FASTKERNEL AND APFELGRID: ACCELERATING PDF FITS

Valerio Bertone, Stefano Carrazza, NH

Nathan Hartland

University of Oxford

XFitter developer's meeting 26 April 2016

PDF4BSM

Parton Distributions in the Higgs Boson Era

erc

European Research Council

FASTKERNEL (FK) TABLES

The FastKernel (FK) procedure simplifies the convolution step in PDF fits by pre-computing the PDF evolution and combining it with interpolated weight grids

APPLgrid/FastNLO

$$\sigma_{pp\to X} = \sum_{p}^{N_\alpha} \sum_{s}^{N_{\rm sub}} \sum_{\alpha,\beta,\tau} \alpha_s^{p+p_{\rm LO}}(Q_\tau^2) W_{\alpha\beta,\tau}^{(p)(s)} F_{\alpha\beta,\tau}^{(s)}$$

$$f_i(x_\alpha, Q_\tau^2) = \sum_{k} \sum_{\beta} A_{\alpha\beta,ik}^{\tau} f_k(x_\beta, Q_0^2)$$

PDF evolution with APFEL

Absorb evolution into precomputed coeff.

$$\sigma_{pp\to X} = \sum_{k,l} \sum_{\delta,\gamma} \widetilde{W}_{kl,\delta\gamma} f_k(x_\delta, Q_0^2) f_l(x_\gamma, Q_0^2)$$

FK Table

FASTKERNEL (FK) TABLES

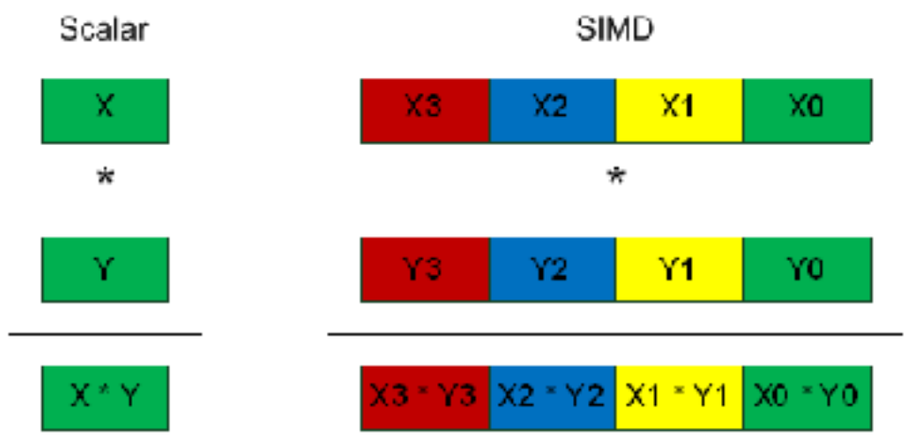
$$\sigma_{pp\to X} = \sum_{k,l} \sum_{\delta,\gamma} \widetilde{W}_{kl,\delta\gamma}\, f_k(x_\delta, Q_0^2) f_l(x_\gamma, Q_0^2)$$

Speed improvements over typical APPLgrid/FastNLO convolution

- *PDF Evolution comes 'for free'*
- *Precompute sums over hard scale*
- *Precompute sums over perturbative order*
- *PDF basis reduced to active flavours at initial scale*

FK product is simple in structure:

- *Simple to express as a scalar product*
- *Easy to make efficient use of SIMD/OpenMP*



(software.intel.com)

LIMITATIONS OF FASTKERNEL TABLES

FK tables are **not** a replacement for APPLgrids

> *By themselves, FK tables cannot perform any parameter variation other than that of PDFs.*

FK tables precompute essentially all theory parameters

> *Variations of e.g* $\alpha_S(M_Z)$ *can be performed by re-computing the table using the information stored in APPLgrids.*

Classic Space-Time tradeoff: FK tables are faster but require more space

INTERPOLATION ACCURACY

The FastKernel procedure can help decouple grid precision from fitting cost

$$\sigma^{(\mathrm{APPL})}_{pp\to X} = \sum_{\alpha,\beta}^{N_x^{(\mathrm{APPL})}} (\ldots) \longrightarrow \sigma^{(\mathrm{FK})}_{pp\to X} = \sum_{\alpha,\beta}^{N_x^{(\mathrm{FK})}} (\ldots)$$

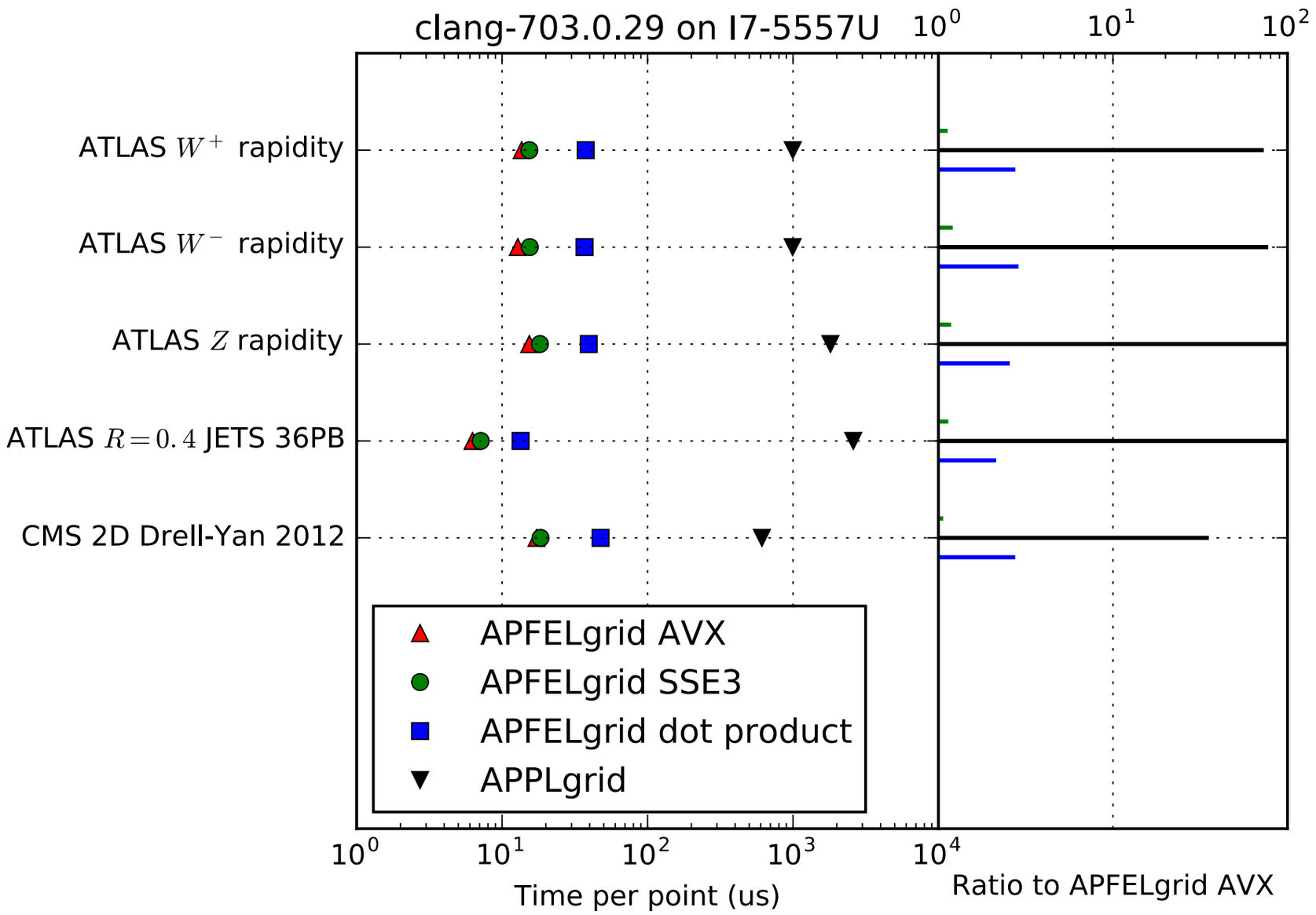
Sum over APPLgrid x-grid **Sum over PDF evolution x-grid**

While $N_x^{(\mathrm{APPL})}$ is fixed by the APPLgrid, $N_x^{(\mathrm{FK})}$ is set according by the user

From a single (high accuracy) APPLgrid, the user can generate FK tables with varying interpolation accuracy as per their requirements/preferences.

PRACTICAL EXAMPLE

Comparison of APPLgrid speed vs FK tables as used in NNPDF fits



Dataset	Speedup
ATLAS W+	73x
ATLAS W-	78x
ATLAS Z	117x
ATLAS Jets	414x
CMS 2D-DY	35x

APFELGRID

- Public tool to perform the combination of APPLgrid weight grids and APFEL PDF evolution into FastKernel tables

Release version almost complete - ETA ~weeks.

- APFELgrid plugin

```
NNPDF::FKTable<double>* FK = APFELgrid::computeFK(Q0, setname, grid, gridpath);
```

Attaches to APFEL and provides routines for the generation of FK tables to APFEL

- FastKernel driver

```
FK.Convolute(pdf_pointer, iMember, results);
```

Supplied as a single C++ header, handles FK table I/O and convolution

Suggestions/Comments welcome!