Recent developments with the APPLgrid project

Mark Sutton
University of Sussex

On behalf of the APPLgrid developers

2nd September 2015
What is the APPLgrid project?

• APPLgrid is a fully open source package to build a library of C++ utility classes for performing fast (N)NLO convolutions with PDFs (with an optional FORTRAN interface)

• Philosophy: from the beginning, make available as much functionality required for utility in QCD fits as possible - implement as many processes, and features required for actual PDF fits

• Uses a customised grid storage for reduced memory footprint

• Fast cross section production with pre-existing grids
  • Arbitrary renormalisation and factorisation scale variation
  • Arbitrary beam-energy rescaling
  • Different input PDFs for each incoming hadron
  • Any external QCD evolution or PDF code can be used

• Since the beginning, could be used by the user to generate custom grids for different cross sections and processes

• Rich environment of interfaces with (N)NLO calculations currently available ...
  • NLOjet++ for jet production
  • All processes in MCFM: Electroweak boson production, heavy flavour production, boson + jet production (plus generic interface for all remaining processes)
  • All fixed order NLO processes in aMC@NLO, using the aMCfast interface, arXiv:1406.7693
  • Sherpa for fixed NLO processes using MCgrid, arxiv:1312.4460 - in addition an independent native APPLgrid - Sherpa interface is available

• Currently working on an interface to DYNNLO for EW boson production at NNLO

• In the past we have also toyed with interfaces to JETRAD (NLO) and Vrap (NNLO)
applgrid.hepforge.org
**Grid Downloads**

Here you can download grids for APPLgrid version 1.4.70 for fast cross section evaluation. These grids are fully differential and should require no additional scaling. Each should include the non-perturbative (and any additional) bin-by-bin corrections or K-factors, the application of which is discussed in the relevant papers - see the Documentation page for more information.

Please note that these grids should be based on the grids that were used for the relevant papers, and were not necessarily created by the the appgrid authors.

**LHC: pp @ sqrt(s) = 2.76TeV**

- ATLAS inclusive jets (2011 data 0.2pb⁻¹)
  - arXiv:1304.4739v1 Inclusive jets anti-kT: Tables 4-10 (R=0.4) and 11-17 (R=0.6)

- hepdata
  - grid tarball contains
    - atlas-incljets-arxiv-1304.4739v1/atlas-incljets-eta{1-7}.root (R=0.4)
    - atlas-incljets-arxiv-1304.4739v1/atlas-incljets-eta{1-7}.root (R=0.6)

**LHC: pp @ sqrt(s) = 7TeV**

- ATLAS inclusive jets (2010 data 17nb⁻¹)
  - arXiv:1009.5908v2 Inclusive jets anti-kT: Tables 1-3 (R=0.4) and 4-6 (R=0.6)

- hepdata
  - grid tarball contains
    - atlas-incljets-arxiv-1009.5908v2/atlas-incljets-eta{1-5}.root (R=0.4)
    - atlas-incljets-arxiv-1009.5908v2/atlas-incljets-eta{1-5}.root (R=0.6)

- ATLAS inclusive jets (2010 data 37pb⁻¹)
  - arXiv:1112.6297 Inclusive jets anti-kT: Tables 5-11 (R=0.4) and 12-18 (R=0.6)

- hepdata
  - grid tarball contains
    - atlas-incljets-arxiv-1112.6297v0/atlas-incljets-eta{1-7}.root (R=0.4)
    - atlas-incljets-arxiv-1112.6297v0/atlas-incljets-eta{1-7}.root (R=0.6)

- ATLAS inclusive dijets (2010 data 37pb⁻¹)
  - arXiv:1112.6297 Inclusive dijets anti-kT: Tables 19-27 (R=0.4) and 28-36 (R=0.6)

- hepdata
  - grid tarball contains
    - atlas-incldijets-arxiv-1112.6297v0/atlas-incldijets-eta{1-9}.root (R=0.4)
    - atlas-incldijets-arxiv-1112.6297v0/atlas-incldijets-eta{1-9}.root (R=0.6)

**Code Download**

Various code for download is available:

- You can download the latest version (1.4.70) of the standard APPLgrid convolution code [here](#).
- If you wish to use arbitrary factorisation scale variation, you will need to have hoppet installed first, which you can download from [here](#).
- There are some simple examples which you can download from [here](#). These examples requires that you have LHAPDF installed which you can find [here](#).
- In addition, APPLgrid uses root files for storage, although not internally, so you should also install root which you can find [here](#).

**Calculation Code**

If you wish to generate your own grids, we currently have NLOjet++ available for jet production at NLO and MCFM for most other processes.

- For NLOjet++ we have our own custom version 4.0.1 which can be downloaded from [here](#).
  This requires the nlojet pdf module that you can download from [here](#).
  The user module can be downloaded from [here](#).

- For MCFM, you should download the standard version of MCFM (6.7 or later).
  To link with appgrid, before installing MCFM, you should download and install the mcfm-bridge code, which you install with the usual ./configure ; make ; make install etc.

After running this you will need to patch the standard installation using the patch file which you should unar in the MCFM base directory and then remove the file

```bash
src/User/gridwrap.f
```

then just build using make. To link with appgrid, before running make, you should set the LDFLAGS environment variable to the output from

```bash
mcfm-bridge --ldflags
```

NB: you should not use the standard install script that comes with MCFM.

Other interfaces are being developed and will be released when available.

**Downloads Archive**

The old downloads directory can be found [here](#) (Warning, not for the faint of heart)
### Open Source

![The APPLgrid Project](image)

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Rev</th>
<th>Age</th>
<th>Author</th>
<th>Last Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>amcatnlo-2.0.0</td>
<td>443</td>
<td>20 months</td>
<td>rojo</td>
<td>trivial edit</td>
<td></td>
</tr>
<tr>
<td>amcatnlo-bridge</td>
<td>1234</td>
<td>7 months</td>
<td>sutt</td>
<td>fix autotools</td>
<td></td>
</tr>
<tr>
<td>amfast</td>
<td>1252</td>
<td>7 months</td>
<td>bertone</td>
<td>amcfast-shower executable added</td>
<td></td>
</tr>
<tr>
<td>applgrid</td>
<td>1294</td>
<td>9 days</td>
<td>tcatli</td>
<td>corrected spelling mistake in comment</td>
<td></td>
</tr>
<tr>
<td>dynnlo</td>
<td>387</td>
<td>22 months</td>
<td>sutt</td>
<td>create directory structure</td>
<td></td>
</tr>
<tr>
<td>example</td>
<td>1282</td>
<td>5 months</td>
<td>sutt</td>
<td>fixes for pdfschool</td>
<td></td>
</tr>
<tr>
<td>jetmod</td>
<td>591</td>
<td>19 months</td>
<td>sutt</td>
<td>rename</td>
<td></td>
</tr>
<tr>
<td>lhpdf-1.0.0</td>
<td>593</td>
<td>19 months</td>
<td>sutt</td>
<td>rename</td>
<td></td>
</tr>
<tr>
<td>mcfm-6.0</td>
<td>902</td>
<td>18 months</td>
<td>sutt</td>
<td>fix cmd line arguments</td>
<td></td>
</tr>
<tr>
<td>mcfm-6.6</td>
<td>1079</td>
<td>15 months</td>
<td>sutt</td>
<td>fix broken frixione cone epsilon setting</td>
<td></td>
</tr>
<tr>
<td>mcfm-bridge</td>
<td>1296</td>
<td>12 hours</td>
<td>sutt</td>
<td>fix runmcfm</td>
<td></td>
</tr>
<tr>
<td>mcfm-patch</td>
<td>1298</td>
<td>12 hours</td>
<td>sutt</td>
<td>update makefile</td>
<td></td>
</tr>
<tr>
<td>nlojet-4.0.1</td>
<td>589</td>
<td>19 months</td>
<td>sutt</td>
<td>rename</td>
<td></td>
</tr>
<tr>
<td>obsolete</td>
<td>367</td>
<td>23 months</td>
<td>sutt</td>
<td>move tags to obsolete directory</td>
<td></td>
</tr>
<tr>
<td>sherpa</td>
<td>1268</td>
<td>5 months</td>
<td>tcatli</td>
<td>added steering for ATLAS ttbar+jets</td>
<td></td>
</tr>
<tr>
<td>utils</td>
<td>1219</td>
<td>8 months</td>
<td>sutt</td>
<td>improve tagging and preparation</td>
<td></td>
</tr>
<tr>
<td>web</td>
<td>1182</td>
<td>9 months</td>
<td>sutt</td>
<td>modify news bg colours</td>
<td></td>
</tr>
</tbody>
</table>

View changes...
## Timeline of grid technology

<table>
<thead>
<tr>
<th>Year</th>
<th>Event</th>
<th>Details</th>
</tr>
</thead>
</table>
| 2000 | First implementation of grid technique for DIS at H1                  | Markus Wobisch  
DESY-THESIS-2000-049 |
| 2001 | First basic fit in DIS using jets H1                                  | EPJ C19, 289 (2001) |
| 2005 | Early APPLgrid implementation for jets in hadron-hadron collisions   | Carli, Salam, Siegert  
- C++, fully open source  
- user code for grid generation available  
- arbitrary scale variation |
| < 2006 | fastNLO implementation for DIS and jets in hadron-hadron            | Kidonakis et al, Kluge Rabbertz, Wobisch  
- Separate Fortran routines for only precomputed grids  
- limited precomputed scale choices  
- No user grid generation |
| Nov 2009 | APPLgrid for jets in hadron-hadron - first full release          | Markus Wobisch  
DESY-THESIS-2000-049 |
| 2010 | APPLgrid extension to heavy flavour production                      | Carli, Salam, Siegert  
- custom sparse memory structure for more efficient storage  
- arbitrary beam energy scaling  
- fastNLO interface  
- First implementation of non-jet cross sections - MCFM interface for inclusive W and Z production at NLO |
| Aug 2012 | fastNLO version 2 + toolkit produced                               | Carli, Salam, Siegert  
- New C++ interface and user grid generation code for jet production made available |
| Dec 2013 | APPLgrid developments for all remaining MCFM processes              | Carli, Salam, Siegert  
- including Z, W + jets, W+charm etc |
| July 2013 | APPLgrid Native interface to Sherpa                                | Carli, Salam, Siegert  
- All fixed order NLO QCD in Sherpa |
| June 2014 | APPLgrid modifications for MCFM integrated into official MCFM 6.7   | Carli, Salam, Siegert  
- all NLO QCD in Sherpa from within Rivet |
| June 2014 | AMCfast - aMC@NLO intererface to APPLgrid                           | Carli, Salam, Siegert  
- All fixed order NLO QCD in aMC@NLO |
| Sep 2015 | fastNLO Interface to DiffTop                                       | Carli, Salam, Siegert  
- Closed source, grid generation code not available |
| Early 2015 | fastNLO integration with fixed order NLO Sherpa                    | Carli, Salam, Siegert  
- using the MCgrid Sherpa interface |

Watch this space...
MCFM interface

- Currently version 6.8 tested with W, Z production, QQ production (bbar, ttbar) W, Z + jets, W+c, prompt photon production ...
- Since W and Z production in APPLgrid has been around for a good while and is well known - won't discuss here

- Due largely to the efforts of Pavel Starovoitov and Alberto Guffanti since version MCFM 6.7, modifications required to interface with APPLgrid have been included in the standard MCFM - no longer any need for a custom version
- Interface with MCFM 7.0 being validated ...

- Have a new paradigm for the interface with MCFM
  - Non obtrusive hook functions based on null function pointers that can be left compiled into the standard MCFM code
  - New bridge package that includes a setup function which initialises the hook functions to external filling routines ...
    - APPLgrid interface included, or excluded simply by linking against the APPLgrid libraries, without requiring any recompilation ...
    - If you don't link against the APPLgrid libraries you get standard MCFM, no MCFM code needs to be compiled
- This paradigm is also used to great success in the aMCfast interface for aMC@NLO
Current MCFM status ...

- All MCFM processes, in principle now included
- Those that we have (personally) verified ...
  - EW Boson production ...
    - Inclusive $W^\pm$ production (processes 1, 6)
    - $W^\pm$ + jet production (processes 11, 15)
    - Inclusive Z production (process 31)
    - Z + jet production (processes 41-43)
    - Prompt photon production (processes 280 - 286) - including fragmentation
  - Heavy flavour ...
    - ttbar production (semi-leptonic decay) with/without spin correlations (proc. 141, 142, 144, 145 )
    - ttbar production (hadronic decay), with radiative corrections in t/W decays etc (proc. 146 - 151 )
    - Inclusive ttbar production (157)
    - Inclusive bbar production (158)
    - Inclusive ccbar production (159)
  - Combined EW+heavy flavour ...
    - $W^\pm$ + c production (processes 13-16)
MCFM top - dynamic scale patch

- Without the patch, small $\sim 2\%$ non-closure of the top cross section at high $p_T$
- With the patch closes to better than $0.1\%$
Just a selection of interesting processes ...

- $W^+$ charm production implemented and well tested
- Bare charm - for realistic comparison with experimental fiducial data some model of hadronisation is required
Prompt photon production (Andrey Sapronov)

- Prompt photon production at NLO + LO fragmentation process
- Small non-closure at the level of 0.02%
- Some validation underway: NLO+NLO fragmentation contribution available from JetPhox - calculating LO → NLO fragmentation k-factors
Generic subprocesses

- For generic calculations, can use a representative **basic** configuration containing $11 \times 11$ (non-top) subprocesses.
- This is rather **too large** for general use, so that we have a feature of the grids that can reduce this to the smallest number of unique subprocesses.
- For example, for inclusive photon production with 121 processes, these can be reduced to 6 subprocesses for LO and 33 for NLO.

|   0 6 -5 0 -3 0 -1 0 1 0 3 0 5 0 |
| 1 6 -5 5 -3 3 -1 1 -1 3 -3 5 -5 |
| 2 4 -4 0 -2 0 2 0 4 0 |
| 3 4 -4 4 -2 2 2 -2 4 -4 |
| 4 6 0 -5 0 -3 0 -1 0 1 0 3 0 5 |
| 5 4 0 -4 0 -2 0 2 0 4 |

| 0 3 -5 -5 -3 -3 -1 -1 |
| 1 6 -5 -4 -5 -2 -3 -4 -3 -2 -1 -4 -1 -2 |
| 2 6 -5 -3 -5 -1 -3 -5 -3 -1 -1 -5 -1 -3 |
| 3 3 -5 0 -3 0 -1 0 |
| 4 6 -5 1 -5 3 -3 1 -3 5 -1 3 -1 5 |
| 5 6 -5 2 -5 4 -3 2 -3 4 -1 2 -1 4 |
| 6 3 -5 5 -3 3 -1 1 |
| 7 6 -4 -5 -4 -3 -4 -1 -2 -5 -2 -3 -2 -1 |
| 8 2 -4 -4 -2 -2 |
| 9 2 -4 -2 -2 -4 |
| 10 2 -4 0 -2 0 |
| 11 6 -4 1 -4 3 -4 5 -2 1 -2 3 -2 5 |
| 12 2 -4 2 -2 4 |
| 13 2 -4 4 -2 2 |
| 14 3 0 -5 0 -3 0 -1 |
| 15 2 0 -4 0 -2 |
| 16 1 0 0 |
| 17 3 0 1 0 3 0 5 |
| 18 2 0 2 0 4 |
| 19 6 1 -5 1 -3 3 -5 3 -1 5 -3 5 -1 |
| 20 6 1 -4 1 -2 3 -4 3 -2 5 -4 5 -2 |
| 21 3 1 -1 3 -3 5 -5 |
| 22 3 1 0 3 0 5 0 |
| 23 3 1 1 3 3 5 5 |
| 24 6 1 2 1 4 3 2 3 4 5 2 5 4 |
| 25 6 1 3 1 5 3 1 3 5 5 1 5 3 |
| 26 6 2 -5 2 -3 2 -1 4 -5 4 -3 4 -1 |
| 27 2 2 -4 4 -2 |
| 28 2 2 -2 4 -4 |
| 29 2 2 0 4 0 |
| 30 6 2 1 2 3 2 5 4 1 4 3 4 5 |
| 31 2 2 2 4 4 |
| 32 2 2 4 4 2 |

- **NB:** This **may not** be the minimal number for which symmetries of the matrix elements could be used, but it is a significant reduction none-the-less.
Sherpa : Native APPLgrid - Sherpa interface

- Uses the native Sherpa output ntuple: Tancredi Carli, Cameron Embree MS
- Uses the generic process decomposition
- Can be used for any fixed order NLO sherpa process
- **Caveat:** when generating weights for individual parton-parton subprocesses, in order to get agreement with reference at low statistics, need to use the full 121 parton-parton luminosities
  - Imposing known symmetries in the matrix element to fill the grid with fewer subprocesses provides more precise grids, but at the cost of better than statistical agreement with the reference histogram at low statistics
MCgrid Sherpa interface - Del Debbio et al

- Available for all fixed order NLO processes using the generic PDF decomposition

**ME generator**
LO/NLO QCD events

**HepMC**

**Rivet**
observable projections

**MCgrid**

**APPLGRID**
differential xsec grids

- MCgrid works as a Rivet plugin using the HepMC event record

**Inclusive jets at the LHC**

**MCgrid/APPLgrid** comparison for different grid resolutions

**Inclusive jets in the ttbar event**

```
\[ \text{MCgrid/APPLgrid} \]
```

**Plot from Enrico Bothmann**

```
\[ \text{Grid / Sherpa} \]
```

**First jet p_T [GeV]**
**aMCfast** - A fast interface between MadGraph5_aMC@NLO and APPLgrid

*aMCfast* [arXiv:1406.7693] is an automated interface which bridges the automated cross section calculator MadGraph5_aMC@NLO [arXiv:1405.0301] with the fast interpolator APPLgrid [arXiv:0911.2965].

The chain MadGraph5_aMC@NLO – aMCfast – APPLgrid will allow one to include, in a straightforward manner, any present or future LHC measurement in an NLO global PDF analysis.

The basic idea behind the use of these three codes is that of computing user-defined observables relevant to arbitrary processes, and to represent them in terms of look-up grids, which can be accessed at later times, and used to obtain predictions for such observables with any PDFs. This a-posteriori computation is both accurate and very fast. Contrary to other APPLgrid application, factorisation scale variation can be performed without linking to any third-party code.

The following representative figures show the rapidity of the top quark in top-pair production and the lepton-pair invariant mass in dilepton production in association with one jet at the 14 TeV LHC. For more details, please refer to the original publication.

- Valerio Bertone, Rik Frederix, Stefano Frixione, Juan Rojo, MS
aMCFast interface MC@NLO interface

- Implementation of aMC@NLO standalone calculations uses 4 sets of weights - one at tree level and three for the NLO part

\[ d\sigma = \alpha_s(Q)^n \left[ W^B + \alpha_s(Q) \left[ W^0 + W^R \log \frac{\mu_R}{Q} + W^F \log \frac{\mu_F}{Q} \right] \right] \]

- Minor internal modifications to the grid class to make use of additional weight grids

- Implements all fixed order NLO calculations within aMC@NLO

- Using generic subprocesses automatically configured using the information provided by aMC@NLO and the new "MCFM" interface paradigm discussed earlier

- Makes extensive use of the grid combination utilities to combine grids after generation
aMCfast interface to aMC@NLO

- Agreement to better than 0.05%
Upcoming developments

- Many new developments are on their way
  - Just a taster here - significantly faster convolution implementation
  - Aspects of internal grid structure being reimplemented, new utilities
- Several new calculations being implemented, watch this space for details

Atlas 2011 inclusive jets convolution time

- applgrid 1.4.70
- applgrid 1.4.71

Latency

<table>
<thead>
<tr>
<th>Entries</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>93.58 ± 0.5778</td>
</tr>
<tr>
<td>RMS</td>
<td>2.584 ± 0.4086</td>
</tr>
</tbody>
</table>

Cross section

PT (jet) [GeV]

Reference

Fast convolution

Ratio

0.999
0.9992
0.9994
0.9996
0.9998
1.0000
1.0002
1.0004
1.0006
1.0008
1.001

200 400 600 800 1000 1200 1400 1600 1800

M Sutton - Recent developments with APPLgrid
• New plotting web utility for simpler production of comparisons of data with calculations code

• Still under development but should be available soon

• From Tancredi Carli and Joe Gibson
Several sets of cross section data and grids are available

- Compare data and calculation with any choice of PDF, with or without hadronisation, EW corrections etc
- Aim is to try to collect all available processes and grids - grids will be downloadable, with appropriate authorship attribution etc, so please donate your grids!
Outlook

• The APPLgrid project is now reasonably mature with an increasingly extensive portfolio of cross section processes available

• An increasingly large user base is developing for both the fast convolution code and grid generation

• Developments are underway for interfaces with new calculation code
  • Cross community discussions might be useful for more extensive and rapid progress

• We are always open to new suggestions or requests - particularly if people want to contribute to the project