Herwig++ @ NLO

Simon Plätzer

DESY Theory Group

with contributions from Ken Arnold, Stefan Gieseke, Jan Kotanski, Malin Sjödahl, Martin Stoll

A bit of history

A long history of NLO matrix elements in Herwig++.

One of the first generators to provide POWHEG matching.
Now a bunch of builtin processes available.

So far: handmade. But full truncated showering.

This talk: change of paradigm to automation.
Also optional shower and different matching strategies → systematics.

[Herwig++ 2.6 release, arXiv:1205.4902]

Outline

- The Matchbox NLO framework
- Dipole shower MC@NLO
- Which POWHEG?
- Adaptive sampling of Sudakov-type densities
- Some results
- Conclusions and outlook

The Matchbox NLO framework

[SP & S. Gieseke, arXiv:1109.6256 and work in progress]

A framework to automatically assemble NLO calculations.
Includes matching to showering.

Need external code to deliver tree level and one loop amplitudes.
But not more.

Behaves just as plain Herwig++.
No separate codes to run, no intermediate event files.

Matchbox in a nutshell

Generic interfaces:

- phasespace generation
- squared/correlated matrix elements
- colour subamplitudes and colour bases

Built in (behind these interfaces):

- multi channel phasespace
- simple colour sturctures, general case in progress

Then assembles full-fledged NLO calculation:
Automated dipole subtraction and integrated dipoles.

Many things behind the scenes:
(Tree) diagram generation, caching, spinor helicty helpers ...

Matchbox dipole subtraction

Use diagram information to determine dipoles.

- Look at mergings at external legs.
- Match to Born diagrams: gives assignment of tilde kinematics.

Need colour/spin correlated matrix elements:

- Either directly from external code.
- Or interface colour subamplitudes.

Simple colour structures built in.
General case from ColorFull package.

- Backbone for subleading-N improved showers [M. Sjödahl, SP – arXiv:1201.0260]

Matchbox dipole subtraction

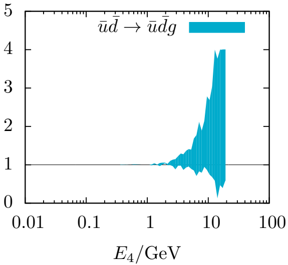
All massless dipoles and insertion operators available.

Various conventions for ϵ-expansions supported, also CDR and DR.

Massive ones in progress. [M. Stoll, diploma thesis KIT 2012]

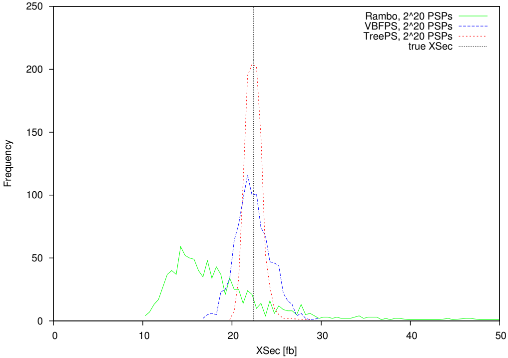
Validated for several processes.

Here $pp \to$ jets, amplitudes from nlojet++.



Matchbox phasespace performance

Use diagram information to generate phase space as well:
Multi channel, map out structures topology by topology.



Outperforms VBFNLO builtin phasespace.

NLO Matching in a Nutshell.

Fixed-order expansion of NLO+PS: double counting evident

$$
\begin{aligned}
\sigma[u] \;=\; & \int_n u(p_n)\mathrm{d}\sigma^{(n,0)}(p_n) \\
& +\alpha_s \int_n u(p_n)\left[\mathrm{d}\sigma^{(n,1)}(p_n) + \int_1 \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]_{\epsilon=0} \\
& +\alpha_s \int_{n+1}\left[-u(p_n)\mathrm{d}P(p_{n+1}|p_n) - u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right] \\
& +\alpha_s \int_{n+1}\left[u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) + u(p_{n+1})\mathrm{d}P(p_{n+1}|p_n)\right]
\end{aligned}
$$

NLO Matching in a Nutshell.

Matched calculation: subtract double counting

$$\begin{aligned}\sigma[u] \; = \; & \int_n u(p_n)\mathrm{d}\sigma^{(n,0)}(p_n) \\ & +\alpha_s \int_n u(p_n)\left[\mathrm{d}\sigma^{(n,1)}(p_n) + \int_1 \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]_{\epsilon=0} \\ & +\alpha_s \int_{n+1}\left[u(p_n)\mathrm{d}P(p_{n+1}|p_n) - u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right] \\ & +\alpha_s \int_{n+1}\left[u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) - u(p_{n+1})\mathrm{d}P(p_{n+1}|p_n)\right]\end{aligned}$$

NLO Matching in a Nutshell.

MC@NLO type: tedious (basically redo NLO calculation). [Webber, Frixione, '02–]

$$\begin{aligned}\sigma[u] \;=\; & \int_n u(p_n)\mathrm{d}\sigma^{(n,0)}(p_n) \\ & +\alpha_s \int_n u(p_n)\left[\mathrm{d}\sigma^{(n,1)}(p_n) + \int_1 \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]_{\epsilon=0} \\ & +\alpha_s \int_{n+1}\left[u(p_n)\mathrm{d}P(p_{n+1}|p_n) - u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right] \\ & +\alpha_s \int_{n+1}\left[u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) - u(p_{n+1})\mathrm{d}P(p_{n+1}|p_n)\right]\end{aligned}$$

NLO Matching in a Nutshell.

MC@NLO made easy: $\mathrm{d}P(p_{n+1}|p_n) = \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})$

$$\begin{aligned}
\sigma[u] \;\; = \;\; & \int_n u(p_n)\mathrm{d}\sigma^{(n,0)}(p_n) \\
& +\alpha_s \int_n u(p_n) \left[\mathrm{d}\sigma^{(n,1)}(p_n) + \int_1 \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]_{\epsilon=0} \\
& +\alpha_s \int_{n+1} \left[u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1}) - u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right] \\
& +\alpha_s \int_{n+1} \left[u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) - u(p_{n+1})\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]
\end{aligned}$$

NLO Matching in a Nutshell.

POWHEG type: $\mathrm{d}P(p_{n+1}|p_n) = \mathrm{d}\sigma^{(n+1,0)}(p_{n+1})$ [Nason, '04–]

$$\begin{aligned}
\sigma[u] \;\; = \;\; & \int_n u(p_n)\mathrm{d}\sigma^{(n,0)}(p_n) \\
& +\alpha_s \int_n u(p_n) \left[\mathrm{d}\sigma^{(n,1)}(p_n) + \int_1 \mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right]_{\epsilon=0} \\
& +\alpha_s \int_{n+1} \left[u(p_n)\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) - u(p_n)\mathrm{d}\sigma_A^{(n+1,0)}(p_{n+1})\right] \\
& +\alpha_s \int_{n+1} \left[u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1}) - u(p_{n+1})\mathrm{d}\sigma^{(n+1,0)}(p_{n+1})\right]
\end{aligned}$$

Dipole shower MC@NLO

Matching greatly simplified, if shower uses subtraction terms.

Use Herwig++ dipole shower. [based on SP, S. Gieseke – arXiv:0909.5593]

Matching will be exact only up to colour suppressed terms.
Cured by subleading-N improved shower, running in the same framework.

[cf independent approach by Sherpa]

Which POWHEG?

Matchbox can as well turn NLO automatically into POWHEG matching. There's actually a family of POWHEG matchings:

- Paritioning of the real emission

  $$|\mathcal{M}_R|^2 = \sum_i \frac{w_i}{\sum_j w_j} |\mathcal{M}_R|^2$$

  Natural to use dipoles or splitting functions.
  Dipoles without cuts, not the subtraction terms (0/0 ...)

- Splitting of real emission into singular and finite terms

  $$|\mathcal{M}_R|^2 = \frac{\mathrm{d}\sigma_B}{\mathrm{d}\sigma_B + \mathrm{d}\sigma_H} |\mathcal{M}_R|^2 + \frac{\mathrm{d}\sigma_H}{\mathrm{d}\sigma_B + \mathrm{d}\sigma_H} |\mathcal{M}_R|^2$$

  $\mathrm{d}\sigma_H \sim p_\perp^\alpha$; also allows to get rid of instabilities in PDF ratios.

Adaptive sampling of Sudakov-type densities

We frequently need to draw q and a number of variables z from

$$\frac{\mathrm{d}S_P(\mu, q|Q; z; \xi)}{\mathrm{d}q\ \mathrm{d}^n z} = \Delta_P(\mu|Q;\xi)\delta(q-\mu) + \theta(Q-q)\theta(q-\mu)P(q;z;\xi)\Delta_P(q|Q;\xi)$$

with

$$\Delta_P(q|Q;\xi) = \exp\left(-\int_q^Q \int P(k;z;\xi)\mathrm{d}^n z\ \mathrm{d}k\right)\ .$$

For some (floating) hard scale Q.
And a potentially big number of parameters ξ.

Adaptive sampling of Sudakov-type densities

Dealt with by the Sudakov veto algorithm.
Can also be extended to non-positive kernels P. [SP & M. Sjödahl, EPJ Plus 127 (2012) 26]

But requires an overestimate R very close to P for all (q, z, ξ).
Up to know figured out by hand.

This may neither be ‘portable’ nor most efficient.

→ Use adaptive methods.

[Very much motivated by ACDC (L. Lönnblad) and Foam (S. Jadach et al.)]

The ExSample library

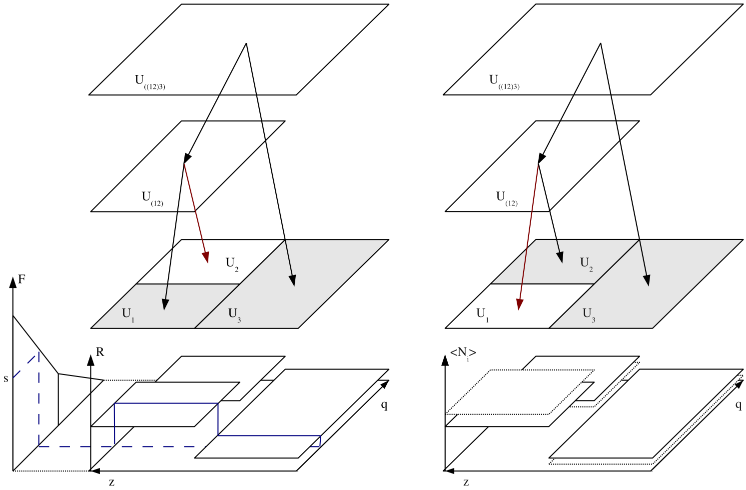
[SP, EPJ C72 (2012) 1929]

Get a glimpse on P from the first few points, then refine.
Organize in a binary tree of sub-hypercubes in (q, z, ξ) space.

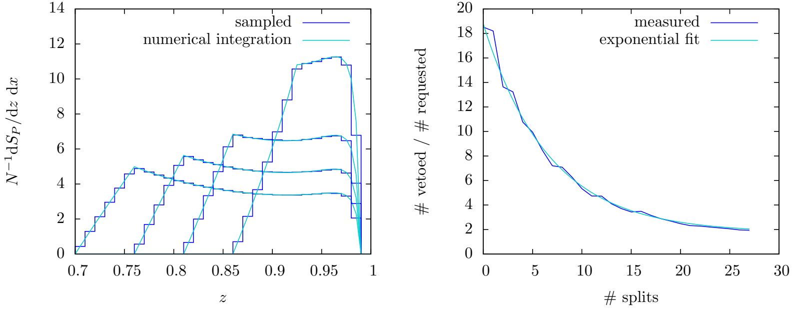
Refine by splitting hypercubes → fractal structure.
Determine new splits to optimize overall performance.

Note that the 'first glimpse' will not yield the true maximum.
Compensation needed for errornous overestimates.

The ExSample library



The ExSample library



Proof of concept: Simple processes.

e^+e^-, DIS, Drell-Yan [SP, S. Gieseke – arXiv:1109.6256]

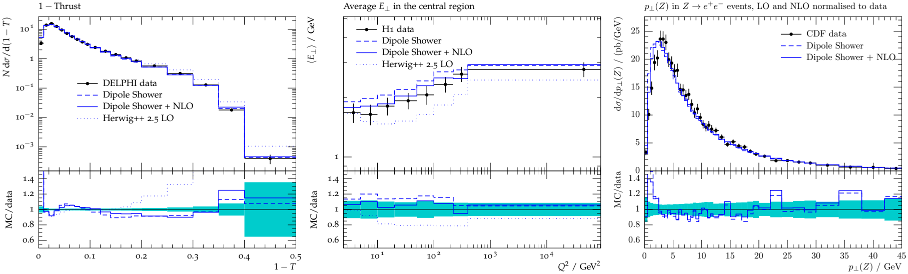
Separate LO and NLO tunes → understand systematics.

- Reasonable description of data.
- Apart from normalization marginal improvements at NLO.
- α_s determined more precisely at NLO.
- NLO pushes IR cutoff to smaller values
  → better modelling of perturbative dynamics

Proof of concept: Simple processes.

e^+e^-, DIS, Drell-Yan

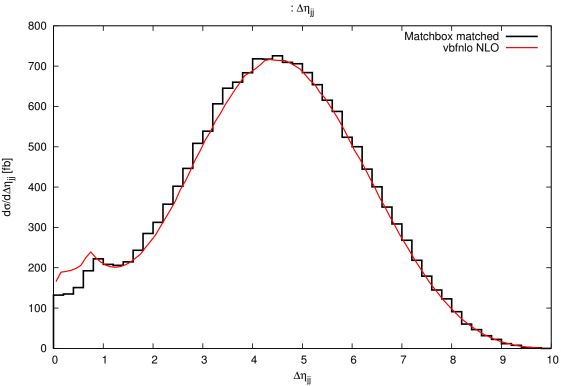
[SP, S. Gieseke – arXiv:1109.6256]



Matchbox/VBFNLO

[K. Arnold]

$H + 2$ *jets* with Matchbox MC@NLO



VBF $Z + 2$ *jets*, $W + 2$ *jets* in progress.

Conclusions and outlook

- NLO well established within Herwig++
- Various specialized POWHEG implementations
- Matchbox provides full-fledged NLO framework
  → proof of concept with simple processes
  → target now at more complicated processes
  → switch between MC@NLO and POWHEG for systematics
- Further related developments
  → dipole shower as alternative shower → systematics
  → subleading N improved showering vs. MC@NLO
  → merging NLOs ...

Herwig++ @ NLO

Simon Plätzer

DESY Theory Group