

# fastNLO for Jetproduction in Diffractive DIS

D. Britzger, F. Ceccopieri

H1 Collaboration Meeting 2012  
MPI Munich  
September 12 2012



# fastNLO

## Motivation

- Calculation of jet cross sections in higher orders are time consuming
- Often: Calculations have to be repeated for same measurement but different PDFs or  $\alpha_s$  (e.g. PDF uncertainties, scale variations, etc...)
- Inclusion of jet-data in a fit (e.g. PDF or  $\alpha_s$ ) requires very fast recomputation of almost the same cross sections

**Need procedure for fast repeated computations of NLO cross sections**

use: *fast***NLO**

# fastNLO

## fastNLO concept

- Jet-production cross section in DIS

$$\sigma = \sum_{a,n} \int_0^1 dx \alpha_s^n(\mu_r) \cdot c_{a,n}\left(\frac{x_{Bj}}{x}, \mu_r, \mu_f\right) \cdot f_a(x, \mu_f)$$

- Introduce interpolation kernels for 'x' using Eigenfunctions  $E_i(x)$

$$f_a(x) \cong \sum_i f_a(x_i) \cdot E^{(i)}(x)$$

- Single PDF is replaced by a linear combination of interpolation kernels

- We can calculate once a table with perturbative coeff. using e.g. nlojet++

- Cross section is then just a sum over all interpolation nodes  $\sigma_{DIS}^{Bin} = \sum_{i,a,n,m} \alpha_s^n(\mu^{(m)}) \cdot f_a(x_1^{(i)}, \mu^{(m)}) \cdot \tilde{\sigma}_{a,n}^{(i)(m)}$

## More features of fastNLO

e.g. Scale interpolation, PDF reweighting, scale independent contr., etc..

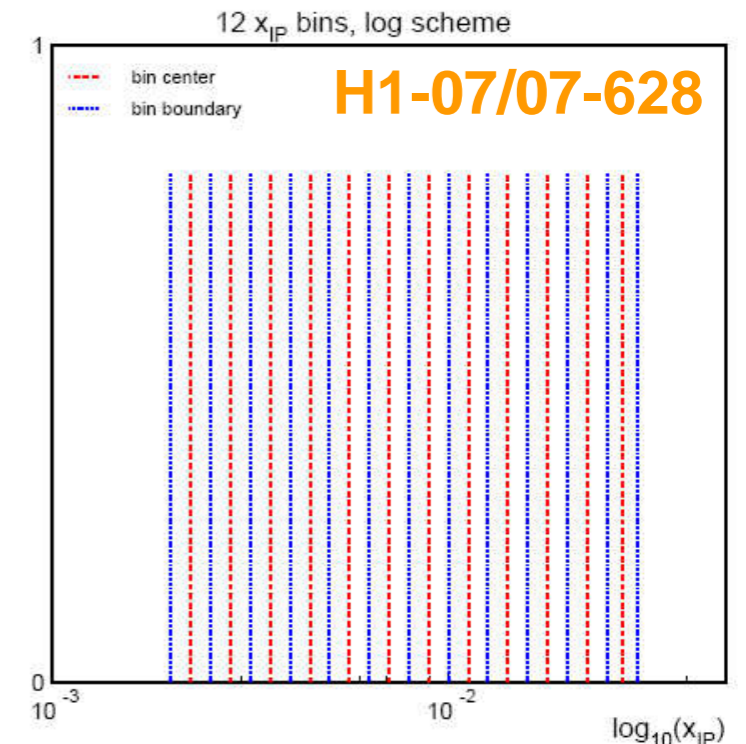
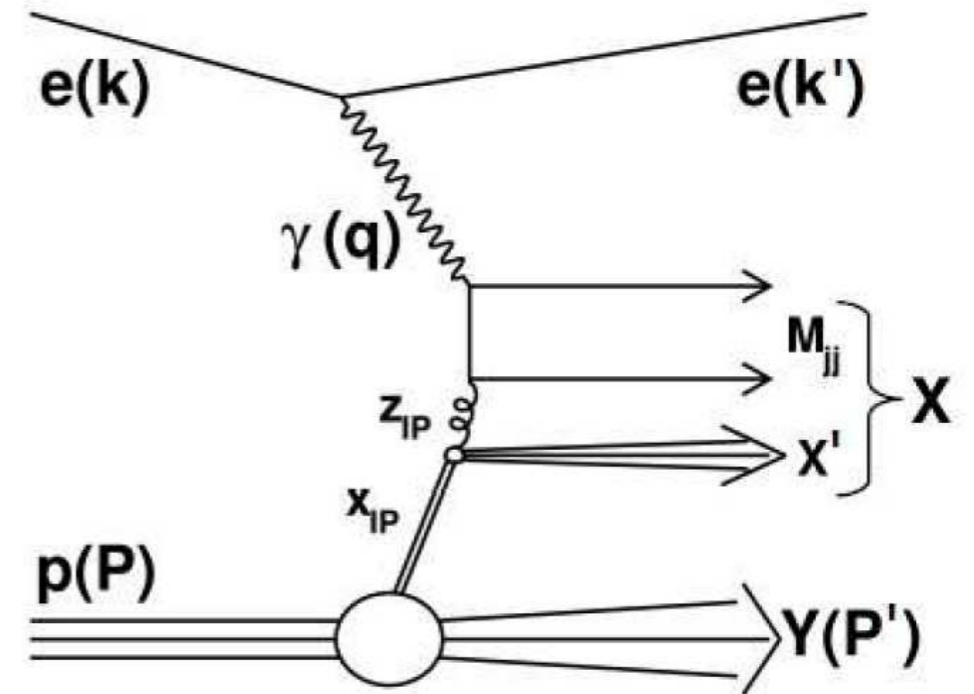
# Jet production in diffractive DIS

## Jet production in diffractive DIS (t=0)

$$\sigma = \sum_{a,n} \int_0^1 dx_{IP} \int_0^1 dz_{IP} \alpha_s^n \cdot c_{a,n} \cdot f_a(x_{IP}, z_{IP}, \mu_f)$$

## Standard method of calculating NLO cross sections: 'Slicing method'

- Riemann-Integration of  $dx_{IP}$ 
  - Discretize the  $x_{IP}$  range into k bins (k~10)
- Repeated cross section calculation with reduced hadron energy for each slice of  $x_{IP}$ 
  - fixed value of  $x_{IP,i}$
  - At reduced center of mass energy of  $\sqrt{s} = x_{IP} \cdot 4E_p E_e$
  - slice-width  $\Delta x_{IP}$

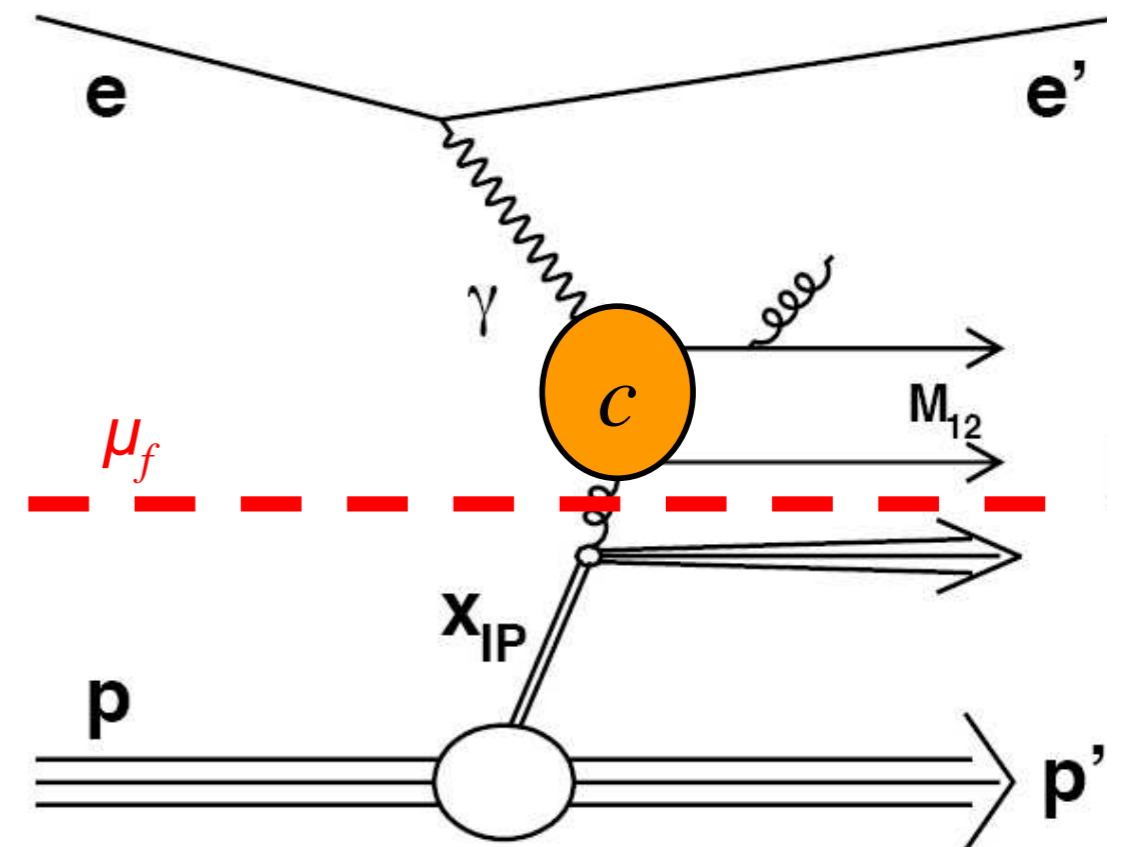


$$\int_0^1 dx_{IP} f_{IP/a}(x_{IP,i}) \sigma_{IP}(x_{IP}) \cong \sum_k \Delta x_{IP,i} f_{IP/a}(x_{IP,i}) \sigma_{IP}(x_{IP})$$

# Jet production in diffractive DIS

**Perturbative coefficients have only dependence on momentum fraction**

- No direct dependence of the two momentum fractions  $x_{IP}$  and  $z_{IP}$
- Each slice calculates basically **same coefficients  $c$**
- Factorization is independent of incoming parton



**Perform  $x_{IP}$ -Integration a-posteriori**  
**Calculate one fastNLO table with hadron energy = proton energy**

**Calculation of only one single fastNLO table is needed**

- Very high statistical precision
- No limitations in  $x_{IP}$  integration

# Proof of concept

Compare two cross section calculations  
for one single fixed  $x_{IP}$  slice at

$$x_{IP} = 10^{-1.5}$$

Cross section contribution with reduced  
CME (standard slicing method)

$$\begin{aligned} \text{Hadron} &= \text{Pomeron} \\ E_h &= x_{IP} \cdot E_P \end{aligned}$$

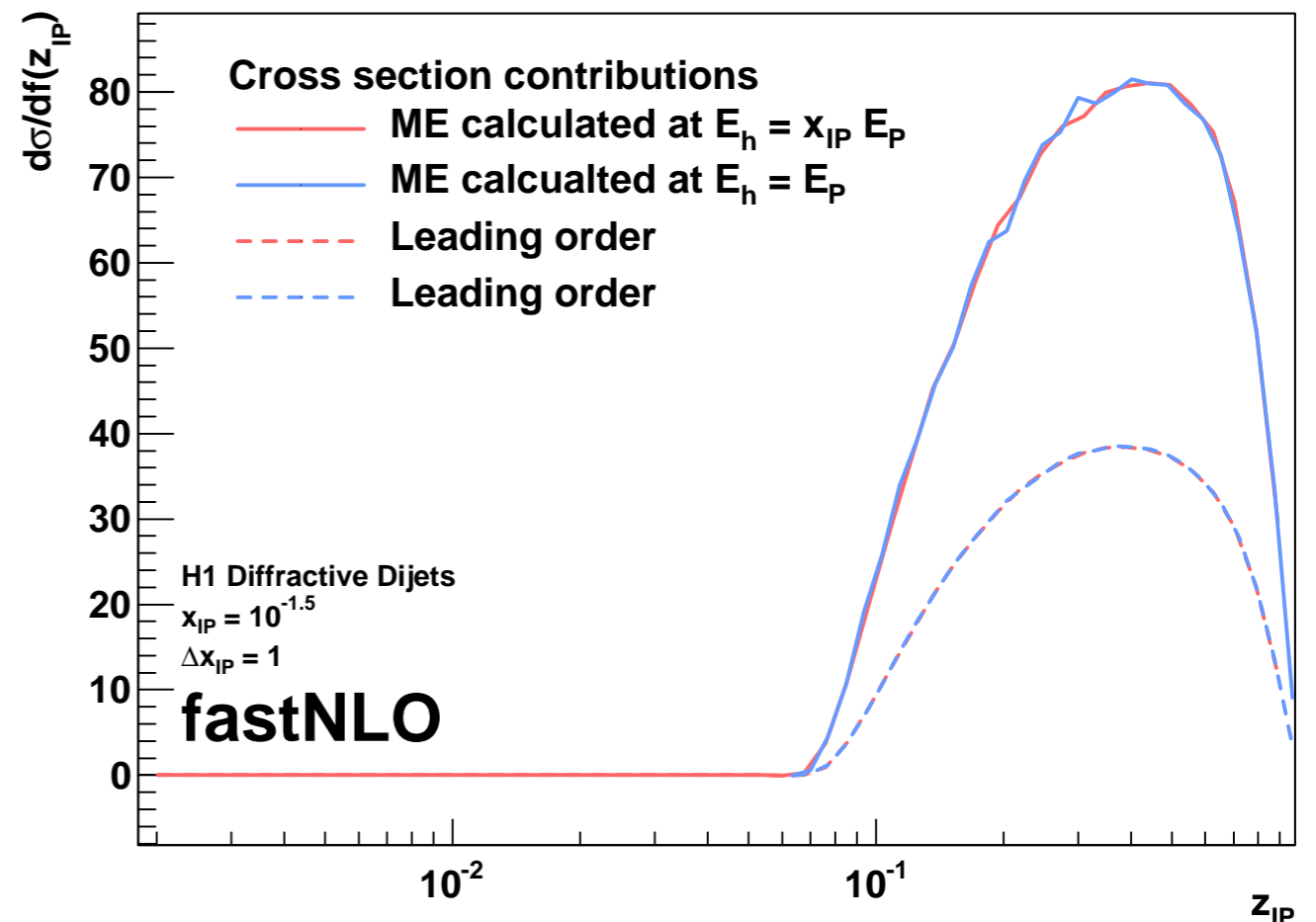
Calculate matrix elements using proton  
energy (new method)

$$\begin{aligned} \text{Hadron} &= \text{Proton} \\ E_h &= E_P \end{aligned}$$

Compare  $z_{IP}$  dependent cross section  
contribution

$$\begin{aligned} z_{IP} &= x_{\text{hadron}} \\ z_{IP} &= x_{\text{hadron}} / x_{IP} < 1 \end{aligned}$$

$x_{\text{hadron}}$  being the momentum fraction of the parton  
wrt. to the incoming hadron (Pomeron) momentum



**Perfect agreement**

➤ Perturbative coefficients are identical for each  $x_{IP}$  slice

# Jets in diffractive DIS with fastNLO

## 1. Fixed center-of-mass calculation

- Calculate only one fastNLO table at proton energy  $E_p$
- Increased number of x-nodes in low-x region

## 2. Adapt the slicing method

- Define arbitrary  $x_{IP}$  slicing
- Calculate cross section by Riemann-integrating  $x_{IP}$
- Integrate x wrt.  $E_p$
- Formulae calculated properly by Federico !!!

$$\sigma_{n,a} = \sum_k \Delta x_{IP,k} \int_0^{x_{IP,k}} \frac{dx}{x_{IP,k}} \alpha_s^n \cdot c_0(x) \cdot f_a(x_{IP,k}, z_{IP} = \frac{x}{x_{IP,k}}, \mu_f)$$

## Integral becomes a standard fastNLO evaluation

Upper integration interval needs to be respected properly

- x-integration runs over discrete x-nodes
- Uppermost x-node is weighted according to its range inside/outside of the integration interval

FastNLO procedure improves previously used approach

# Code example for pre-calculated fastNLO table

```
// FastNLO example code in C++ for reading
//   H1 diffractive dijets dP*_T,1
//   Eur.Phys.J. C70 (2010) 15

FastNLODiffH12006FitB fnlodiff( "fnhd1012.tab" );
fnlodiff.SetXPomLinSlicing( 30, 0.0, 0.1 );
vector<double> xs = fnlodiff.GetDiffCrossSection();

// optional printout
fnlodiff.PrintCrossSections();
```

```
// output
*   This is a single-differential table in p*_T,1.
*
*   ---  p*_T,1  ---          - Bin -          -- XS-FNLO --
*       4.000 -      6.500      0           9.4173e+01
*       6.500 -      8.500      1           6.6364e+01
*       8.500 -     12.000      2           1.6610e+01
```

Pretty easy to use (if fastNLO table is already calculated)



# Application: Scale study

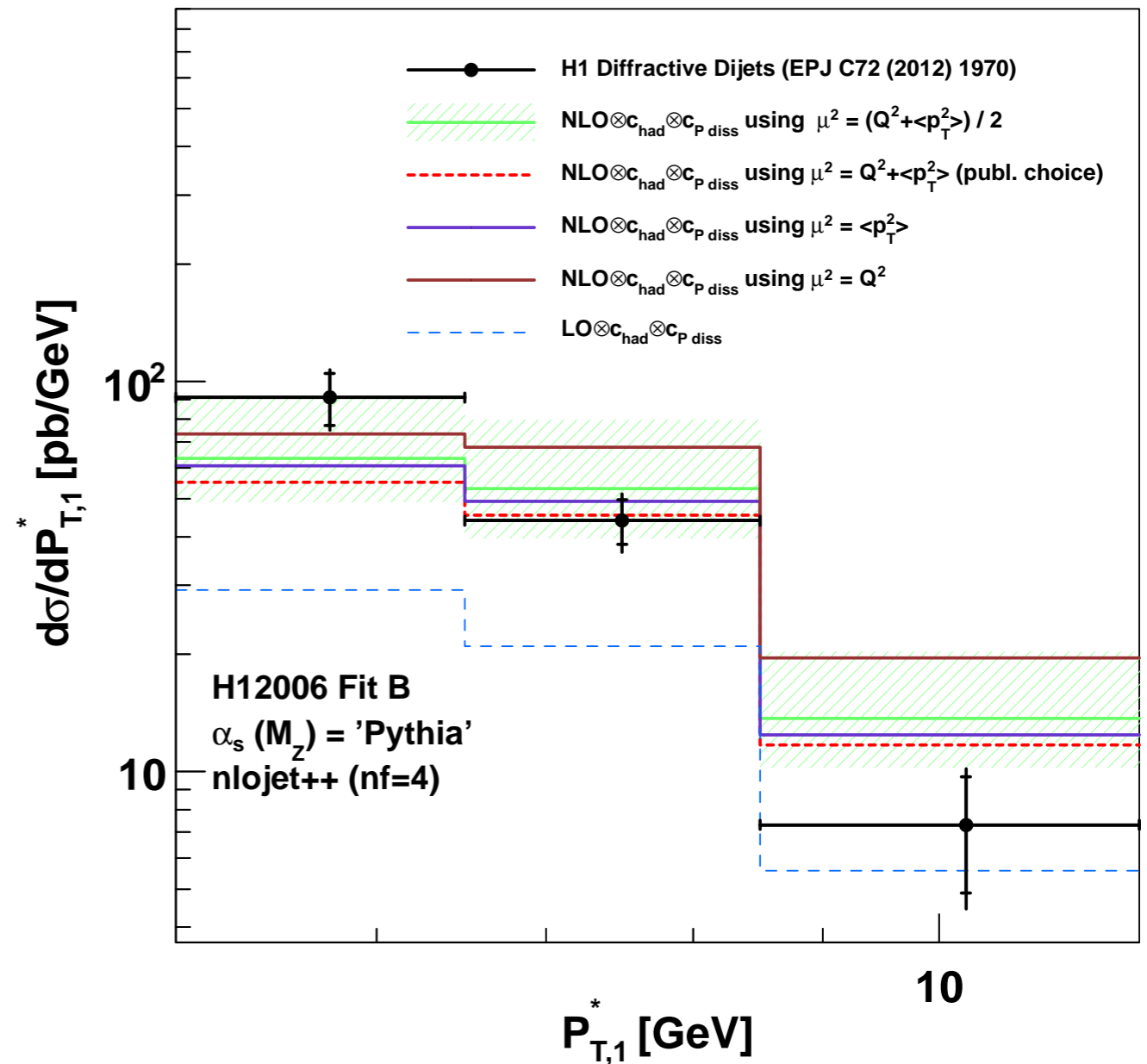
## Dijet production in diffractive DIS

(H1) EPJ C72 (2012) 1970

- Calculations available for
  - $d\sigma/dQ^2$
  - $d\sigma/dp_{T,1}^*$
- Possibility to derive by restricting  $x_{IP}$  integration interval
  - $d\sigma/dx_{IP}$

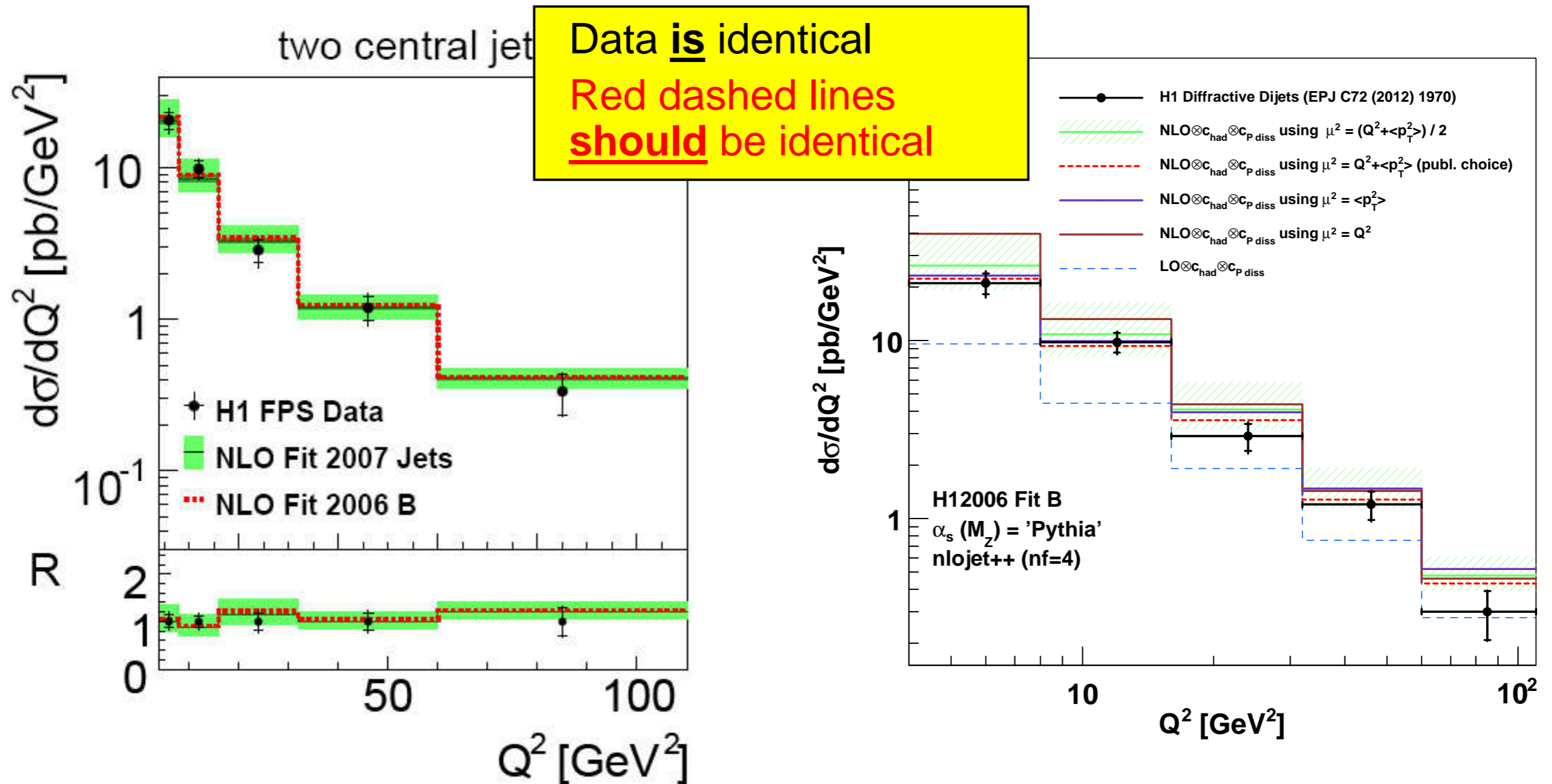
## Full fastNLO features accessible

- Scale studies are easily possible
- Compare various scale choices
- Interface various DPDFs
- Direct access to k-factors



**NEW:** Facilitate inclusion of diffractive jets in DPDF fits  
 -> Will help to constrain the gluon in diffractive PDFs

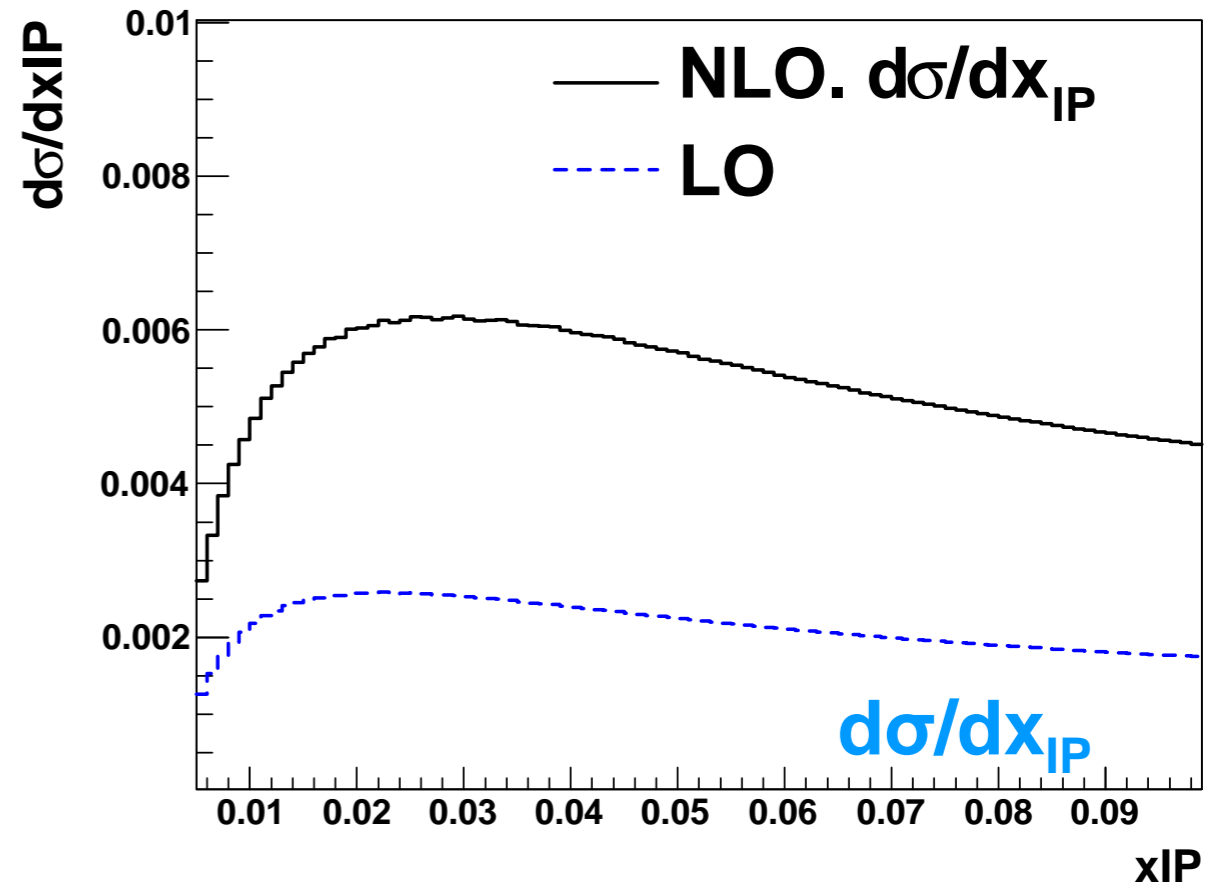
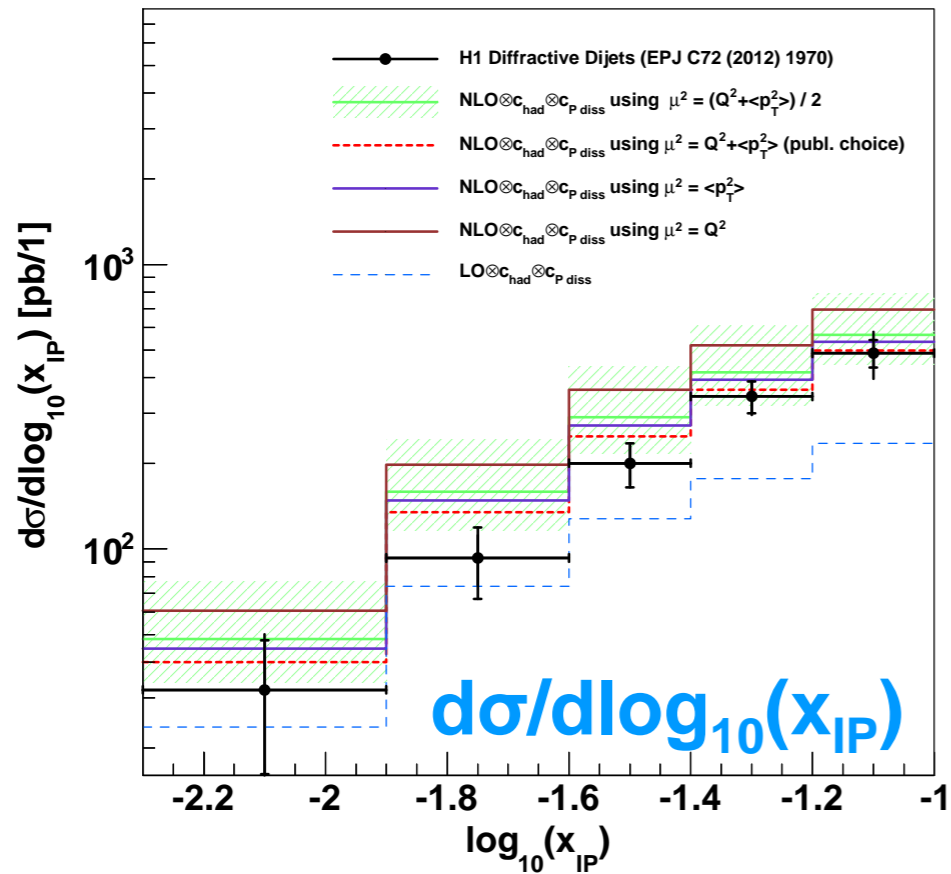
# Test scenario: Diffractive Dijets $dQ^2$



## Small differences

- Automatic (but wrong) nlojet++ interpolation of NLO subtraction terms
- Jet-finder recombination scheme (~1%)
- Different (but very fine) xIP (and dxIP) slicing (~0-2%)
- Limited statistic (26×6·106) vs. very high statistic (2·10<sup>10</sup>) (<1%)

# Which slicing is preferred?



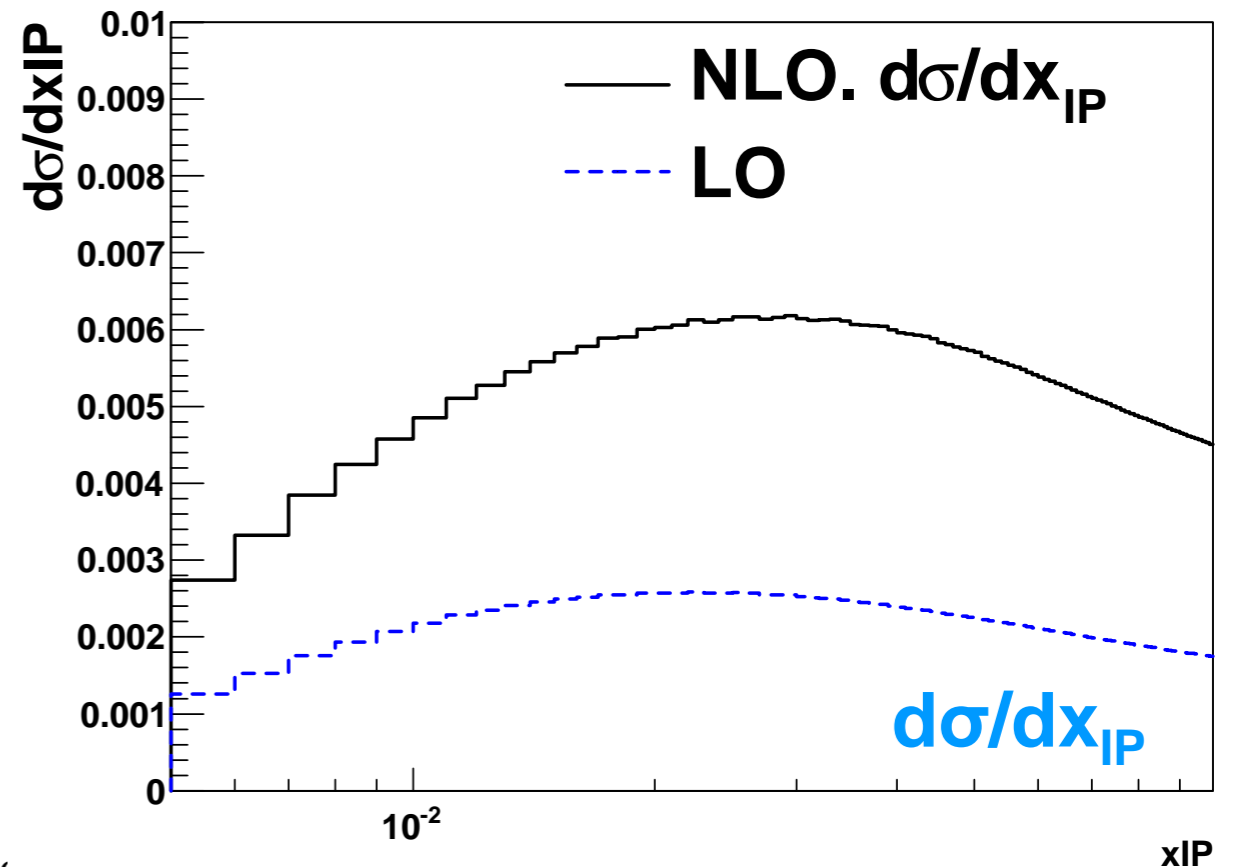
Linear or logarithmic slicing of  $x_{IP}$  ?

Both are available

```
fnloddiff.SetXPomLinSlicing( 30, 0.0, 0.1 );
fnloddiff.SetXPomLogSlicing( 30, 0.0, 0.1 );
```

Or any other

```
double xp[] = {...};
double dxp[] = {...};
fnloddiff.SetXPomSlicing( 30, xp, dxp );
```



# Convergence of slicing method

## Study cross section dependence on number of $x_{IP}$ slices

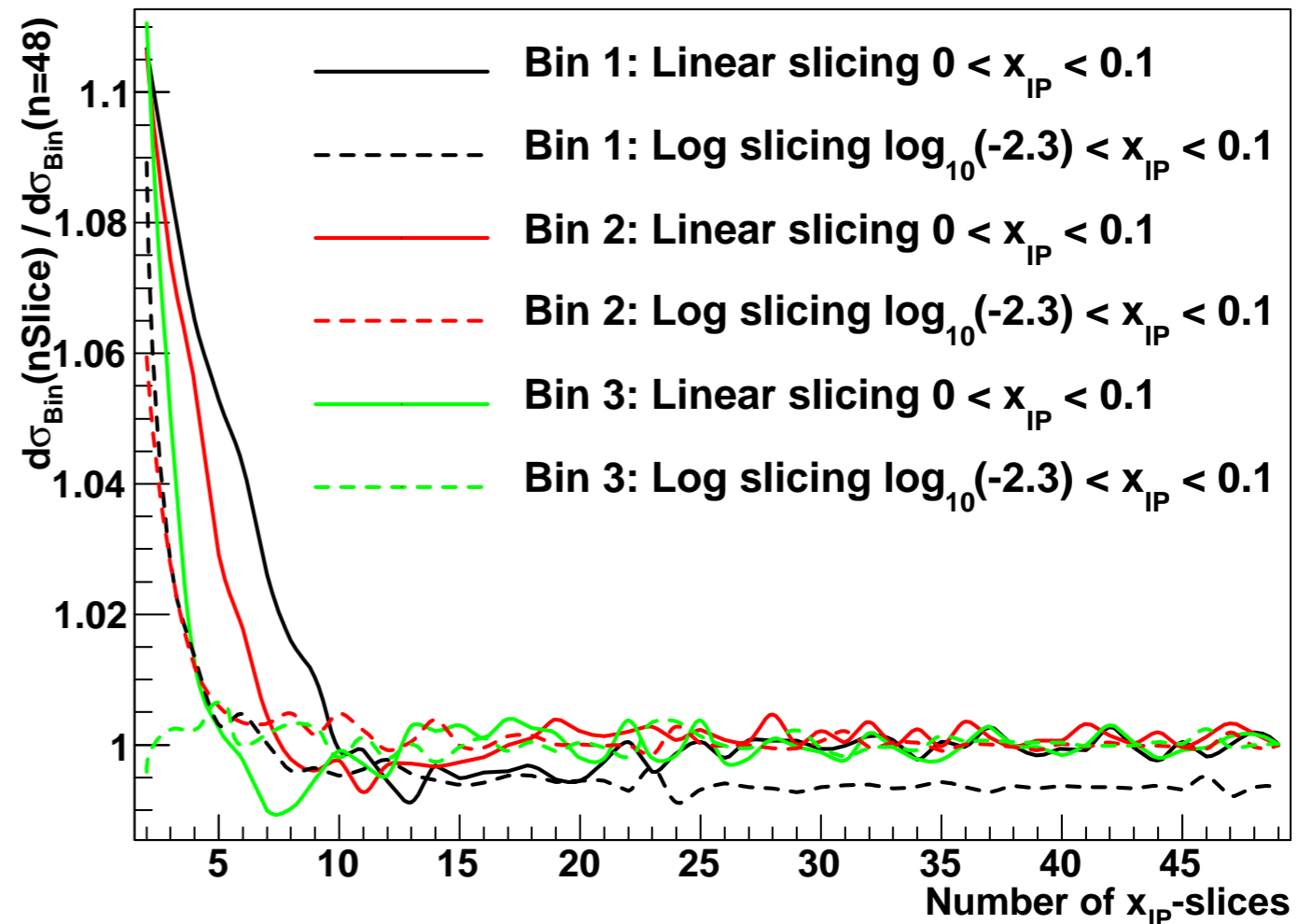
Compare:  
linear and logarithmic slicing

## Convergence after 10 slices Logarithmic slicing seems to converge faster

But one must know lowest  $x_{IP}$   
-> Otherwise bias!

## Fluctuations

- arise from discrete  $x_{hadron}$  interpolation within fastNLO
- despite (simple) 'smoothing' at upper integration interval at  $x_{hadron} \sim x_{IP,k}$



- **Very fast convergence of cross section**
- **FastNLO accuracy (for this table) could be estimated to  $< 0.25\%$** 
  - Accuracy smaller than cut on  $\log(x_{IP,min})$  for log slicing

# What about $d\sigma/dz_{IP}$ ??

## Warning!

No  $d\sigma/dz_{IP}$  available through restricting  $z_{IP}$  integration interval

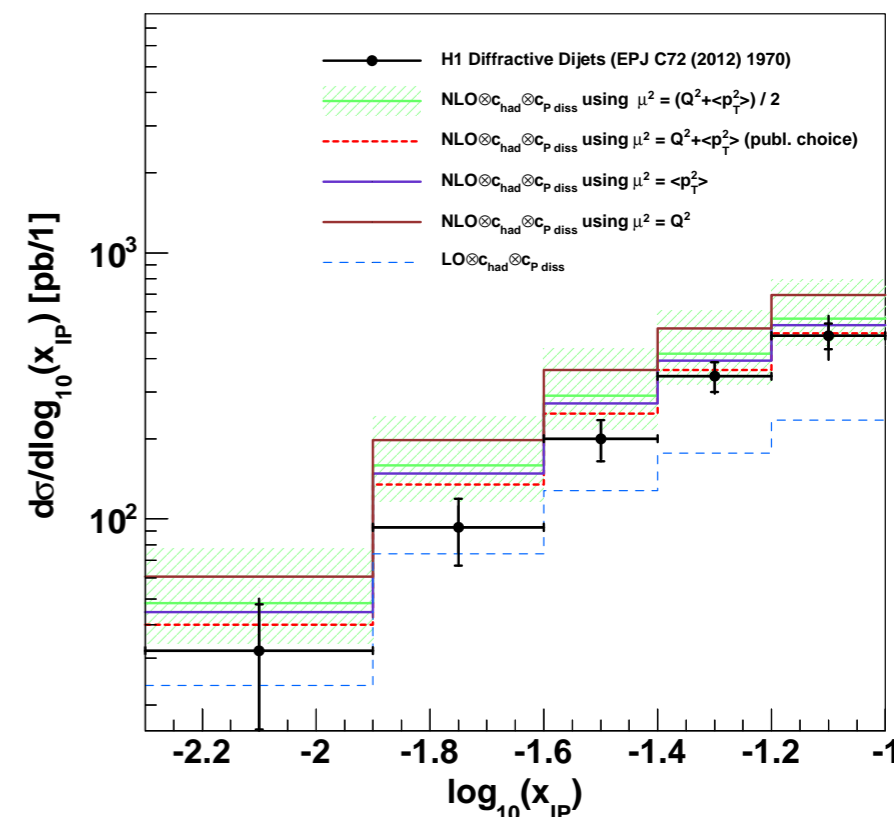
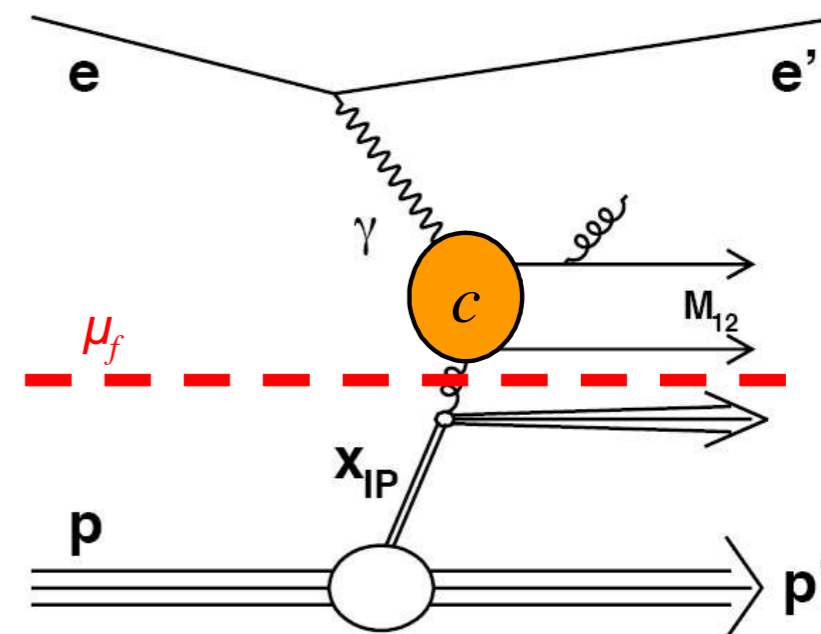
$z_{IP} \neq \xi_{IP}$  in higher orders

$z_{IP}$  = Parton momentum fraction compared to Hadron (Pomeron)

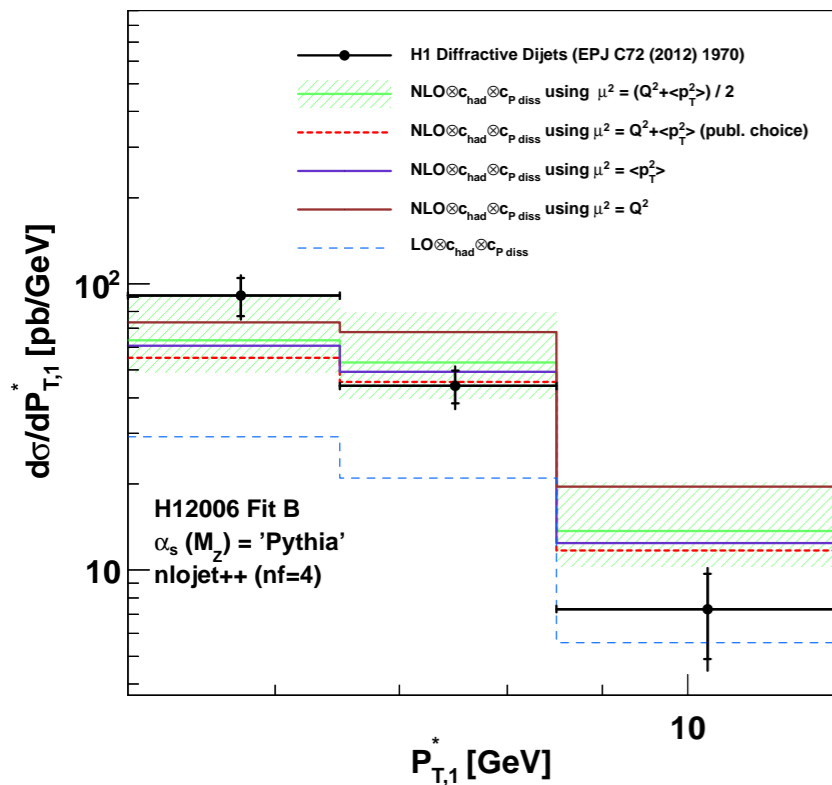
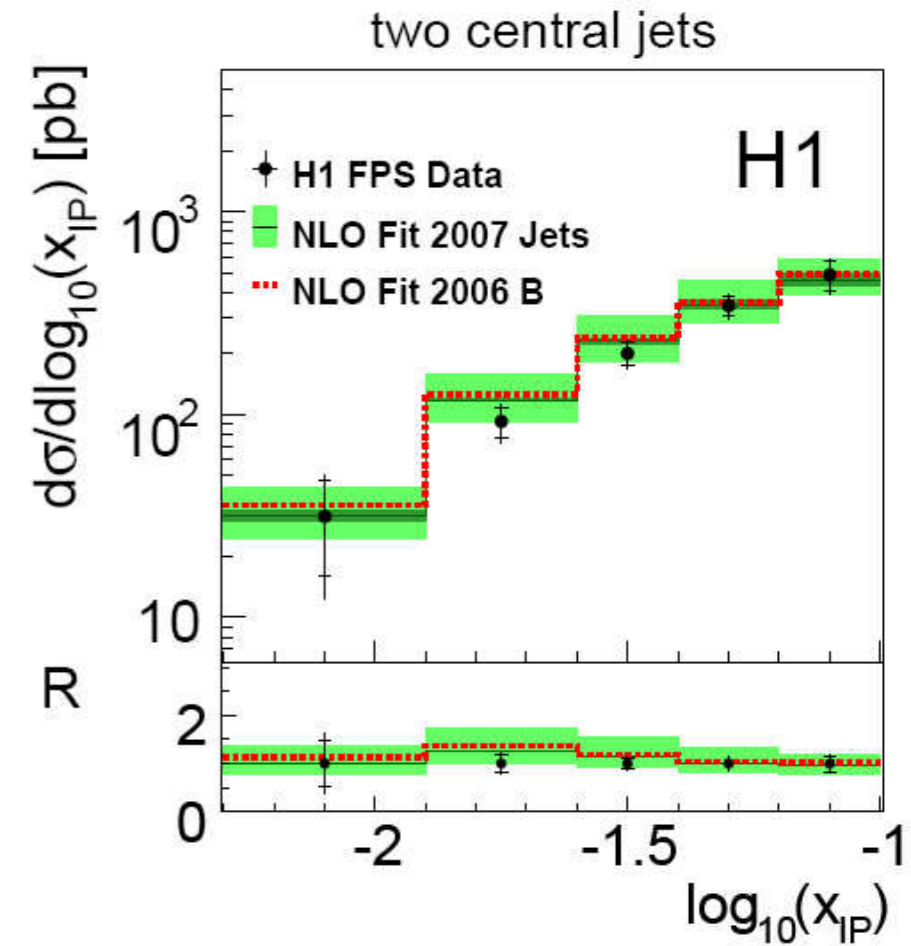
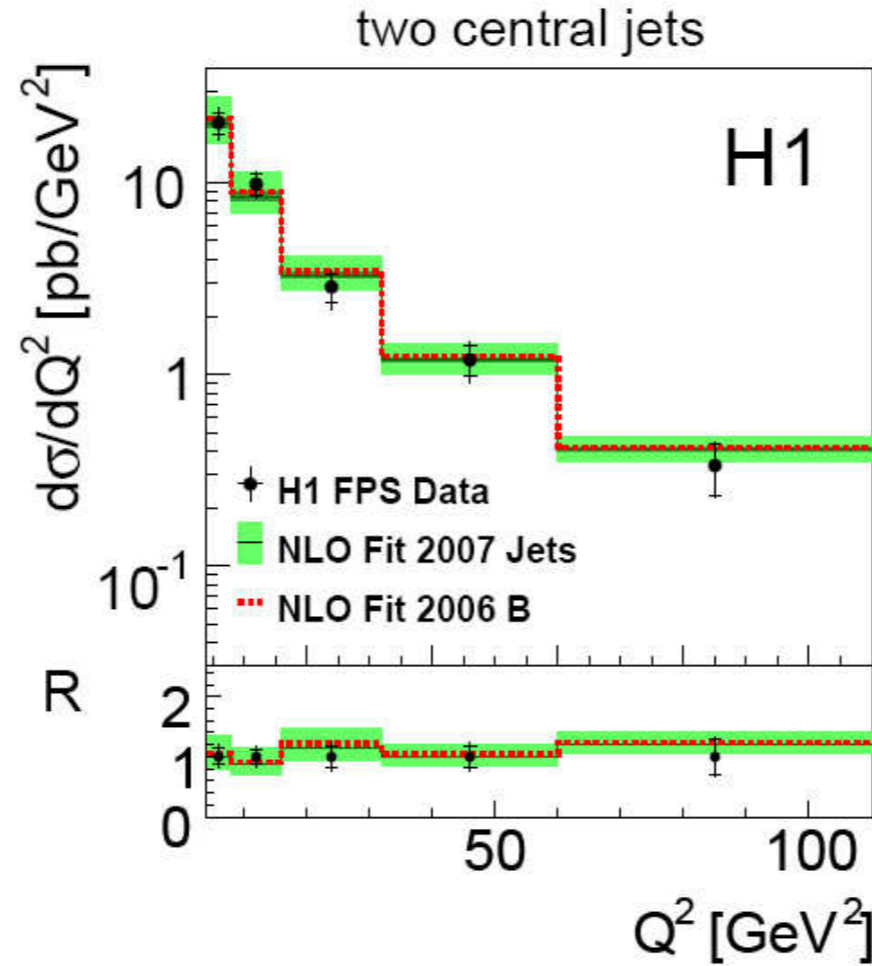
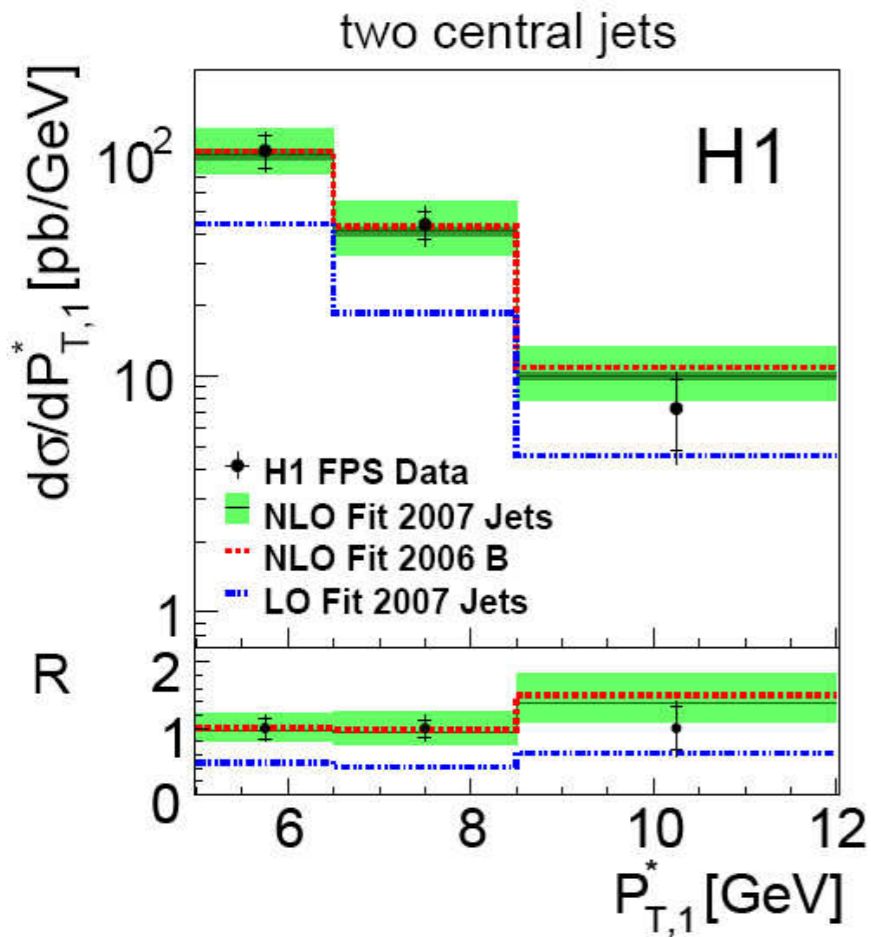
$$\xi_{IP} = \beta_{Bj} \cdot (1 + M_{jj}/Q^2)$$

# Summary

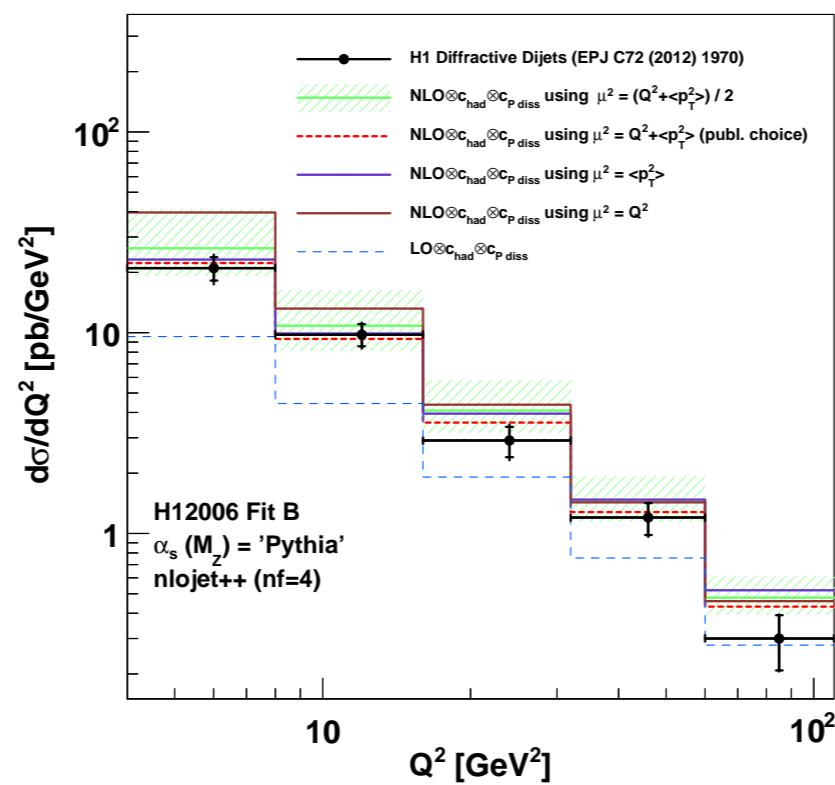
- **New method of calculating diffractive jet-cross sections was developed**
- **FastNLO code was developed and intensively tested**
- **FastNLO for jet production in diffractive DIS is ready to use**
- **Use it for your analysis or for fits of diffractive PDFs**



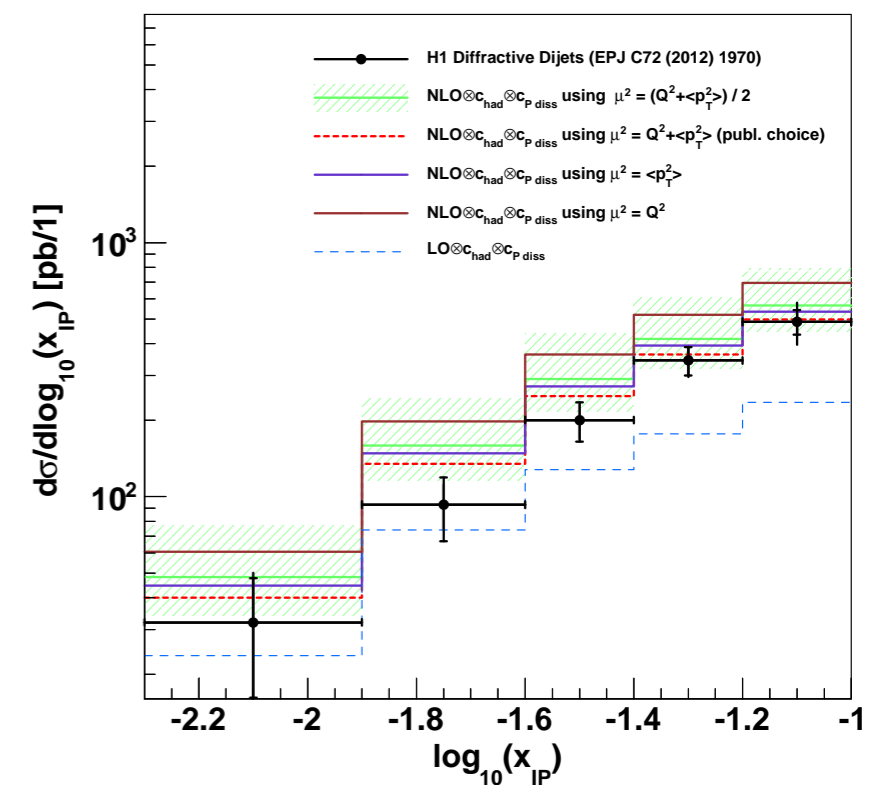
# Diffractive dijets



a)



c)





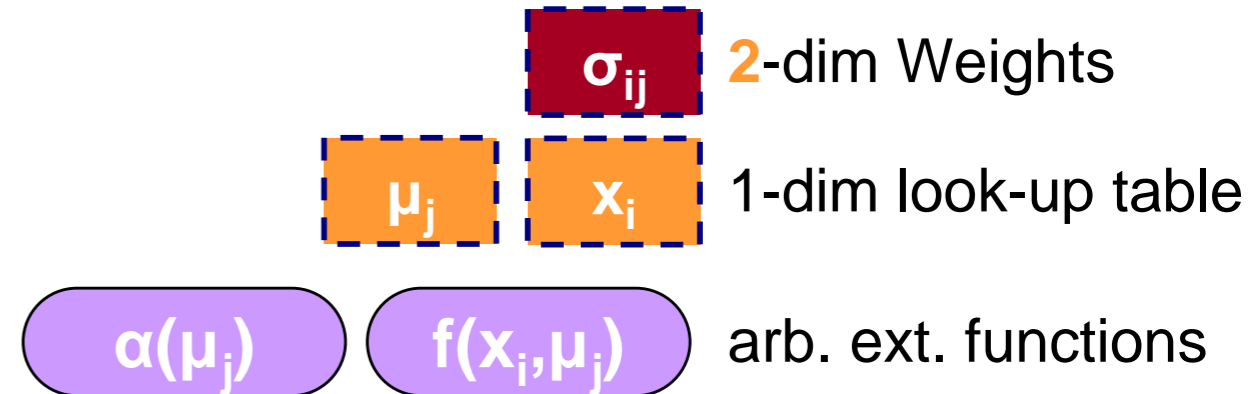


# Generalized fastNLO concept in v2.0

## We know

$$\sigma \xrightarrow{\text{fastNLO}} \sum_j^\mu \sum_i^x \tilde{\sigma}_{ij}(\mu_j) f(x_i, \mu_j) \alpha_s(\mu_j)$$

We can use variables from look-up tables for 'any' further calculation (like  $\alpha_s(\mu)$ )



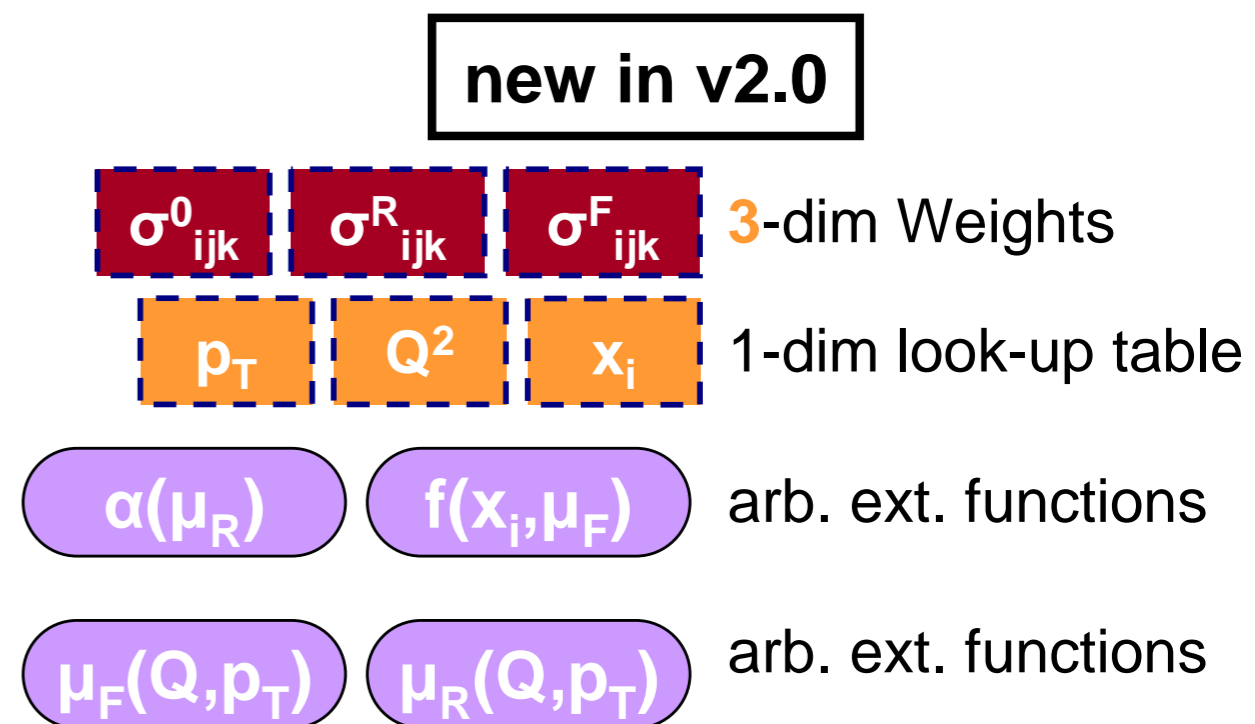
## Scale independent weights

$$\omega(\mu_R, \mu_F) = \omega_0 + \log\left(\frac{\mu_R}{Q}\right) \omega_R + \log\left(\frac{\mu_F}{Q}\right) \omega_F$$

- ' $\log(\mu/Q)$ ' can be done at evaluation time
- $\mu$ 's are 'freely' choosable functions
- $\mu \rightarrow \mu(Q, p_T)$

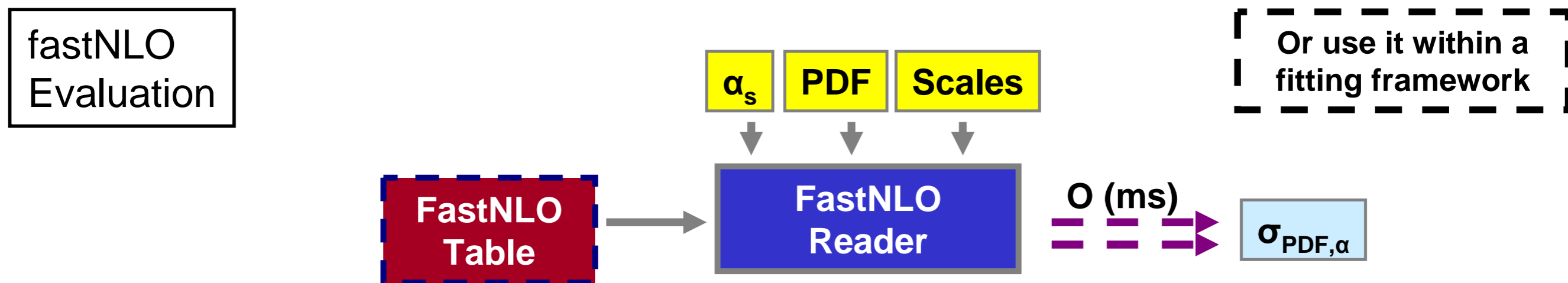
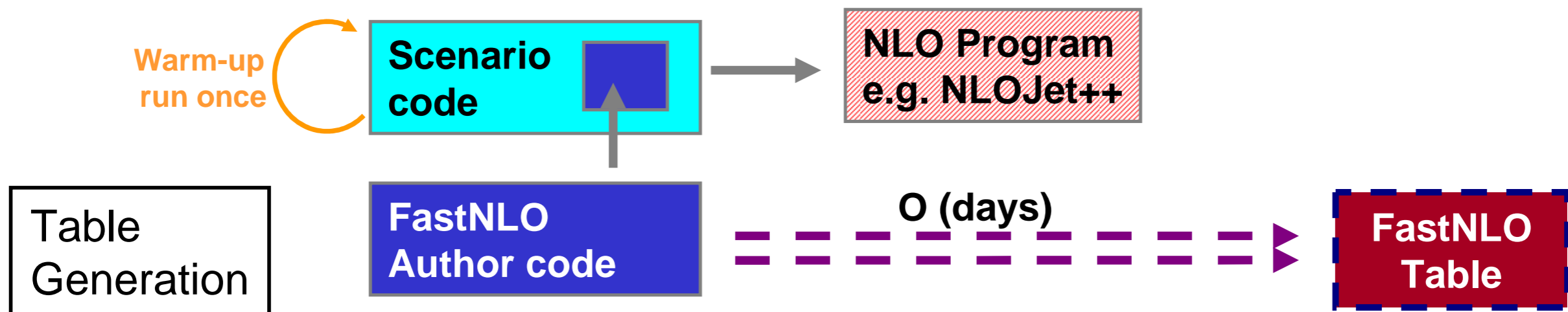
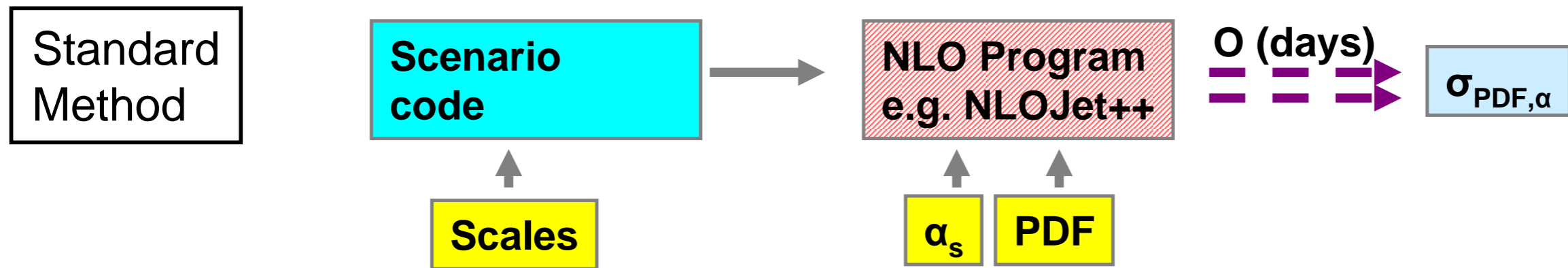
## We store scale independent contribution

Three tables holding the weights  
Further scale-variables  $\rightarrow \sigma_{ijk}$  need more dimensions



- 1) We can choose  $\mu_R$  independently from  $\mu_F$
- 2) We can choose the functional form of  $\mu_{R/F}$  as functions of **look-up-variables**

# The fastNLO concept



# The fastNLO concept

## Introduce n discrete x-nodes $x_i$ 's

- with  $x_n < \dots < x_i < \dots < x_0 = 1$
- $x_n$  is lowest x-value in each bin
- needs reasonable choice of discretization e.g.

$$f(x) = -\sqrt{\log_{10}(1/x)}$$

## Around each $x_i$ define n (cubic) Eigenfunction $E_i(x)$

$$E_i(x_j) = \begin{cases} 1 & (i = j) \\ 0 & (i \neq j) \end{cases}$$

$$\sum_i E_i(x) = 1 \quad \text{for all } x$$

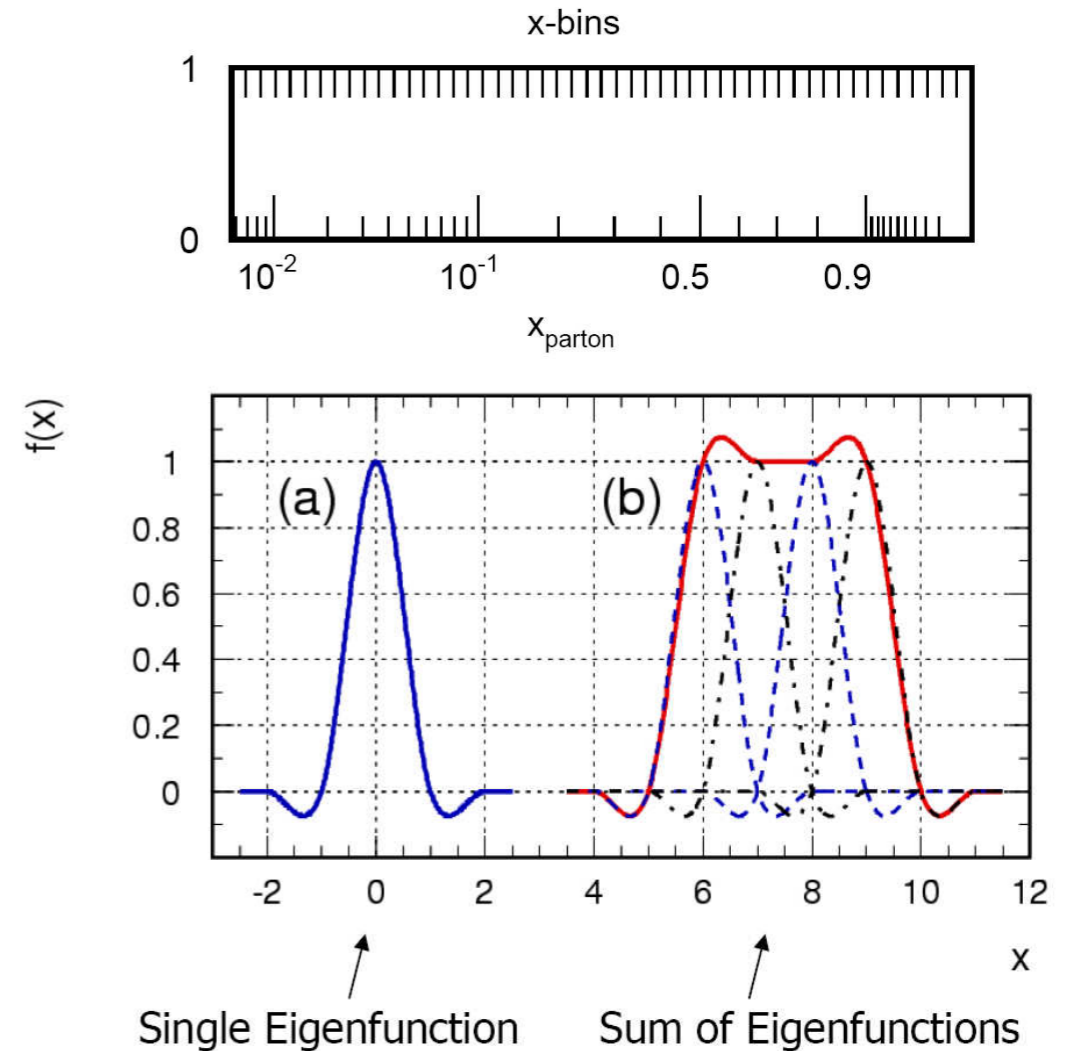
## Hadron-hadron collision need two dimensions

2D-Eigenfunctions

$$E^{(i,j)}(x_1, x_2) = E^{(i)}(x_1)E^{(j)}(x_2)$$

13x13 partonic subprocesses reduce to 7

$$\sum_{a,b}^{13 \times 13} f_{1,a}(x_1, \mu_f) f_{2,b}(x_2, \mu_f) \rightarrow \sum_k^7 H_k(x_1, x_2, \mu_f)$$



$gg \rightarrow \text{jets}$		$\bar{q}g \rightarrow \text{jets}$	$\propto$	$H_1(x_1, x_2)$
$qg \rightarrow \text{jets}$	plus	$g\bar{q} \rightarrow \text{jets}$	$\propto$	$H_2(x_1, x_2)$
$gq \rightarrow \text{jets}$	plus	$gq \rightarrow \text{jets}$	$\propto$	$H_3(x_1, x_2)$
$q_i q_j \rightarrow \text{jets}$	plus	$\bar{q}_i \bar{q}_j \rightarrow \text{jets}$	$\propto$	$H_4(x_1, x_2)$
$q_i q_i \rightarrow \text{jets}$	plus	$\bar{q}_i \bar{q}_i \rightarrow \text{jets}$	$\propto$	$H_5(x_1, x_2)$
$q_i \bar{q}_i \rightarrow \text{jets}$	plus	$\bar{q}_i q_i \rightarrow \text{jets}$	$\propto$	$H_6(x_1, x_2)$
$q_i \bar{q}_j \rightarrow \text{jets}$	plus	$\bar{q}_i q_j \rightarrow \text{jets}$	$\propto$	$H_7(x_1, x_2)$

# The fastNLO concept

## Flatten PDFs by reweighting with simple function $w(x)$

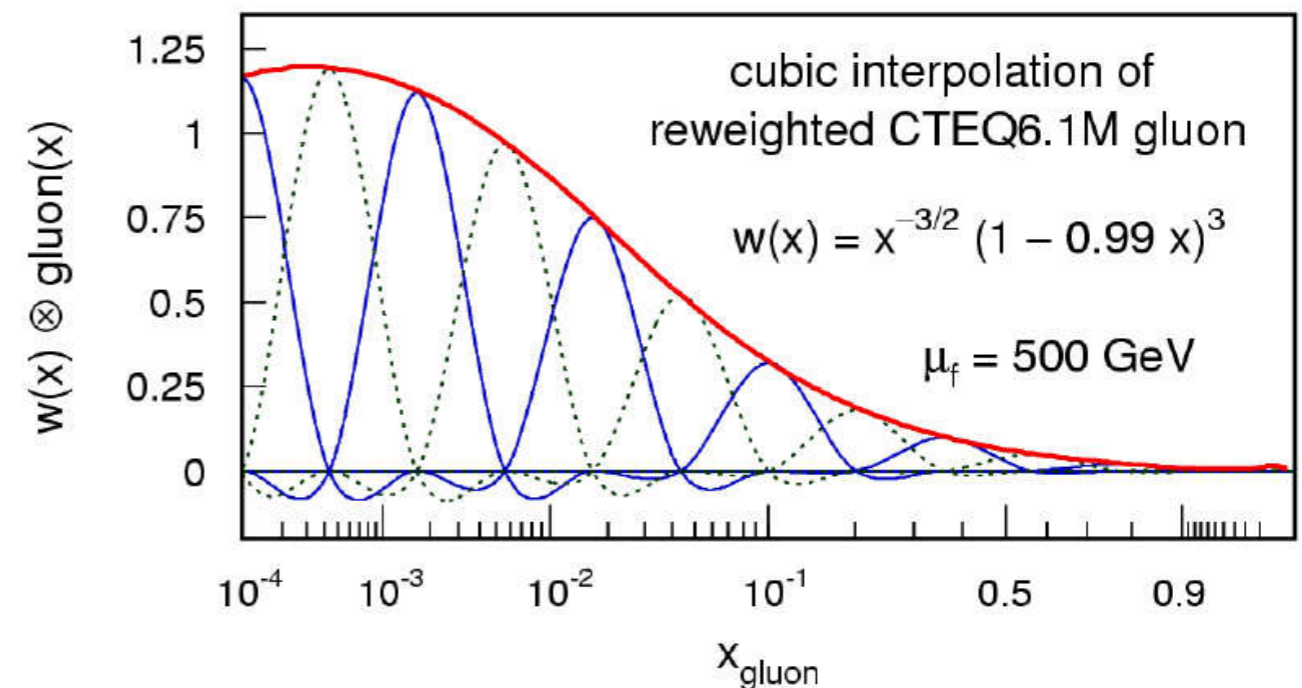
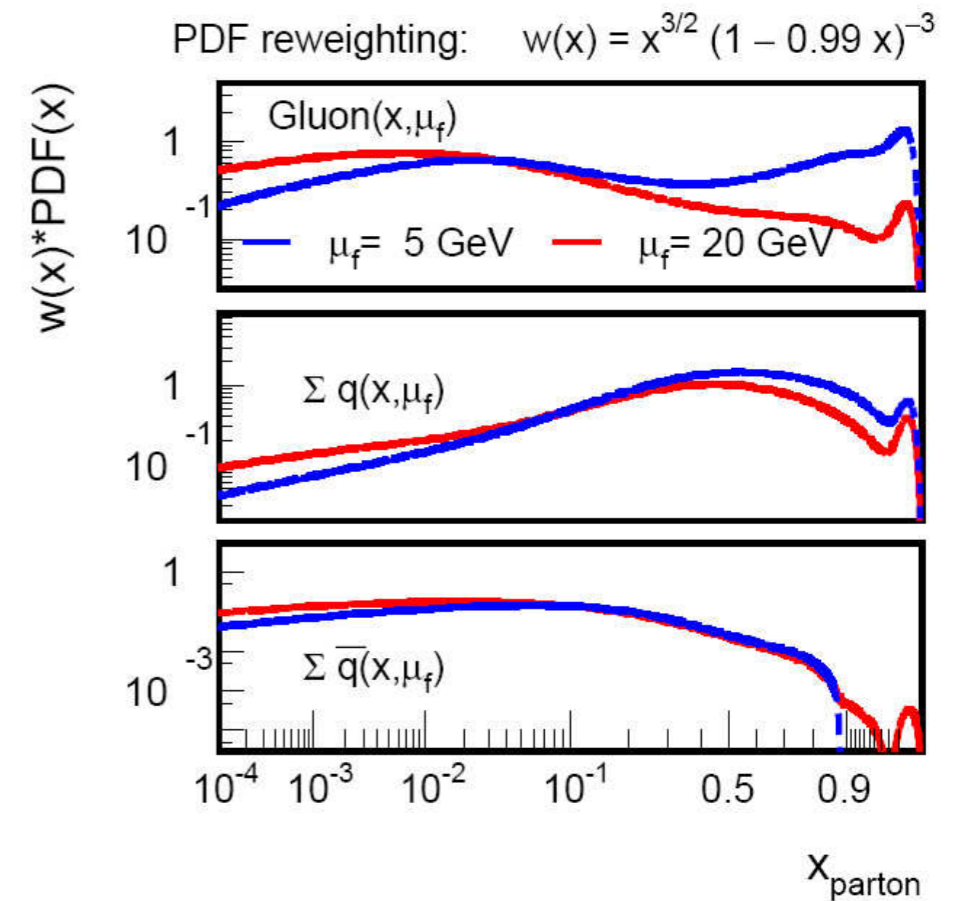
We choose

$$w(x) = x^{-3/2} (1 - 0.99x)^3$$

- Improve high- $x$  gluon
- PDF curvatures are reduced for all scales
- independent of  $\mu_f$
- $w(x)^{-1}$  is absorbed in  $E_i$

## Single PDF is replaced by a linear combination of eigenfunctions

$$f_a(x) \cong \sum_i f_a(x_i) \cdot E^{(i)}(x)$$



# The fastNLO concept

With these definitions the cross section reads

$$\sigma_{hh} = \int dx_1 \int dx_2 \sum_n \alpha_s^n(\mu_r) \sum_{k=1}^7 c_{k,n}(x_1, x_2, \mu_r, \mu_f) H_k(x_1, x_2, \mu_f)$$

Now express PDF linear combinations  $H_k$  by the 2D-eigenfunction

$$\sigma_{hh} = \int dx_1 \int dx_2 \sum_n \alpha_s^n(\mu_r) \sum_{k=1}^7 c_{k,n}(x_1, x_2, \mu_r, \mu_f) \left( \sum_{i,j} H_k(x^{(1)}, x^{(2)}) \cdot E^{(i,j)}(x_1, x_2) \right)$$

Rewrite the cross section

$$\sigma_{hh} = \sum_n \alpha_s^n(\mu_r) \sum_{k=1}^7 \sum_{i,j} H_k(x_1^{(i)}, x_2^{(j)}) \cdot \underbrace{\int dx_1 \int dx_2 c_{k,n}(\mu_r, \mu_f) \cdot E^{(i,j)}(x_1, x_2)}_{\text{Independent of PDFs and } \alpha_s}$$

**Important:** Integral is independent of PDFs!

# Last steps

## Scale dependence

- Perturbative coefficients are scale dependent
- PDFs and  $\alpha_s$  need to be evaluated at certain scale values

## Introduce interpolation procedure also for scales

- Assume  $\mu_r = \mu_f$
- Introduce  $m$  scale nodes with distances

$$f(\mu) = \log(\log(4 \cdot \mu))$$

Coefficient table gets one additional dimension for  $\mu$

## Final fastNLO cross sections

- Define  $\sigma$ -table and store it as fastNLO table

$$\tilde{\sigma}_{k,n}^{(i,j)(m)} = \sigma_{k,n}(\mu) \otimes E^{(i,j)}(x_1, x_2) \otimes E^{(m)}(\mu)$$

- Contains all information on the observable

Final cross section formula

$$\sigma_{hh}^{Bin} = \sum_{i,j,k,n,m} \alpha_s^n(\mu^{(m)}) \cdot H_k(x_1^{(i)}, x_2^{(j)}, \mu^{(m)}) \cdot \tilde{\sigma}_{k,n}^{(i,j)(m)}$$