpdf.github.io/pipeline>

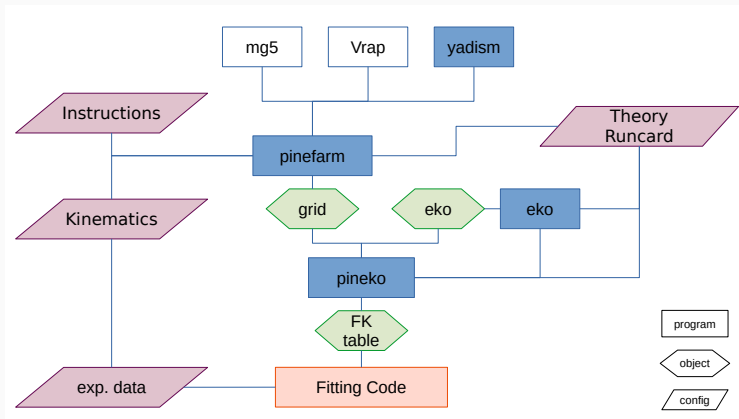
- **Industrialization:** collect diverse generators in an “assembly line”
→ NNP4.0[EPJC82.428]: > 4.5k datapoints + > 10 generators
- **I/O format:** single input → translation layer → single output
- **Reproducibility:** track data and metadata
- **Open Source:** crucial to the above

⇒ please provide new calculations in an “interfaceable” way

1.2. PineAPPL [JHEP12.108]

New Theory Prediction Pipeline

Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

PineAPPL is a fast interpolation grid library that

- extends to arbitrary orders in QCD and EW coupling
- provides a very good Command Line Interface
- provides several interfaces: C, C++, Fortran, Rust, Python
- can convert APPLgrid [EPJC66.503] and FastNLO [DIS12.217]

🔗 <https://github.com/NNPDF/pineappl>

📖 <https://nnpdf.github.io/pineappl/>

The Command Line Interface

```
$ pineappl convolute CMS_DY_14TEV_MLL_5000_COSTH.pineappl.lz4 \  
NNPDF40_nnlo_as_01180
```

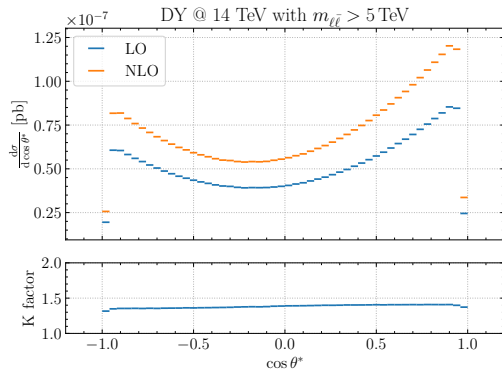
b	costh	dsig/dcosth	scale uncertainty
	[]	[pb]	[%]

0	-1	-0.96	5.0382145e-8	-4.73	4.32
1	-0.96	-0.92	1.6366674e-7	-4.98	4.62
2	-0.92	-0.88	1.6611145e-7	-5.06	4.70
3	-0.88	-0.84	1.5983761e-7	-5.08	4.74
4	-0.84	-0.8	1.5374426e-7	-5.09	4.75
5	-0.8	-0.76	1.4800320e-7	-5.10	4.76
6	-0.76	-0.72	1.4238050e-7	-5.10	4.76
7	-0.72	-0.68	1.3708378e-7	-5.10	4.76
8	-0.68	-0.64	1.3191722e-7	-5.12	4.78

[...]

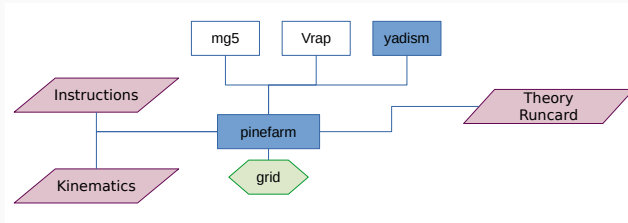
The Python Interface

```
1 import lhapdf
2 import pineappl
3 # load PDF
4 pdf = lhapdf.mkPDF("
      NNPDF40_nnlo_as_01180",0)
5 # load grid
6 grid = pineappl.grid.Grid.read("
      CMS_DY_14TEV_MLL_5000_COSTH.
      pineappl.lz4")
7 # convolute
8 print(grid.convolute_with_one(2212,pdf.
      xfxQ2,pdf.alphasQ2))
9 # prints the same list of numbers
```

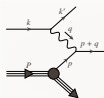


1.3. pinefarm

Interface to Other Programs: pinefarm



```
*****
*
*  WELCOME to MADGRAPH5_aMC@NLO
*
*
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*
*  The MadGraph5_aMC@NLO Development Team - Find us at
*  https://server06.fynu.ucl.ac.be/projects/madgraph
*  and
*  http://amcatnlo.cern.ch
*
*  Code download from:
*  https://launchpad.net/madgraph5
*
*  Please refer to: MadGraph5_aMC@NLO paper
*  J. Alwall et al.
*  arXiv:1405.0301, JHEP 1407 (2014) 079
*
*****
```



Yadism
Yet Another DIS Module

[in preparation]












Vrap [PRD69.094008]

more MC interfaces coming
soon (e.g. MATRIX
[EPJC78.537])



The pinecards repo

Collect available observables:  <https://github.com/NNPDF/pinecards>

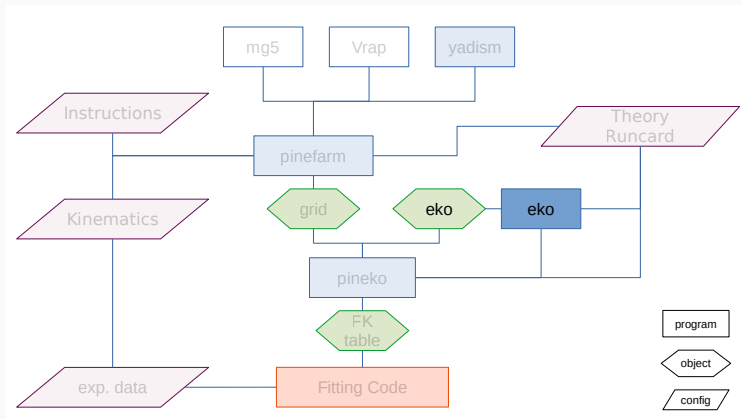
 ATLAS_TTB_8TEV_LJ_TTRAP	Fix ordering of model loading and model-specific settings	2 weeks ago
 ATLAS_TTB_8TEV_TOT	Fix ordering of model loading and model-specific settings	2 weeks ago
 ATLAS_WM_7TEV	Fix ordering of model loading and model-specific settings	2 weeks ago
 ATLAS_WP_7TEV	Fix ordering of model loading and model-specific settings	2 weeks ago
 BCDMS_NC_EM_D_F2	Export pinefarm to its own repo	3 months ago
 BCDMS_NC_EM_P_F2	Export pinefarm to its own repo	3 months ago
 CHORUS_CC_NB_PB_SIGMARED	Export pinefarm to its own repo	3 months ago
 CHORUS_CC_NU_PB_SIGMARED	Export pinefarm to its own repo	3 months ago
 CMS_2JET_7TEV_0005	Fix ordering of model loading and model-specific settings	2 weeks ago
 CMS_2JET_7TEV_0510	Fix ordering of model loading and model-specific settings	2 weeks ago
 CMS_2JET_7TEV_1015	Fix ordering of model loading and model-specific settings	2 weeks ago

a similar effort for the experimental data is ongoing!

1.4. EKO [EPJC82.976]

New Theory Prediction Pipeline

Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL



DGLAP:

$$\mu_F^2 \frac{d\mathbf{f}}{d\mu_F^2}(\mu_F^2) = \mathbf{P}(a_s(\mu_R^2), \mu_F^2) \otimes \mathbf{f}(\mu_F^2)$$

as operator equation for the evolution kernel operator (EKO) \mathbf{E} :

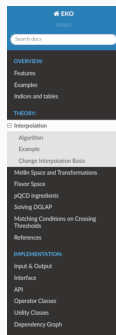
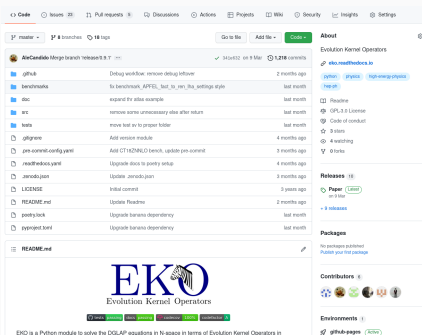
$$\mu_F^2 \frac{d}{d\mu_F^2} \mathbf{E}(\mu_F^2 \leftarrow \mu_{F,0}^2) = \mathbf{P}(a_s(\mu_R^2), \mu_F^2) \otimes \mathbf{E}(\mu_F^2 \leftarrow \mu_{F,0}^2)$$

with

$$\mathbf{f}(\mu_F^2) = \mathbf{E}(\mu_F^2 \leftarrow \mu_{F,0}^2) \otimes \mathbf{f}(\mu_{F,0}^2)$$

- independent of boundary condition \rightarrow PDF fitting
- Mellin (N -) space solution, but momentum (x -) space delivery via piecewise Lagrange-interpolation
- Intrinsic heavy quark distributions \rightarrow see part 2
- Backward VFNS evolution (i.e. across thresholds and with intrinsic) \rightarrow see part 2

EKO Project Management



• Interpolation [Edit on GitHub](#)

Interpolation

Implementation: `eko.interpolation`

In order to obtain the operators in an PDF independent way we use approximation theory. Therefore, we define the basis grid

$$\mathbb{G} = \{x_j : 0 < x_j < 1, j = 0, \dots, N_{grid} - 1\}$$

from which we define our interpolation

$$f(x) \sim \tilde{f}(x) = \sum_{j=0}^{N_{grid}-1} f(x_j) p_j(x)$$

Thus each grid point x_j has an associated interpolation polynomial $p_j(x)$ (represented by `eko.interpolation.BasisFunction`). We interpolate in $\ln(x)$ using Lagrange interpolation among the nearest $N_{approx} + 1$ points, which renders the $p_j(x)$: polynomials of order $O(\ln N_{approx}(x))$.

Algorithm

First, we split the interpolation region into several areas (represented by `eko.interpolation.Area`), which are bound by the grid points:

$$A_j = [x_j, x_{j+1}], \quad \text{for } j = 0, \dots, N_{grid} - 2$$

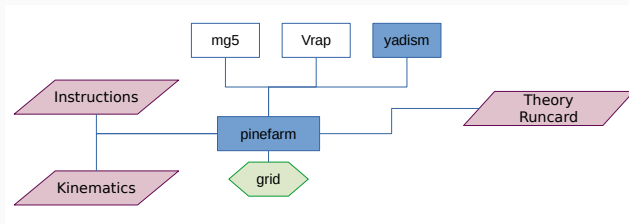
Note, that we include the right border point into the definition, but not the left which keeps all areas disjoint. This assumption is based on the physical fact, that PDFs do have a fixed upper bound ($x = 1$), but no fixed lower bound.

Second, we define the interpolation blocks, which will build the interpolation polynomials and contain the needed amount of points:

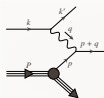
- Fully open source: <https://github.com/NNPDF/eko>
- Written in Python
- Fully documented: <https://eko.readthedocs.io/>

1.5. yadism [in preparation]

Interface to Other Programs: pinefarm



```
*****
*
*  WELCOME to MADGRAPH5_aMC@NLO
*
*
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*  *      *      *
*
*  The MadGraph5_aMC@NLO Development Team - Find us at
*  https://server06.fynu.ucl.ac.be/projects/madgraph
*  and
*  http://amcatnlo.cern.ch
*
*  Code download from:
*  https://launchpad.net/madgraph5
*
*  Please refer to: MadGraph5_aMC@NLO paper
*  J. Alwall et al.
*  arXiv:1405.0301, JHEP 1407 (2014) 079
*
*****
```



Yadism
Yet Another DIS Module

[in preparation]

Vrap [PRD69.094008]

more MC interfaces coming
soon (e.g. MATRIX
[EPJC78.537])





- DIS coefficient function database
- independent of boundary condition \rightarrow PDF fitting
- separate features: TMC, FNS
- constant benchmark against APFEL

same improvement in terms of project management as EKO!

 <https://github.com/NNPDF/yadism>

 <https://yadism.readthedocs.io>

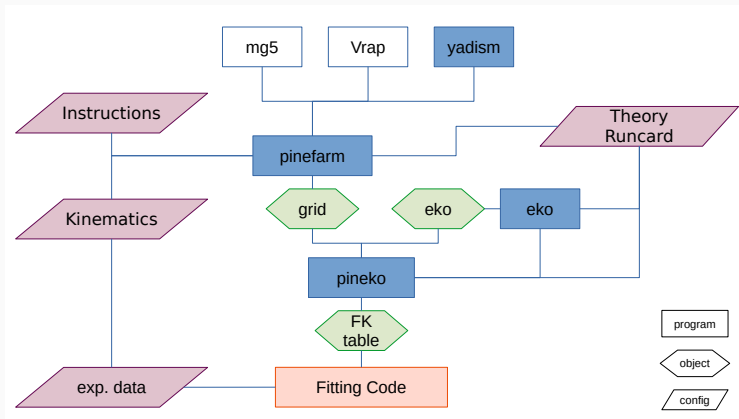
- implemented coefficient functions:

	light	heavy	intrinsic
NC	$O(a_s^2)$ [VVM05],[MVV05],[MV00]	$O(a_s^2)$ [Hek19]	$O(a_s)$ [KS98]
CC	$O(a_s^2)$ [MRV08],[MVV09]	$O(a_s)$ [GKR96]	$O(a_s)$ [in prep.]

- implemented flavor number schemes: FFNS, ZM-VFNS, FONLL

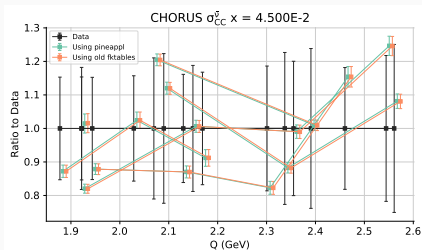
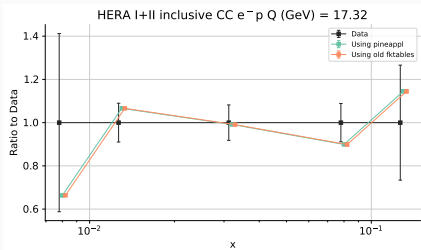
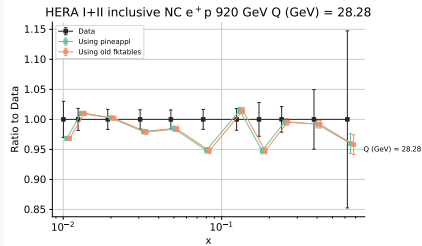
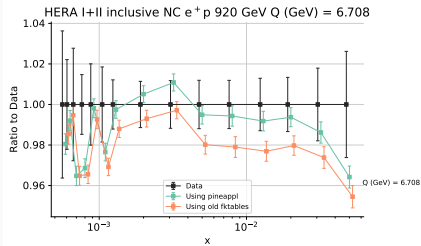
New Theory Prediction Pipeline

Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

Comparison yadism against APFEL



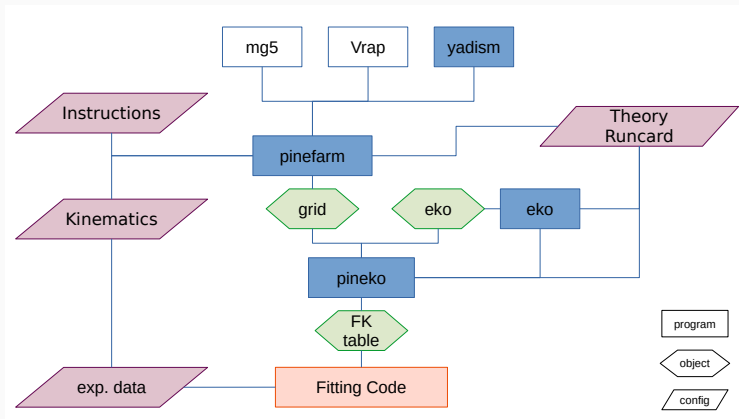
green, "pineappl" = pineline vs. orange, "old" = APFEL

1.6. Outlook

- extend to N3LO
- include missing higher order uncertainty (MHOU)
- include QED corrections
- add polarized setup
- extend to fragmentation function
- add EW corrections
- ...

New Theory Prediction Pipeline

Produce FastKernel (FK) tables!



The workhorse in the background: PineAPPL

2. Intrinsic Charm [Nature608.483]

- **perturbative charm**

- is fully perturbative, i.e., predictable at all scales
- generated by matching conditions and evolution
- always present for $\mu_F > \mu_h = m_h$
- it is $g \rightarrow c\bar{c}$, so no asymmetry possible ($c \neq \bar{c}$)
- default for CTEQ and MSHT

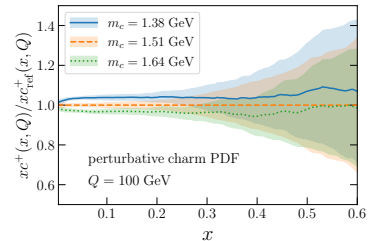
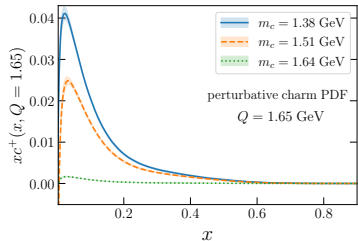
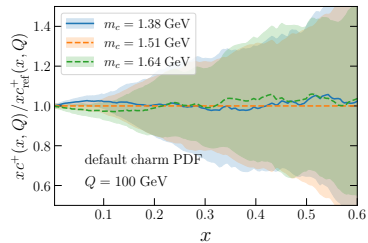
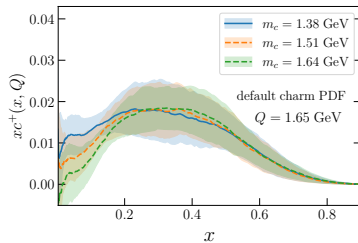
- **intrinsic charm**

- is non-perturbative
- charm in 3 flavor scheme
- CTEQ: use a model (e.g. [BHPS])

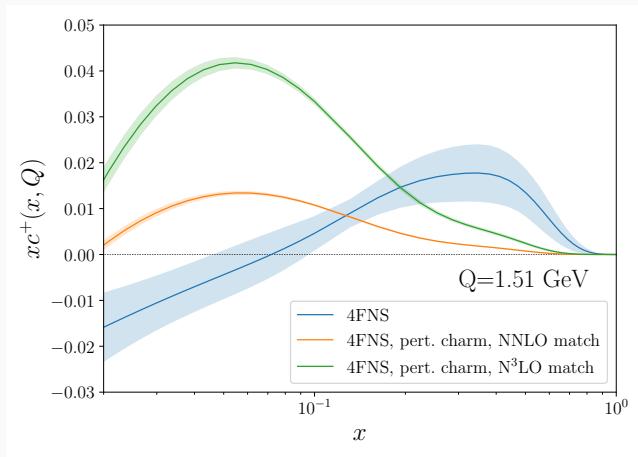
- **fitted charm**

- default for NNPDF
- don't assume anything - just fit charm in 4 flavor scheme!
- is an arbitrary mixture of intrinsic and perturbative charm

Mass Dependency on PDFs



Mass Dependency by OMEs



- slow perturbative convergence of OME: NNLO and N3LO differ significantly

Matching Conditions and Backward Evolution

For (forward) evolution across a matching scale μ_h^2 :

$$\tilde{\mathbf{f}}^{(n_f+1)}(\mu_{F,1}^2) = \tilde{\mathbf{E}}^{(n_f+1)}(\mu_{F,1}^2 \leftarrow \mu_h^2) \mathbf{R}^{(n_f)} \tilde{\mathbf{A}}^{(n_f)}(\mu_h^2) \tilde{\mathbf{E}}^{(n_f)}(\mu_h^2 \leftarrow \mu_{F,0}^2) \tilde{\mathbf{f}}^{(n_f)}(\mu_{F,0}^2) \quad (1)$$

with $\mathbf{R}^{(n_f)}$ a flavor rotation matrix and $\tilde{\mathbf{A}}^{(n_f)}(\mu_h^2)$ the operator matrix elements (partially known up to N³LO)

Matching Conditions and Backward Evolution

For (forward) evolution across a matching scale μ_h^2 :

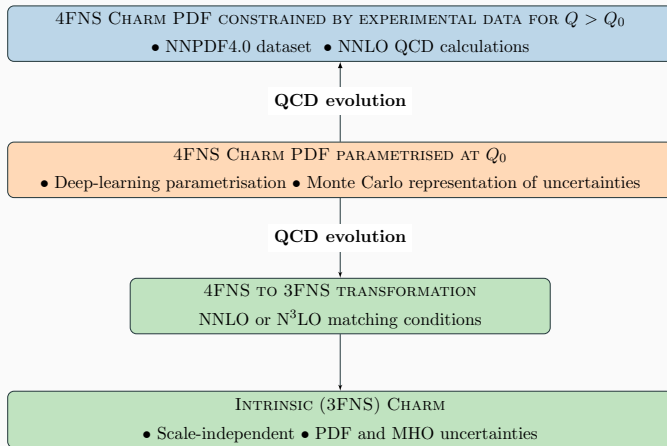
$$\tilde{\mathbf{f}}^{(n_f+1)}(\mu_{F,1}^2) = \tilde{\mathbf{E}}^{(n_f+1)}(\mu_{F,1}^2 \leftarrow \mu_h^2) \mathbf{R}^{(n_f)} \tilde{\mathbf{A}}^{(n_f)}(\mu_h^2) \tilde{\mathbf{E}}^{(n_f)}(\mu_h^2 \leftarrow \mu_{F,0}^2) \tilde{\mathbf{f}}^{(n_f)}(\mu_{F,0}^2) \quad (1)$$

with $\mathbf{R}^{(n_f)}$ a flavor rotation matrix and $\tilde{\mathbf{A}}^{(n_f)}(\mu_h^2)$ the operator matrix elements (partially known up to N³LO)

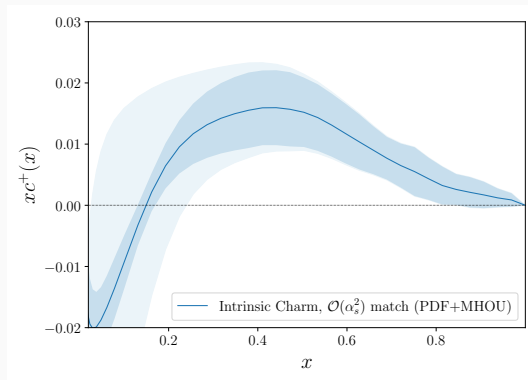
for backward evolution:

- invert $\tilde{\mathbf{E}}^{(n_f)}$: simple (invert RGE flow) ✓
- invert $\mathbf{R}^{(n_f)}$: simple (static matrix) ✓
- invert $\tilde{\mathbf{A}}^{(n_f)}$: expanded or exact

based on NNPDF4.0 [EPJC82.428]

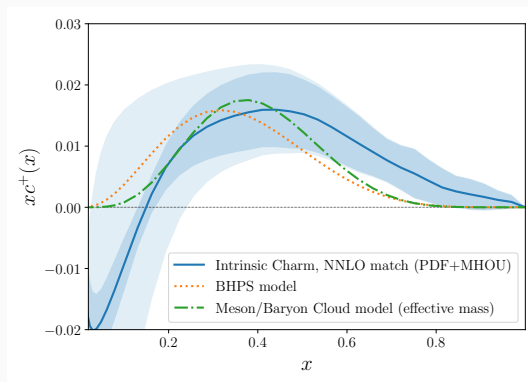


The PDF Plot



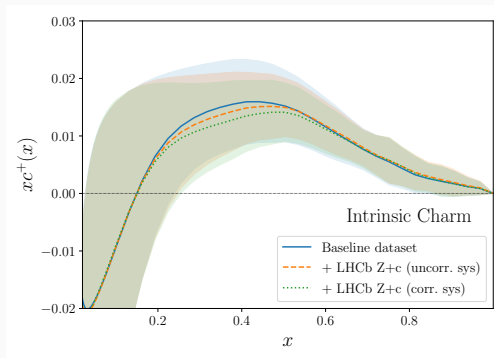
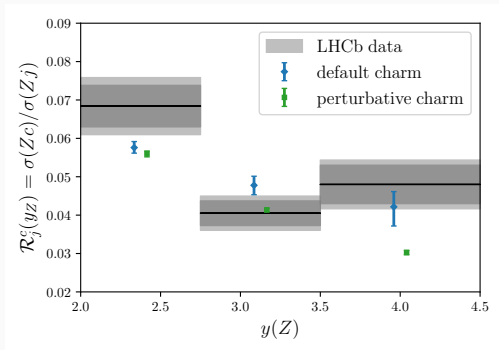
- in **3FNS** a valence-like peak is present
- for $x \leq 0.2$ the perturbative uncertainties are quite large
- the carried momentum fraction is within **1%**

The PDF Plot with Model Comparison

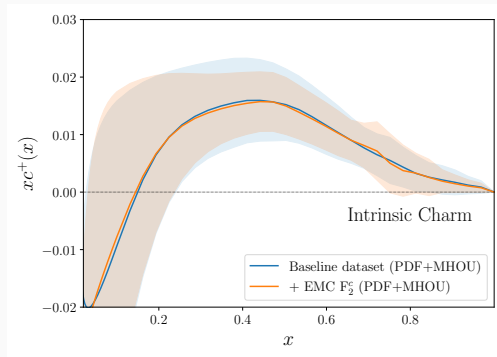


[BHPS] or [Meson/Baryon Cloud Model]

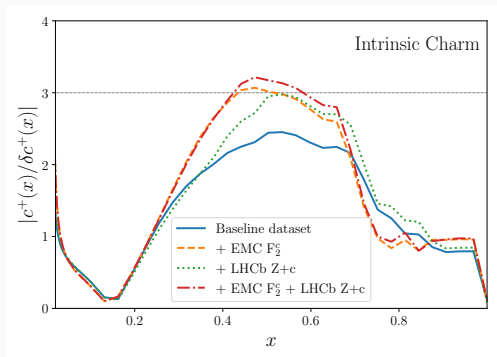
- in **3FNS** a valence-like peak is present
- for $x \leq 0.2$ the perturbative uncertainties are quite large
- the carried momentum fraction is within **1%**



- predict better recent measurement
- reweighting is consistent

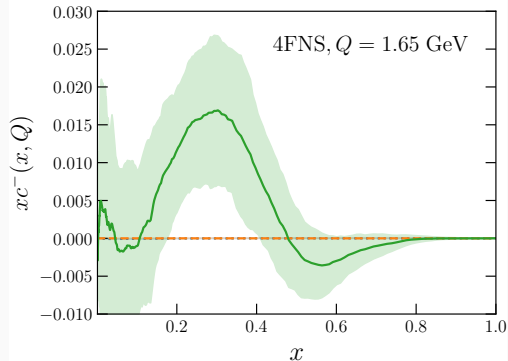
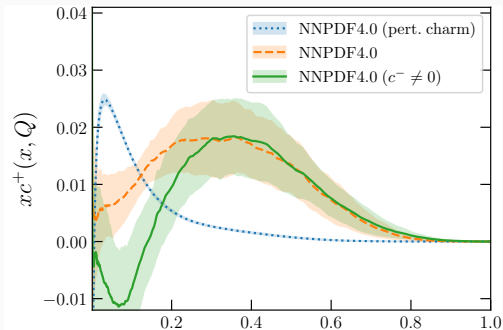


- direct measurement of F_2^c
- evidence for intrinsic charm claimed, but experiment disputed
- adding EMC data is consistent



- we find a **3σ** evidence of intrinsic charm
- result is **stable** with mass variation, dataset variation

Charm Asymmetry (PRELIMINARY!)



- also parametrize $c^- = c - \bar{c} \Rightarrow$ intrinsic!
- significance for baseline now $> 3\sigma$
- $\sim 1.5\sigma$ evidence for $c^- \neq 0$

Thank you!

3. Backup slides

The Command Line Interface (CLI)

The CLI (`pineappl`) serves for the everyday life questions:

- `convolute` - get the predictions for any PDF set including uncertainties
- `channels` - split the predictions into luminosity channels
- `orders` - split the predictions into perturbative orders
- `info` - access meta data
- `plot` - generate a (customizable) python plot script

and many more

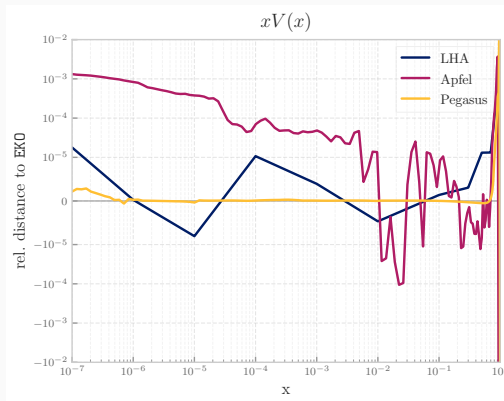
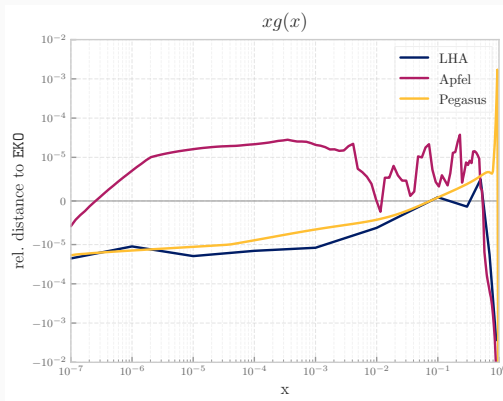
A Monte Carlo Example using C++

For a more complete example see: [examples directory](#) in PineAPPL repo

```
1 void fill_grid(PineAPPL::Grid &grid, std::size_t calls) {  
2     // MC loop  
3     for (std::size_t i = 0; i != calls; ++i) {  
4         // generate a phase-space point  
5         auto tmp = hadronic_pspgen(rng, 10.0, 7000.0);  
6         // compute kinematics  
7         double pt1 = sqrt((t * u / s));  
8         // apply cuts  
9         if (pt1 < 14.0) continue;  
10        // fill grid - here binning in |yll|  
11        auto weight = jacobian * matrix_element(s, t, u);  
12        grid.fill(x1, x2, q2, 0, fabs(yll), 0, weight);  
13    }  
14 }
```

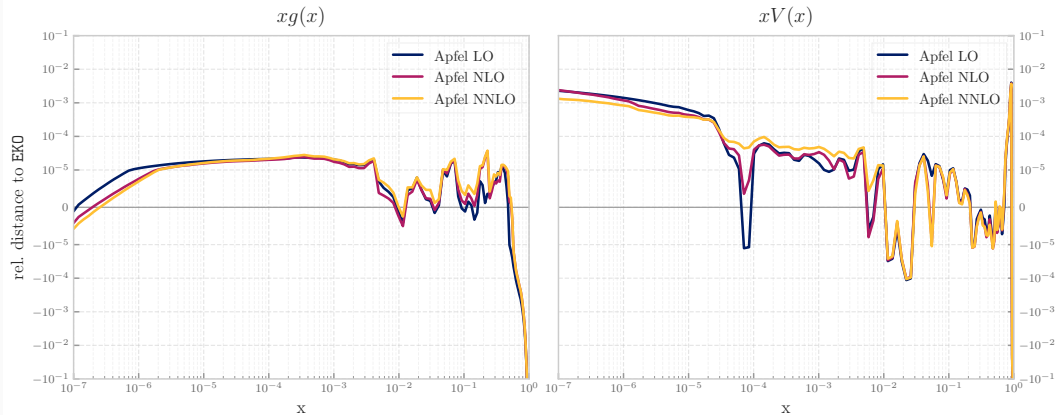
EKO Benchmarks

LHA benchmark [0204316][0511119]:

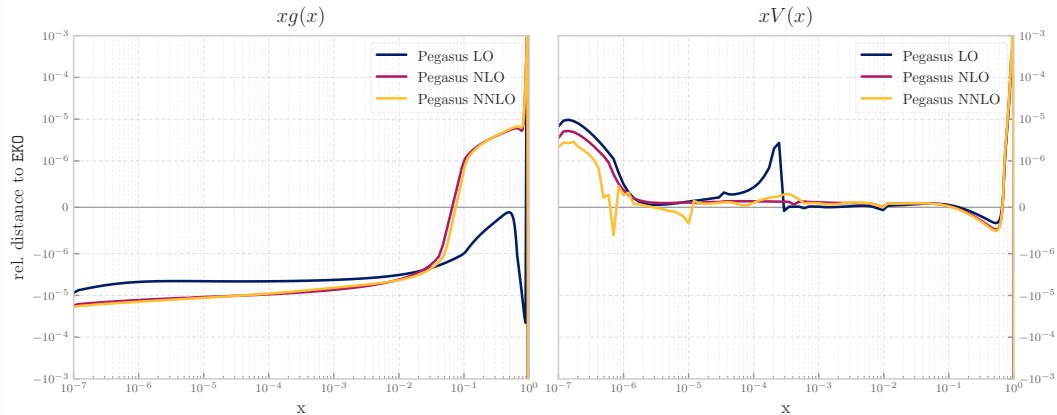


⇒ EKO is working!

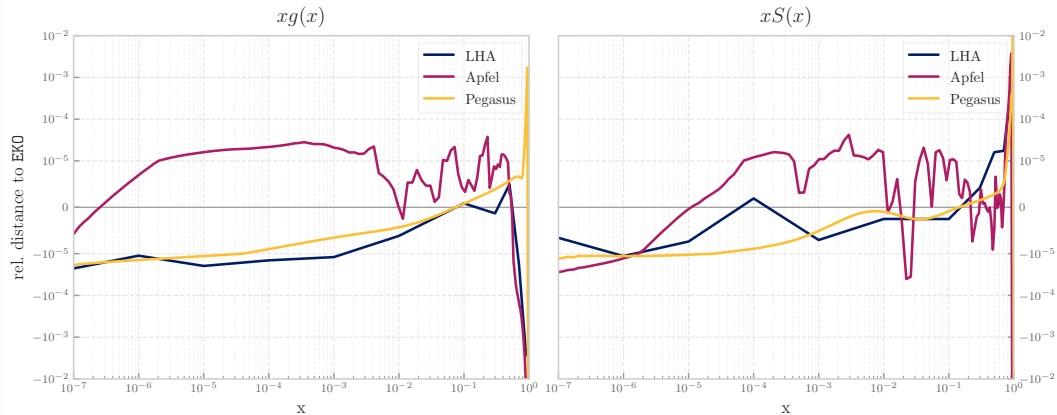
EKO APFEL benchmark



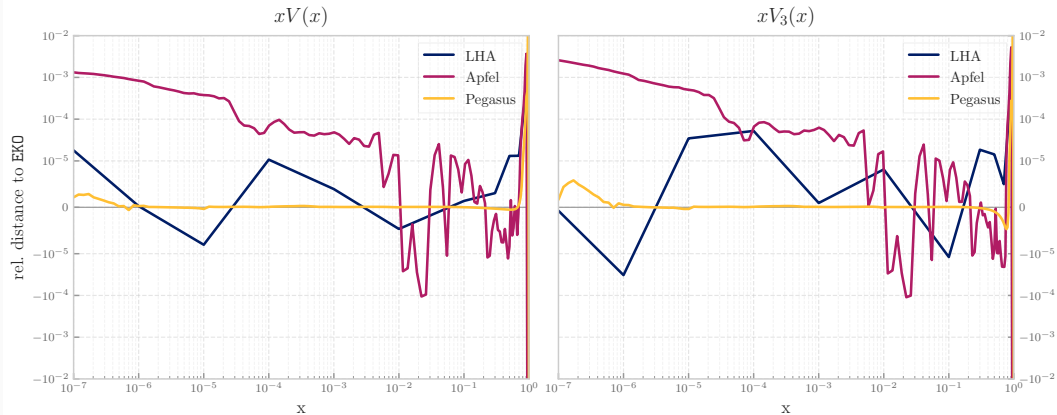
EKO PEGASUS benchmark



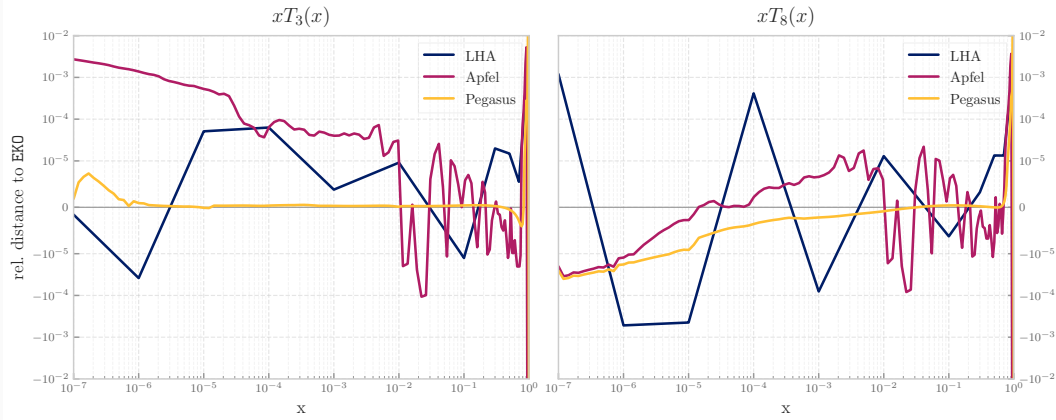
EKO LHA benchmark: g and Σ



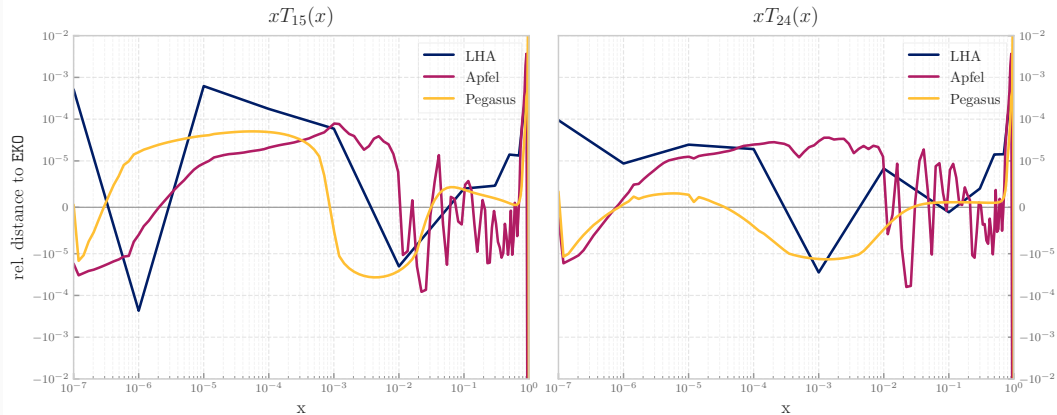
EKO LHA benchmark: V and V_3



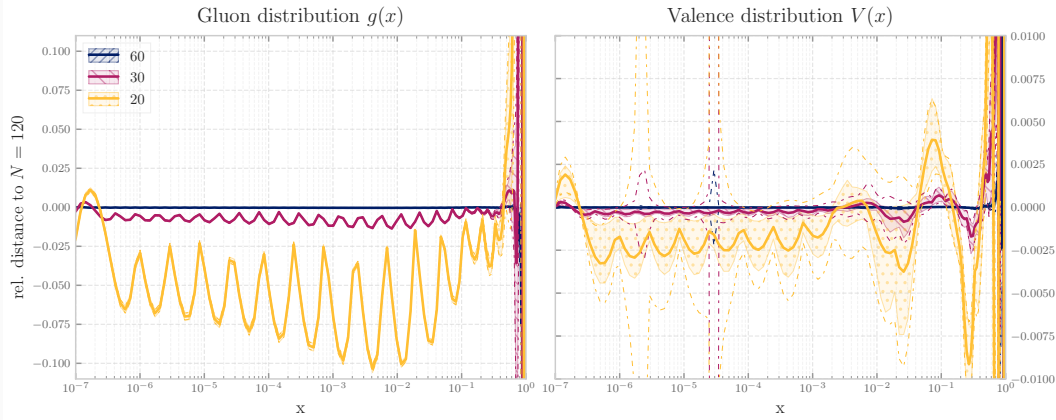
EKO LHA benchmark: T_3 and T_8



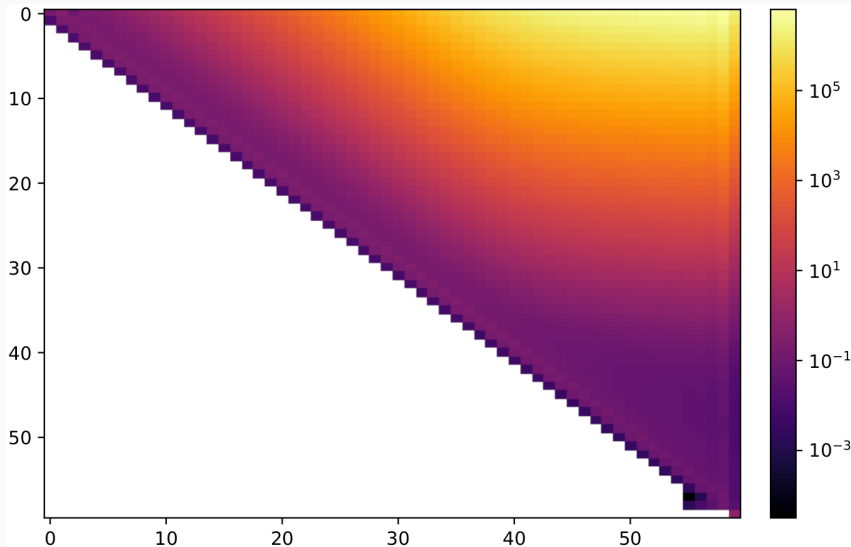
EKO LHA benchmark: T_{15} and T_{24}



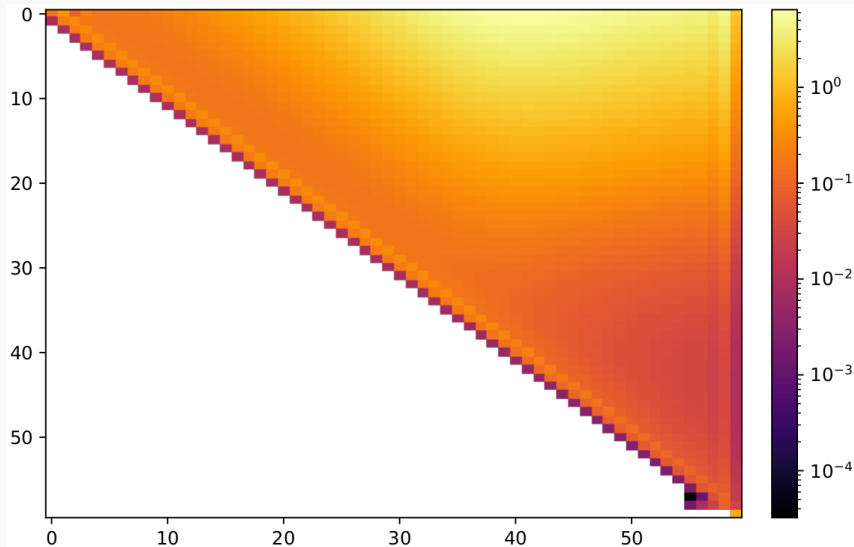
EKO Interpolation Error



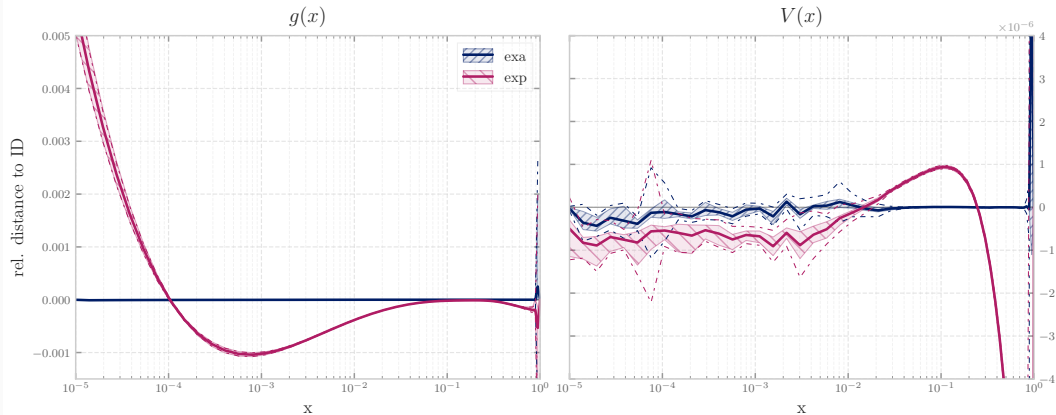
EKO Snapshot $S \leftarrow S$



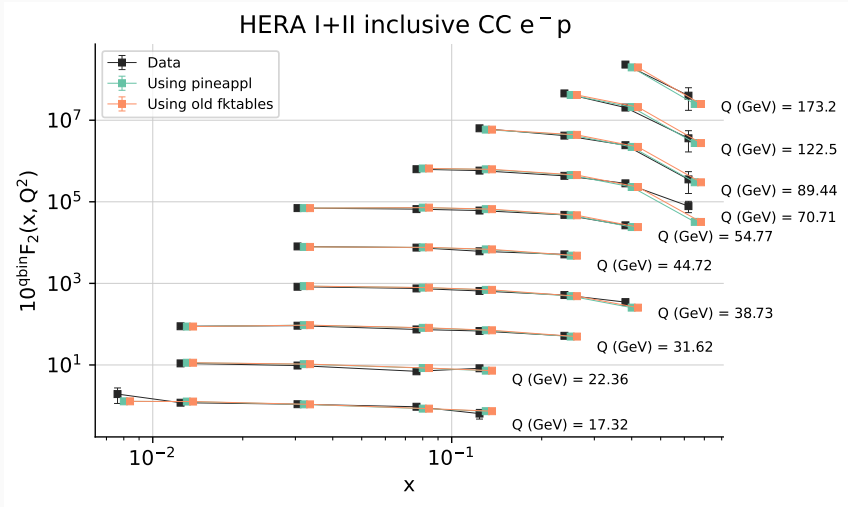
EKO Snapshot $V \leftarrow V$



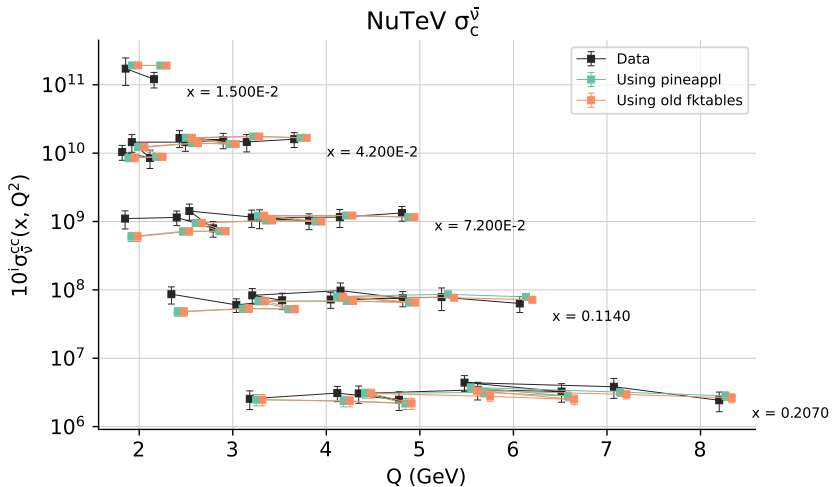
EKO Backward Evolution



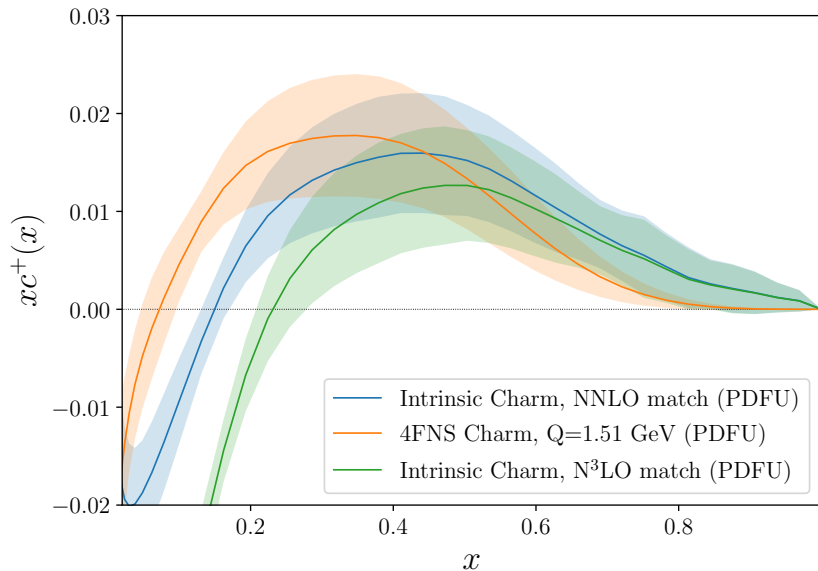
Comparison yadism against APFEL



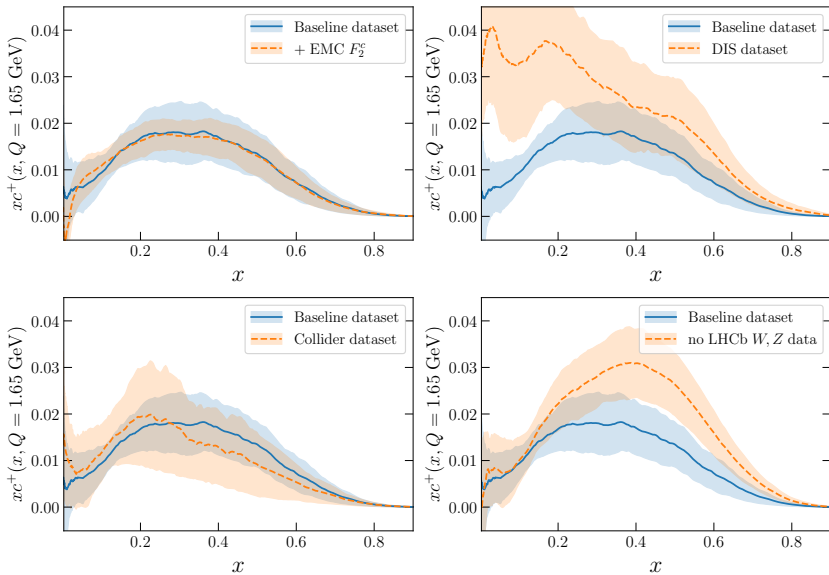
Comparison yadism against APFEL



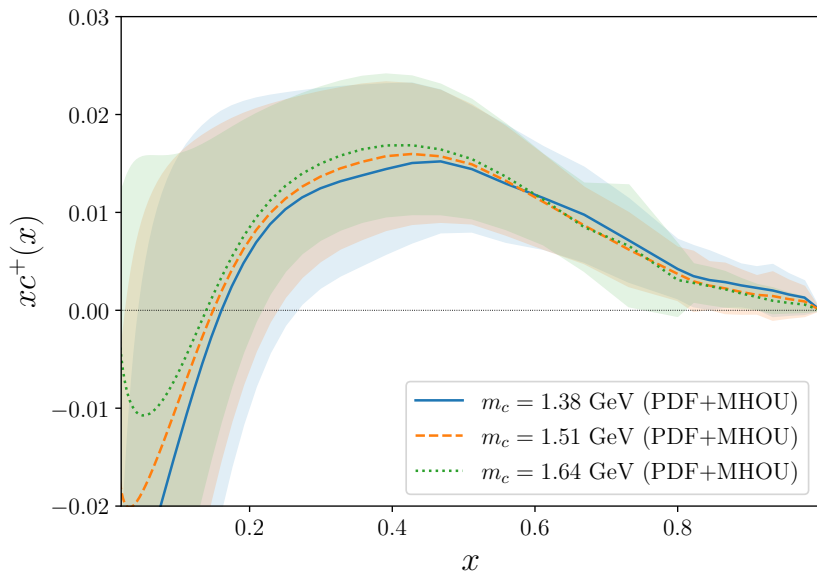
IC - uncertainties splitted



IC - dataset variation



IC - mass dependency



Roadmap towards N3LO PDFs

- DGLAP @ N3LO \rightarrow splitting functions approximation \checkmark
- DIS @ N3LO \rightarrow light + massive coefficient functions
- LHC observables @ NNLO + K-factors
- Inclusion of theory uncertainties both from scale variations and N3LO accuracy

N3LO singlet sector

Analytical calculations of the complete N3LO splitting functions are not available yet. Restricting to the singlet sector the known limits are:

large- n_f

- Davies, Vogt, Ruijl, Ueda, and Vermaseren. Large- n_f contributions to the four-loop splitting functions in QCD. [arXiv:1610.07477]

small- x

- Bonvini and Marzani. Four-loop splitting functions at small- x . [arXiv:1805.06460]
- Davies, Kom, Moch, and Vogt. Resummation of small- x double logarithms in QCD: inclusive deep-inelastic scattering. 2 2022. [arXiv:2202.10362].

large- x

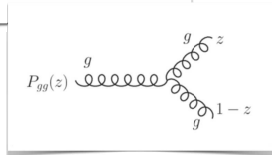
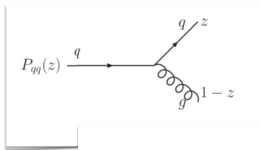
- Duhr, Mistlberger, and Vita. Soft integrals and soft anomalous dimensions at N3LO and beyond. [arXiv:2205.04493].
- Henn, Korchemsky, and Mistlberger. The full four-loop cusp anomalous dimension in $\mathcal{N} = 4$ super Yang-Mills and QCD. [arXiv:1911.10174].
- Soar, Moch, Vermaseren, and Vogt. On Higgs-exchange DIS, physical evolution kernels and fourth-order splitting functions at large x . [arXiv:0912.0369].

Moments

- Moch, Ruijl, Ueda, Vermaseren, and Vogt. Low moments of the four-loop splitting functions in QCD. [arXiv:2111.15561].

+2302.07593

- Theoretical inputs are not enough to determine the full expressions analytically.
- Need to parametrise the unknown part with sub-leading contributions.
- Uncertainties from this determination has to be taken into account during the fit.



Singlet

	n_f^0	n_f^1	n_f^2	n_f^3
$\gamma_{gg}^{(3)}$	✓	✓	✓	✓
$\gamma_{gq}^{(3)}$	✓	✓	✓	✓
$\gamma_{qq}^{(3)}$		✓	✓	✓
$\gamma_{qq,ps}^{(3)}$		✓	✓	✓

Approximation of $P_{gg}(x)$

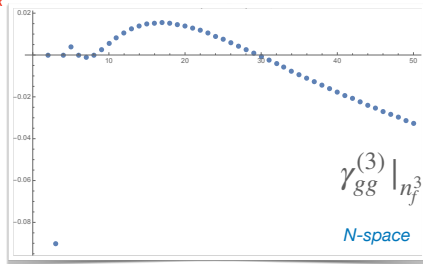
$$\tilde{f}(N) = \int_0^1 x^{N-1} f(x) dx$$

Rule of thumb:
small- $N \rightarrow$ small- x ,
large- $N \rightarrow$ large- x

Comparison w.r.t. known analytical part (%)

The approximation procedure is performed in Mellin space for each n_f part independently:

1. Parametrise the difference between the 4 known moments and known limits with 4 functions $f_i(N)$.
2. Varying the sub-leading unknown $f_i(N)$ to produce a large set of parameterisation candidates (≈ 70).
3. Reduce the number of samples discarding too wiggly parameterisations and looking at the most representative cases.



In $P_{gg}(x)$:

Theoretical constrain include:

- large- N :

$$\gamma_{gg}^{(3)}(N \rightarrow \infty) \approx \Gamma_A S_1(N) + B_{gg} + \mathcal{O}\left(\frac{\ln(N)}{N}\right)$$

- small- N pole at $N = 0$, and $N = 1$ (leading contribution):

$$\gamma_{gg}^{(3)}(N \rightarrow 1) \approx C_4 \frac{1}{(N-1)^4} + C_3 \frac{1}{(N-1)^3} + \mathcal{O}((N-1)^{-2})$$

- 4 lowest moments $N = \{2, 4, 6, 8\}$

Solve the constrain given by the 4 known Mellin

moments with many different candidates $\{f_1, f_2, f_3, f_4\}$:

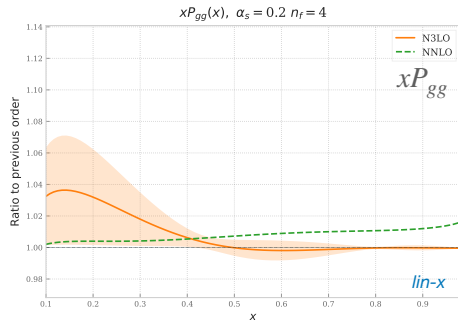
$$f_1 = \frac{S_1(N)}{N}, f_2 = \frac{1}{(N-1)^2}$$

$$f_3 = \left\{ \frac{1}{(N-1)}, \frac{1}{N} \right\}$$

$$f_4 = \left\{ \frac{1}{(N-1)}, \frac{1}{N^4}, \frac{1}{N^3}, \frac{1}{N^2}, \frac{1}{N}, \frac{1}{(N+1)^3}, \frac{1}{(N+1)^2}, \right.$$

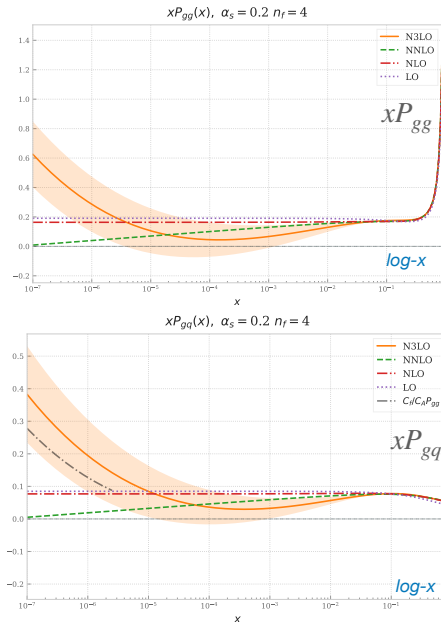
$$\left. \frac{1}{N+1}, \frac{1}{N+2}, \mathcal{M}[\ln(1-x)], \mathcal{M}[(1-x)\ln(1-x)], \frac{S_1(N)}{N^2} \right\}$$

N3LO singlet sector



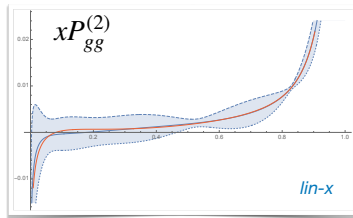
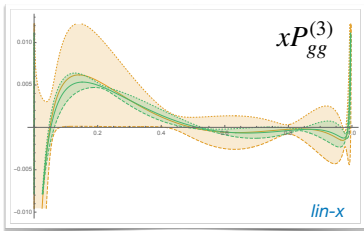
- Singlet approximated splitting functions are less constrained by the known limits. The coefficients of $1/x \ln^2(x)$, $1/x \ln(x)$ play a crucial role in the small- x region.
- Uncertainty arising from the approximation is not negligible.
- Off diagonal terms P_{qg} , P_{gq} are more difficult to estimate (large- N goes to 0).
- Only theoretical inputs are considered.
- All the implemented approximations respect momentum sum rules.

PRELIMINARY RESULTS

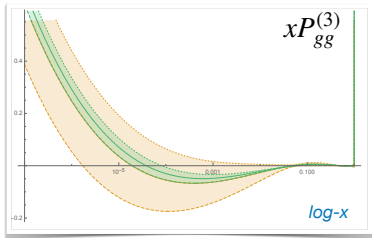
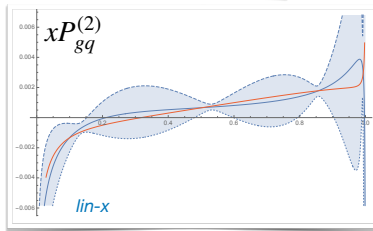


Approximation checks

1. A possible way to validate the procedure is to **reproduce the known NNLO** singlet splitting functions using the very similar constrain that we have right now on the N3LO ones.



— Approx
— Analytic

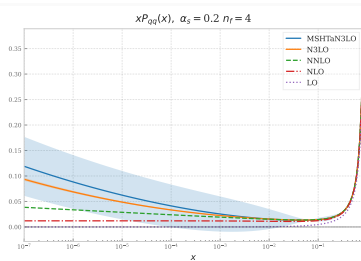
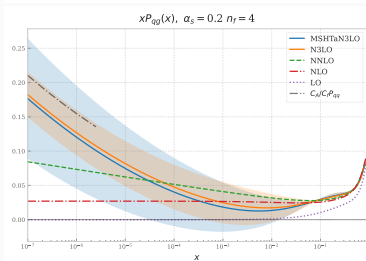
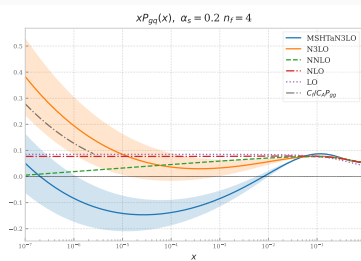
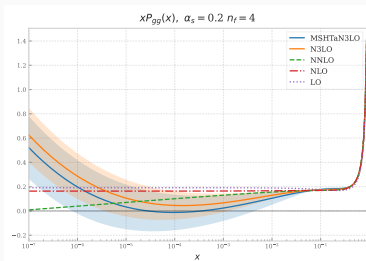


— 4 Moments fixed

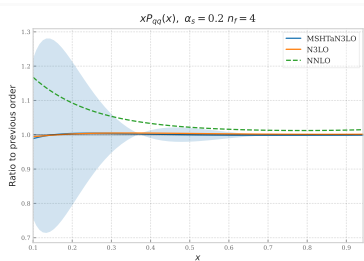
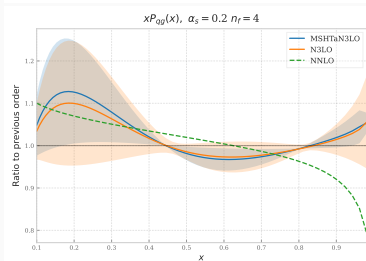
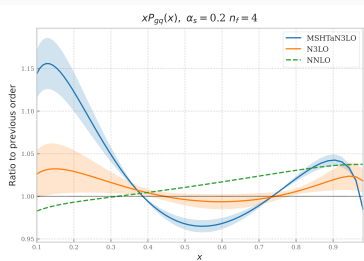
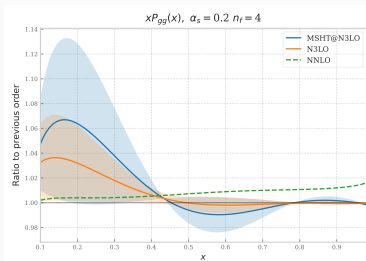
— + N=3 fixed

2. Another way to validate the results is to **interpolate the known moments**, and construct a more constrained parametrisation now including 5/6 moments. If the procedure is working (the samples are varied enough) the uncertainty band obtained in this way should be small than the default one.

Comparison with MSHT [EPJC83.185] at small-x - PRELIMINARY!



Comparison with MSHT [EPJC83.185] at large- x - PRELIMINARY!



- light coefficient functions [\[VVM05\]](#), [\[MVV05\]](#), [\[MV00\]](#), [\[MRV08\]](#), [\[MVV09\]](#) ✓
- massive coefficient functions → approximation in MSc thesis of N. Laurenti ✓
- FONLL [\[FLNR10\]](#) prescription → MSc thesis of S. Zanioli, A. Barontini ✓
or better: use “Numerical FONLL” (thanks to new pipeline)

$$F^{\text{FONLL}} = F^{(n_f+1)} + F^{(n_f)} - F^{(n_f,0)}$$

N3LO non singlet sector

non-singlet 4-loop Anomalous Dimensions

	n_f^0	n_f^1	n_f^2	n_f^3
$\gamma_{ns,-}^{(3)}$	✓	✓	✓	✓
$\gamma_{ns,+}^{(3)}$	✓	✓	✓	✓
$\gamma_{ns,s}^{(3)}$		✓	✓	

- Estimation of the N3LO anomalous dimensions is based on the best available theoretical constraints:

- large-N:

$$\gamma_{ns}^{(3)}(N \rightarrow \infty) \approx \Gamma_f S_1(N) + B + C \frac{S_1(N)}{N} + D \frac{1}{N} + \mathcal{O}\left(\frac{\ln(N)}{N^2}\right)$$

- small-N:

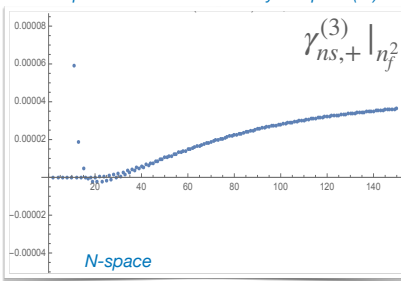
$$\gamma_{ns}^{(3)}(N \rightarrow 0) \approx \sum_{i=1}^7 C_i \frac{1}{N^i}$$

- 8 lowest Mellin moments

- For more details on the procedure used see [EKO N3LO ad documentation](#)

- Non singlet approximated spitting functions are compatible with the known analytical (and much more complex) parts within numerical accuracy.

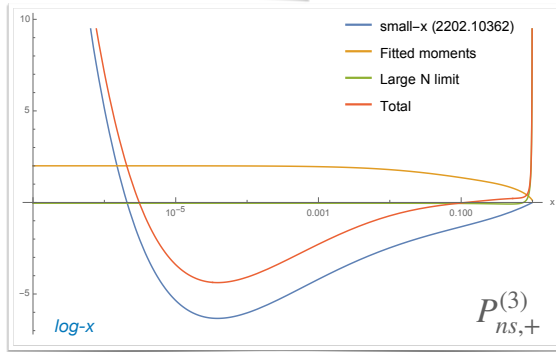
Comparison w.r.t. known analytical part (%)



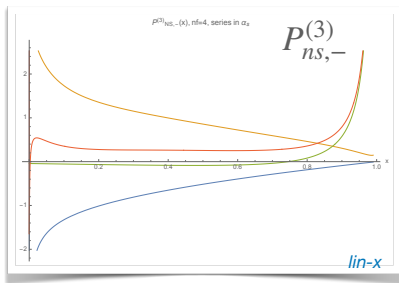
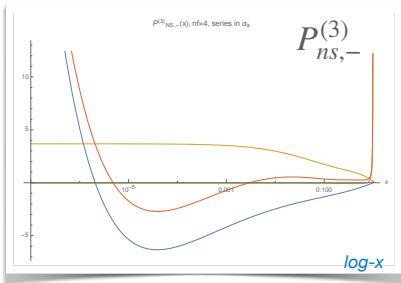
Main references:

- Moch, Ruijl, Ueda, Vermaseren, Vogt [[arXiv:1707.08315](#)].
- Davies, Vogt, Ruijl, Ueda, Vermaseren. [[arXiv:1610.07477](#)]
- Davies, Kom, Moch, Vogt. [[arXiv:2202.10362](#)].

Rule of thumb:
small-N \rightarrow small-x,
large-N \rightarrow large-x



N3LO non singlet



- small-x (2202.10362)
- Fitted moments
- Large N limit
- Total

