GENEVA: A New Framework For Event Generation

Frank Tackmann

University of California, Berkeley
Lawrence Berkeley National Laboratory

Theory Seminar
Fermilab, November 8, 2007

Work with Christian Bauer and Jesse Thaler

arXiv:0712.????

Outline

Outline

Motivation

At LHC and ILC

- Growing need and availability of precise theoretical predictions
  - More and more NLO or even NNLO calculations
  - Subleading log resummation, power corrections (e.g. in SCET)
  - Multi-leg matrix elements
- Shift towards less postdiction/tuning and more prediction
  - Would like to combine most accurate available descriptions for different parts of phase space
- Make theory results available to experiments
  - More easily and more quickly
  - Need a generic way to use *inclusive parton-level* calculations to get *exclusive hadron-level* events

Combining Matrix Elements and Parton Showers

Basic Strategies

- Subtractive: Expand shower and subtract it from matrix elements, e.g.
  - MC@NLO [Frixione, Webber] and similar NLO [Krämer, Mrenna, Nagy, Soper]
  - VINCIA [Giele, Kosower, Skands]
- Multiplicative: Apply correction weights, e.g.
  - Old PYTHIA way to correct first emission
  - CKKW to put in Sudakovs
- Others: MLM, POWHEG [Nason]

Partonic Calculations vs. Algorithmic Tools

- So far, special algorithms for specific problems
- Separate partonic calculations from algorithms
  - Calculations should not have to know specifics of implementations

GENEVA: Generate Events Analytically

Goals of GENEVA

- Provide *generic* and *straightforward* framework to map inclusive parton-level calculations into exclusive hadron-level events
  - Let user think in terms of partonic cross sections
  - Avoid having to think about algorithms on case by case basis
- Use most accurate available prediction for each part of phase space

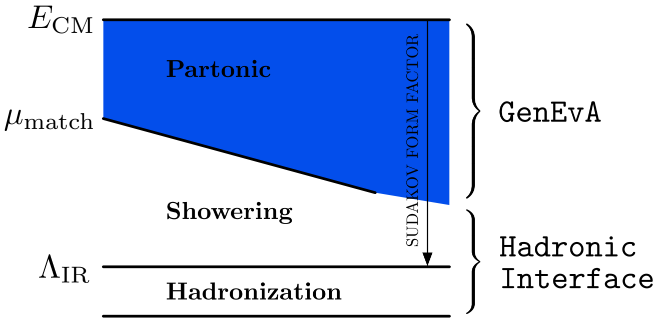
Idea Behind: Reweighting all the way

1. Use parton shower as phase-space generator to distribute points (events) in multiplicity, flavor and phase space
2. Determine *exact* probability $\mathrm{d}P$ for each point (event) to be generated
3. Reweight to desired distribution $\mathrm{d}\sigma$ on event-by-event basis

$$w = \mathrm{d}\sigma / \mathrm{d}P$$

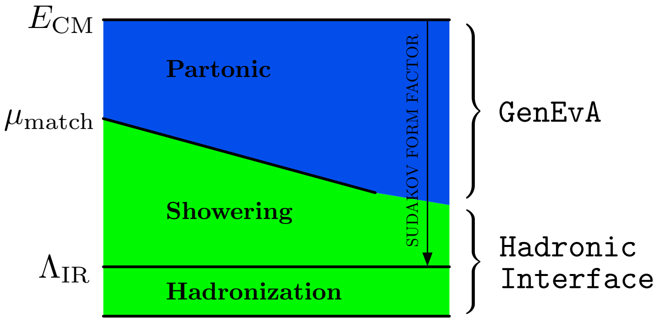
Outline

Three Regimes of Event Generation



- Partonic Regime: Includes full field theory information
  - Partonic calculation with the notion of a matching scale

Three Regimes of Event Generation



- Partonic Regime: Includes full field theory information
  - Partonic calculation with the notion of a matching scale
- Showering Regime: Defined in terms of splitting probabilities
  - Parton shower that can start showering at a given scale
- Hadronization: e.g. PYTHIA, HERWIG

Effective Field Theory Picture

Partonic Regime

- Equivalent to consecutive matching of QCD onto SCET [Bauer, Schwartz]
  - Match full QCD with n emissions onto SCET
  - Run in SCET between scales of n and $n+1$ emissions
- At LL can use QCD matrix elements multiplied by LL Sudakovs

Showering Regime

- Equivalent to additional emissions and LL running in SCET [Bauer, Schwartz]
  - μ_{match} is matching scale of last full QCD emission
- Identify evolution variable (e.g. $\sqrt{Q^2}$ or p_T) of parton shower with running matching scale
  - Clean separation between partonic and showering regimes

Dependence on Matching Scale μ_{match}

Showering has double-logarithmic dependence on scale μ_{match}

- (Leading) scale dependence is property of QCD not a specific shower
- Cancellation is theoretical not algorithmic problem
- Sufficient to include neccessary scale dependence in calculations

In GENEVA

- μ_{match} dependence in partonic regime can be included
  - Analytically: using log-resummed calculation (e.g. SCET)
  - Numerically: on-the-fly via parton shower phase-space generation
- Value of μ_{match} is fluid and determined on event-by-event basis
  - Increase μ_{match}: fewer partonic emissions, more showering
  - Decrease μ_{match}: more partonic emissions, less showering
  - Smooth transition between partonic and showering regimes

Master Formula

$$\sigma = \sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})\,\mathrm{MC}(\mu_{\mathrm{match}}^{(0)})$$

$\mathrm{MC}(\mu_{\mathrm{match}})$: Showering starting at μ_{match}

Partonic cross sections:

- $\sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})$: No emissions above scale $\mu_{\mathrm{match}}^{(0)}$
  - $\sigma_p^{(0)}(E_{\mathrm{CM}})$ is total inclusive cross section
  - Choose $\mu_{\mathrm{match}}^{(0)} = E_{\mathrm{CM}}$ to avoid dead zone

Master Formula

$$\sigma = \sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})\,\mathsf{MC}(\mu_{\mathrm{match}}^{(0)}) + \int_{\mu_{\mathrm{match}}^{(0)}}^{E_{\mathrm{CM}}} \mathrm{d}\mu_1\, \mathrm{d}\sigma_p^{(1)}(\mu_1;\mu_{\mathrm{match}}^{(1)})\,\mathsf{MC}(\mu_{\mathrm{match}}^{(1)})$$

$\mathsf{MC}(\mu_{\mathrm{match}})$: Showering starting at μ_{match}

Partonic cross sections:

- $\sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})$: No emissions above scale $\mu_{\mathrm{match}}^{(0)}$
  - $\sigma_p^{(0)}(E_{\mathrm{CM}})$ is total inclusive cross section
  - Choose $\mu_{\mathrm{match}}^{(0)}$ some low scale (could be as low as Λ_{IR})
- $\sigma_p^{(1)}(\mu_1;\mu_{\mathrm{match}}^{(1)})$: One emission at μ_1, no further ones above $\mu_{\mathrm{match}}^{(1)}$
  - Choose $\mu_{\mathrm{match}}^{(1)} = \mu_1$ to avoid dead zone

Master Formula

$$\sigma = \sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})\,\mathrm{MC}(\mu_{\mathrm{match}}^{(0)}) + \int_{\mu_{\mathrm{match}}^{(0)}}^{E_{\mathrm{CM}}} \mathrm{d}\mu_1\, \mathrm{d}\sigma_p^{(1)}(\mu_1; \mu_{\mathrm{match}}^{(1)})\,\mathrm{MC}(\mu_{\mathrm{match}}^{(1)})$$

$\mathrm{MC}(\mu_{\mathrm{match}})$: Showering starting at μ_{match}

Partonic cross sections:

- $\sigma_p^{(0)}(\mu_{\mathrm{match}}^{(0)})$: No emissions above scale $\mu_{\mathrm{match}}^{(0)}$
  - $\sigma_p^{(0)}(E_{\mathrm{CM}})$ is total inclusive cross section
  - Choose $\mu_{\mathrm{match}}^{(0)}$ some low scale (could be as low as Λ_{IR})
- $\sigma_p^{(1)}(\mu_1; \mu_{\mathrm{match}}^{(1)})$: One emission at μ_1, no further ones above $\mu_{\mathrm{match}}^{(1)}$
  - Choose $\mu_{\mathrm{match}}^{(1)} = \mu_1$ to avoid dead zone
- All IR singularities in $\sigma_p^{(i)}$ cancel *analytically*
- Everything boils down to calculating partonic cross sections

Benefits of GENEVA Approach

Reweighting allows free choice of $d\sigma$

- Use *any* matrix element: multi-leg, NLO, NNLO, ...
- Use *any* matching/resummation scheme
  - Fixed-order matrix elements times your favourite Sudakov
  - Analytically resummed expressions (NLL, NNLL)

Phase space is covered exactly once

- No double-counting or negative weights
- Easy to combine descriptions for different parts of phase space, e.g.
  - 2+3 partons: analytic NNLO matrix element
  - 4+5 partons: One-loop recursion relations [Bern et al.; Ellis et al.]
  - 6 partons: MADGRAPH
  - 7 partons: AMEGIC++
  - 8 partons: ALPGEN
  - 9 partons: O'MEGA
  - 10+ partons: Tree-level recursion relations [Berends, Giele; BCF]
  - 15+ partons: Parton Shower with quantum interference [Nagy, Soper]

GENEVA in a Toy Model

Toy cross sections ($0 < x < 1$)

$$\frac{d\sigma_B}{dx} = B\delta(x)\,, \qquad \frac{d\sigma_V}{dx} = a\left(\frac{B}{\epsilon} + V\right)\delta(x)\,, \qquad \frac{d\sigma_R}{dx} = aB\frac{R(x)}{x^{1+\epsilon}}$$

$$\sigma_{\text{total}} = B + aV + aB\int_0^1 dx\,\frac{R(x)-1}{x} \qquad [R(x\to 0)\to 1]$$

GENEVA in a Toy Model

Toy cross sections ($0 < x < 1$)

$$\frac{d\sigma_B}{dx} = B\delta(x)\,, \qquad \frac{d\sigma_V}{dx} = a\left(\frac{B}{\epsilon} + V\right)\delta(x)\,, \qquad \frac{d\sigma_R}{dx} = aB\frac{R(x)}{x^{1+\epsilon}}$$

$$\sigma_{\text{total}} = B + aV + aB\int_0^1 dx\,\frac{R(x) - 1}{x} \qquad [R(x \to 0) \to 1]$$

Toy shower (identify $x \equiv \mu$)

$$Q(x) \xrightarrow{x \to 0} R(x)\,, \qquad \Delta_Q(x_1, x_2) = \exp\left(-a\int_{x_2}^{x_1} dx\,\frac{Q(x)}{x}\right)$$

$$\text{MC}(x_{\text{start}}) = \Delta_Q(x_{\text{start}}, x_{\text{end}}) + \int_{x_{\text{end}}}^{x_{\text{start}}} dx\, a\,\frac{Q(x)}{x}\,\Delta_Q(x_{\text{start}}, x)\,\text{MC}(x)$$

$$= \Delta_Q(x_{\text{start}}, x_{\text{end}}) + [1 - \Delta_Q(x_{\text{start}}, x_{\text{end}})] = 1$$

Partonic Cross Sections at Tree Level

Tree-level generator

Take
$$\sigma_p^{(0)}(x_{\mathrm{match}}) = B$$
$$\sigma_p^{(1)}(x) = aB\,\frac{R(x)}{x}$$

Get
$$\sigma = B + aB \int_{x_{\mathrm{match}}}^{1} \mathrm{d}x\, \frac{R(x)}{x}$$

$$\sigma^{(1)}(x) = aB\,\frac{1}{x}\begin{cases} R(x) & (x > x_{\mathrm{match}}) \\ Q(x)\,\Delta_Q(x_{\mathrm{match}}, x) & (x < x_{\mathrm{match}}) \end{cases}$$

- σ correct to $\mathcal{O}(1)$, and $\sigma^{(1)}(x)$ to $\mathcal{O}(a)$
- Logarithmic x_{match} dependence in σ and $\sigma^{(1)}(x)$

Partonic Cross Sections at Tree Level

Tree-level generator Sudakov improved

Take
$$\sigma_p^{(0)}(x_{\mathrm{match}}) = B\,\Delta_Q(1, x_{\mathrm{match}})$$
$$\sigma_p^{(1)}(x) = aB\,\frac{R(x)}{x}\,\Delta_Q(1, x)$$

Get
$$\sigma = B + aB\int_{x_{\mathrm{match}}}^{1} \mathrm{d}x\,\frac{R(x) - Q(x)}{x}\,\Delta_Q(1, x)$$

$$\sigma^{(1)}(x) = aB\,\frac{1}{x}\begin{cases} R(x)\,\Delta_Q(1, x) & (x > x_{\mathrm{match}}) \\ Q(x)\,\Delta_Q(1, x_{\mathrm{match}})\Delta_Q(x_{\mathrm{match}}, x) & (x < x_{\mathrm{match}}) \end{cases}$$

- σ correct to $\mathcal{O}(1)$, and $\sigma^{(1)}(x)$ to $\mathcal{O}(a)$
- Logarithmic x_{match} dependence in σ and $\sigma^{(1)}(x)$ cancels
  - ME/PS merging [CKKW, MLM], but no special algorithm needed
  - *Any* $\tilde{Q}(x \to 0) \to 1$ works, gives $\Delta_{\tilde{Q}}$ with correct LL

Partonic Cross Sections at NLO

NLO Slicing

Take $\sigma_p^{(0)}(x_{\mathrm{match}}) = \tilde{B}$

$$\tilde{B} = \sigma_{\mathrm{total}} - aB \int_{x_{\mathrm{match}}}^{1} \mathrm{d}x \, \frac{R(x)}{x}$$

$$\sigma_p^{(1)}(x) = aB \, \frac{R(x)}{x}$$

Get $\sigma = \sigma_{\mathrm{total}}$

$$\sigma^{(1)}(x) = a \, \frac{1}{x} \begin{cases} BR(x) & (x > x_{\mathrm{match}}) \\ \tilde{B} Q(x) \, \Delta_Q(x_{\mathrm{match}}, x) & (x < x_{\mathrm{match}}) \end{cases}$$

- By construction $\sigma = \sigma_p^{(0)}(1) = \sigma_{\mathrm{total}}$, independent of x_{match}
- Two sources of logarithmic x_{match} dependence in $\sigma^{(1)}(x)$

Partonic Cross Sections at NLO

NLO Slicing Sudakov improved = NLO Subtraction [MC@NLO: Frixione, Webber]

Take $\sigma_p^{(0)}(x_{\mathrm{match}}) = \tilde{B}\,\Delta_Q(1, x_{\mathrm{match}})$

$$\tilde{B} = \sigma_{\mathrm{total}} - aB\int_{x_{\mathrm{match}}}^{1} \mathrm{d}x\,\frac{R(x) - Q(x)}{x}$$

$$\sigma_p^{(1)}(x) = aB\,\frac{R(x) - Q(x)}{x} + a\tilde{B}\,\frac{Q(x)}{x}\,\Delta_Q(1, x)$$

Get $\sigma = \sigma_{\mathrm{total}}$

$$\sigma^{(1)}(x) = a\,\frac{1}{x}\begin{cases} BR(x) + (\tilde{B} - B)Q(x)\,\Delta_Q(1, x) & (x > x_{\mathrm{match}}) \\ \tilde{B}Q(x)\,\Delta_Q(1, x_{\mathrm{match}})\Delta_Q(x_{\mathrm{match}}, x) & (x < x_{\mathrm{match}}) \end{cases}$$

- By construction $\sigma = \sigma_p^{(0)}(1) = \sigma_{\mathrm{total}}$, independent of x_{match} (still)
- Two sources of logarithmic x_{match} dependence in $\sigma^{(1)}(x)$ cancel

Partonic Cross Sections at NLO

Simple NLO [inspired by POWHEG: Nason]

Take $\sigma_p^{(0)}(x_{\rm match}) = \sigma_{\rm total}\,\Delta_T(1, x_{\rm match})$

$$\sigma_p^{(1)}(x) = a\,\sigma_{\rm total}\,\frac{T(x)}{x}\,\Delta_T(1, x)$$

$$T(x) = \frac{B}{\sigma_{\rm total}}\,R(x) \qquad \Delta_T = \exp\Big(-a\int \mathrm{d}x\,\frac{T(x)}{x}\Big)$$

Get $\sigma = \sigma_{\rm total}$

$$\sigma^{(1)}(x) = a\,\frac{1}{x}\begin{cases} B R(x)\,\Delta_T(1, x) & (x > x_{\rm match}) \\ \sigma_{\rm total}\,Q(x)\,\Delta_T(1, x_{\rm match})\Delta_Q(x_{\rm match}, x) & (x < x_{\rm match}) \end{cases}$$

- By construction $\sigma = \sigma_p^{(0)}(1) = \sigma_{\rm total}$, independent of $x_{\rm match}$
- No logarithmic $x_{\rm match}$ dependence in $\sigma^{(1)}(x)$

Outline

The Parton Shower as Phase Space Generator

Advantages

- Automatically covers all of multiplicity, flavor, phase space
- Automatically has the right singularity structure
- As a by-product, resums leading logarithms
- It's fast!

Challenges

- Need to know precise probability distribution $\mathrm{d}P$

  Use analytic parton shower algorithm [Bauer, FT, arXiv:0705.1719]

  - Exact momentum conservation and phase-space coverage
  - Full analytic control over $\mathrm{d}P$

- Must take into account that phase space is covered multiple times by parton shower

The Event Weight

$$w = \frac{\mathrm{d}\sigma}{\mathrm{d}P}$$

The Event Weight

$$w = \frac{\sigma(\Phi)\mathrm{d}\Phi}{\mathrm{d}P}$$

- $\mathrm{d}\sigma = \sigma(\Phi)\,\mathrm{d}\Phi$ is function of Lorentz invariant phase space Φ

The Event Weight

$$w = \frac{\sigma(\Phi)\mathrm{d}\Phi}{\mathcal{P}(\Sigma)\mathrm{d}\Sigma}$$

- $\mathrm{d}\sigma = \sigma(\Phi)\,\mathrm{d}\Phi$ is function of Lorentz invariant phase space Φ
- $\mathrm{d}P = \mathcal{P}(\Sigma)\,\mathrm{d}\Sigma$ is function of full parton shower history $\Sigma = \{t_i, z_i\}$

The Event Weight

$$w \equiv w(\Phi) = \frac{\sigma(\Phi)}{\mathcal{P}[\Sigma(\Phi)]J[\Sigma(\Phi)]}$$

- $\mathrm{d}\sigma = \sigma(\Phi)\,\mathrm{d}\Phi$ is function of Lorentz invariant phase space Φ
- $\mathrm{d}P = \mathcal{P}(\Sigma)\,\mathrm{d}\Sigma$ is function of full parton shower history $\Sigma = \{t_i, z_i\}$
- Mapping $\Sigma \to \Phi \equiv \Phi(\Sigma)$, Jacobian $J(\Sigma) = \mathrm{d}\Sigma/\mathrm{d}\Phi$

The Event Weight

$$w \equiv w(\Phi) = \frac{\sigma(\Phi)}{\sum_i \mathcal{P}[\Sigma_i(\Phi)] J[\Sigma_i(\Phi)]}$$

- $d\sigma = \sigma(\Phi)\, d\Phi$ is function of Lorentz invariant phase space Φ
- $dP = \mathcal{P}(\Sigma)\, d\Sigma$ is function of full parton shower history $\Sigma = \{t_i, z_i\}$
- Mapping $\Sigma \to \Phi \equiv \Phi(\Sigma)$, Jacobian $J(\Sigma) = d\Sigma/d\Phi$
- Each Φ can have multiple $\Sigma_i(\Phi)$ that map to it
  - Have to sum over all parton shower histories $\Sigma_i(\Phi)$ that map to the same point Φ in phase space.

Overcounting

$$w \equiv w(\Phi) = \frac{\sigma(\Phi)}{\sum_i \mathcal{P}[\Sigma_i(\Phi)] J[\Sigma_i(\Phi)]}$$

Summing over $\mathcal{P}[\Sigma_i(\Phi)]$ is hard

- Requires to explicitly construct all $\Sigma_i(\Phi)$ for given Φ
- Naively grows like $n!$
- Same problem in subtraction methods

Overcounting

$$w \equiv w(\Sigma) = \frac{\sigma(\Sigma)}{\mathcal{P}(\Sigma)}$$

Instead make weight function of Σ

Overcounting

$$w \equiv w(\Sigma) = \frac{\sigma[\Phi(\Sigma)]}{\mathcal{P}(\Sigma) J(\Sigma)}\, \hat{\alpha}(\Sigma)$$

Instead make weight function of Σ

- Define $\sigma(\Sigma) = \dfrac{\sigma[\Phi(\Sigma)]}{J(\Sigma)}\, \hat{\alpha}(\Sigma)$ with $\hat{\alpha}(\Sigma) = \dfrac{\alpha(\Sigma)}{\sum_i \alpha(\Sigma_i)}$

Overcounting

$$w \equiv w(\Sigma) = \frac{\sigma[\Phi(\Sigma)]}{\mathcal{P}(\Sigma)J(\Sigma)}\,\hat{\alpha}(\Sigma)$$

Instead make weight function of Σ

- Define $\sigma(\Sigma) = \dfrac{\sigma[\Phi(\Sigma)]}{J(\Sigma)}\,\hat{\alpha}(\Sigma)$ with $\hat{\alpha}(\Sigma) = \dfrac{\alpha(\Sigma)}{\sum_i \alpha(\Sigma_i)}$
- $\hat{\alpha}(\Sigma)$ distributes $\sigma(\Phi)$ among $\Sigma_i(\Phi)$, can be chosen freely
  - Trivial but inefficient: $\alpha(\Sigma) = 1$
  - Ideal but hard: $\alpha(\Sigma) = \mathcal{P}(\Sigma)J(\Sigma)$
- Pick $\alpha(\Sigma) \approx \mathcal{P}(\Sigma)J(\Sigma)$ such that $\sum_i \alpha(\Sigma_i)$ looks like sum of graphs
  - Compute with ALPHA algorithm [Caravaglios, Moretti; Mangano, Pittau (ALPGEN)]
  - Only grows like 2^n

Matrix Elements

Fixed-order tree-level matrix elements

- Various numerical algorithms/tools for ME with many final partons
- GENEVA can supply Sudakov factors for LL resummation numerically
  - Schematically $\mathcal{P}(\Sigma) \equiv Q(\Sigma)\Delta(\Sigma)$
  - Keep $\Delta(\Sigma)$ generated by parton shower during phase-space generation

$$w(\Sigma) = \frac{\sigma(\Sigma)\Delta(\Sigma)}{\mathcal{P}(\Sigma)} = \frac{\sigma(\Sigma)}{Q(\Sigma)}$$

Matrix Elements

Fixed-order tree-level matrix elements

- Various numerical algorithms/tools for ME with many final partons
- GENEVA can supply Sudakov factors for LL resummation numerically
  - Schematically $\mathcal{P}(\Sigma) \equiv Q(\Sigma)\Delta(\Sigma)$
  - Keep $\Delta(\Sigma)$ generated by parton shower during phase-space generation

$$w(\Sigma) = \frac{\sigma(\Sigma)\Delta(\Sigma)}{\mathcal{P}(\Sigma)} = \frac{\sigma(\Sigma)}{Q(\Sigma)}$$

NLO and resummed matrix elements

- Supply NLO calculations with LL Sudakovs analytically (see toy model)
- Analytic NLL or NNLL ME (e.g. from SCET) already include correct LL

Outline

Proof of Concept Version

In C++: $e^+e^- \to n\,j$ with $j = \{g,\, u,\, d,\, s,\, c\}$

- Use virtuality as evolution variable and running matching scale
- Use MADGRAPH tree-level matrix elements (currently up to $n = 6$)
  - Pure matrix elements
  - Sudakov improved resummed matrix elements
- NLO
  - One-loop $2j$ + tree-level $3j$ + LL resummation
  - + additional MADGRAPH tree-level matrix elements
- Showering regime covered by underlying analytic parton shower

On paper

- Matched NNLO for toy model

Weighted vs. Unweighted samples

$$N_{\text{eff}} = \frac{\left[\sum_n w_n\right]^2}{\sum_n w_n^2} \gg N_{\text{unw}} = \frac{\sum_n w_n}{w_{\max}}$$

- N_{unw}: expected number of events after unweighting
  - Poor efficiency from rare events with large weights
  - Systematic error due to unknown true $w_{\max}$ or truncation
- N_{eff}: measures statistical power of weighted sample
  - Number of events of a *statistically equivalent* unweighted sample
  - If N_{eff}/N too small, can partially unweight until $N_{\text{eff}} \lesssim N$

$e^+e^- \to 5j$	N_{eff}/N	N_{unw}/N	
MADEVENT	0.89	0.11	(partially unweighted)
		0.50	(truncated)
GENEVA	0.40	0.03	
	0.89	0.12	(partially unweighted)

Comparison with MadGraph/MadEvent

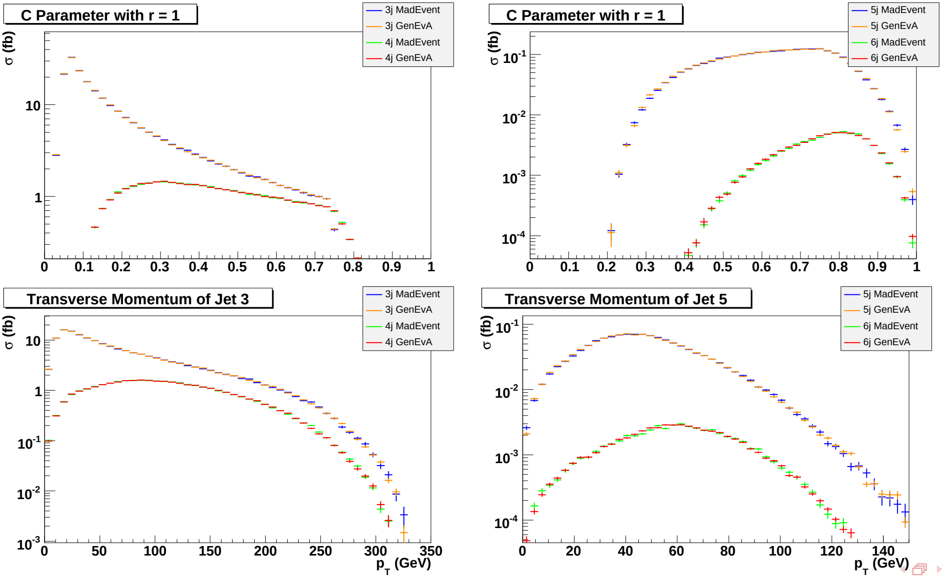
Reweight to pure MADGRAPH ME and compare with MADEVENT

Cross section in ab for

- $E_{\mathrm{CM}} = 1000\ \mathrm{GeV}$
- $\Lambda_{\mathrm{IR}} = 100\ \mathrm{GeV}$

process	MADEVENT	GENEVA
$4j$	36483 ± 49	36439 ± 69
$u\bar{u}gg$	14055 ± 32	14003 ± 44
$d\bar{d}gg$	3490 ± 9	3498 ± 22
$u\bar{u}c\bar{c}$	283.4 ± 1.3	273 ± 7
$u\bar{u}d\bar{d}$	175.9 ± 0.9	184 ± 6
$u\bar{u}u\bar{u}$	131.9 ± 0.9	135 ± 4
$5j$	2540.5 ± 3.3	2550 ± 6
$u\bar{u}ggg$	909.8 ± 2.1	916 ± 3
$d\bar{d}ggg$	227.4 ± 1.0	229 ± 2
$u\bar{u}c\bar{c}g$	54.44 ± 0.31	54 ± 1
$u\bar{u}d\bar{d}g$	33.96 ± 0.31	35 ± 1
$u\bar{u}u\bar{u}g$	25.41 ± 0.16	25 ± 1

Comparison with MadGraph/MadEvent



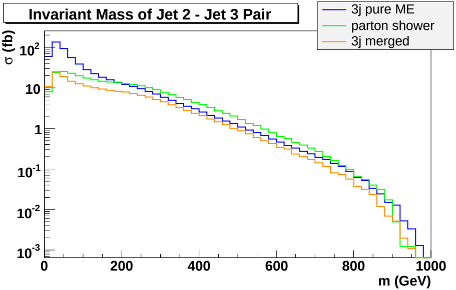
Matching Matrix Elements and Parton Shower

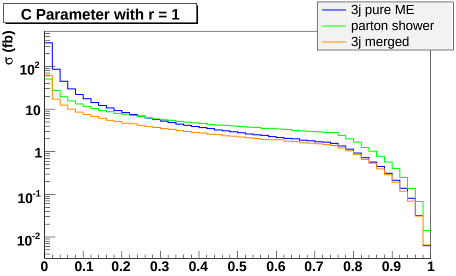
Match MADGRAPH ME

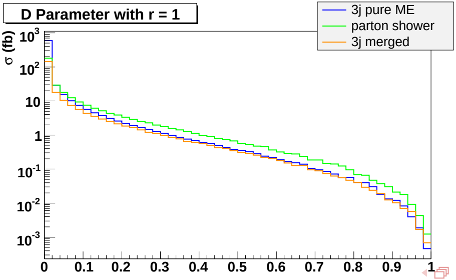
Compare

- pure $3j$ (+ parton shower)
- pure $2j$ (parton shower only)
- matched $3j$

$$E_{\mathrm{CM}} = 1000\,\mathrm{GeV}$$
$$\mu_{\mathrm{match}} = 50\,\mathrm{GeV}$$
$$\Lambda_{\mathrm{IR}} = 10\,\mathrm{GeV}$$







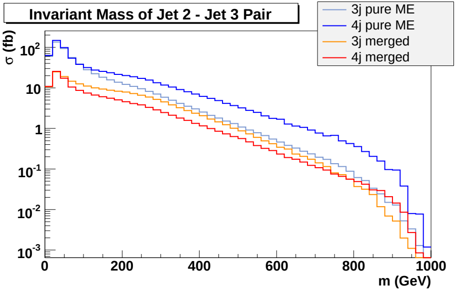
Matching Matrix Elements and Parton Shower

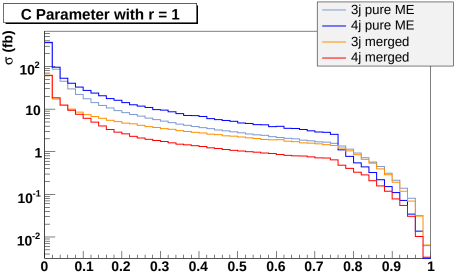
Match MADGRAPH ME

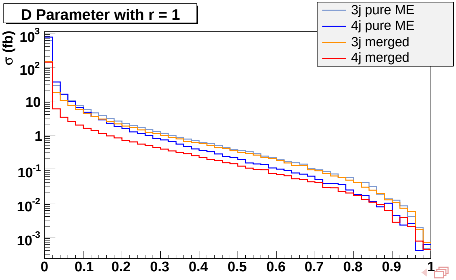
Compare

- pure $3j$ (+ parton shower)
- pure $4j$ (+ parton shower)
- matched $3j$
- matched $4j$

$$E_{\mathrm{CM}} = 1000\,\mathrm{GeV}$$
$$\mu_{\mathrm{match}} = 50\,\mathrm{GeV}$$
$$\Lambda_{\mathrm{IR}} = 10\,\mathrm{GeV}$$







Fully Inclusive Sample

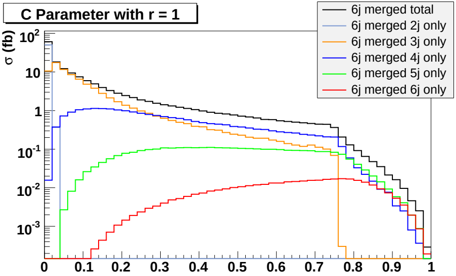
Match MADGRAPH ME

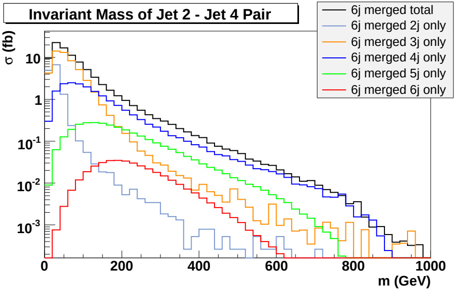
Inclusive sample matched up to $6j$

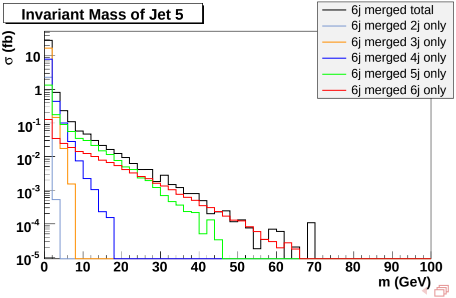
Contributions of individual

$2j, 3j, 4j, 5j, 6j$

$$E_{\mathrm{CM}} = 1000\,\mathrm{GeV}$$
$$\mu_{\mathrm{match}} = 50\,\mathrm{GeV}$$
$$\Lambda_{\mathrm{IR}} = 10\,\mathrm{GeV}$$







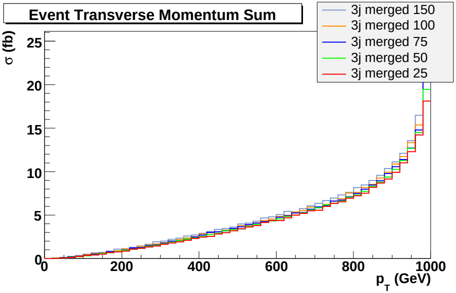
NLO

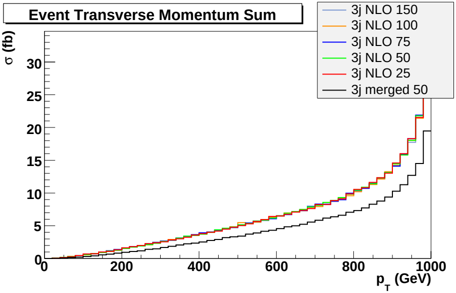
Compare

- $3j$ merged
- $3j$ fully matched at NLO

for different $\mu_{\text{match}} = 150, 100, 75, 50, 25\ \text{GeV}$

$$E_{\text{CM}} = 1000\ \text{GeV}$$
$$\Lambda_{\text{IR}} = 10\ \text{GeV}$$





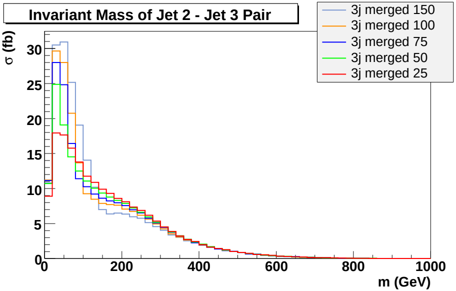
NLO

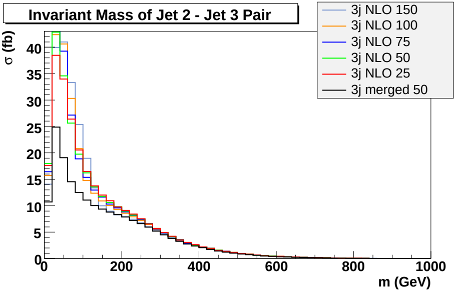
Compare

- $3j$ merged
- $3j$ fully matched at NLO

for different $\mu_{\text{match}} = 150, 100, 75, 50, 25\ \text{GeV}$

$$E_{\text{CM}} = 1000\ \text{GeV}$$
$$\Lambda_{\text{IR}} = 10\ \text{GeV}$$



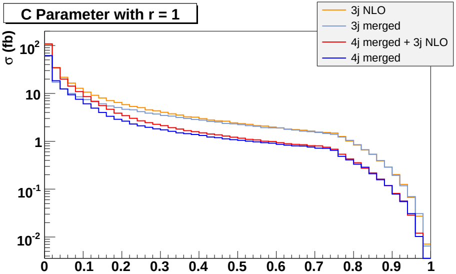


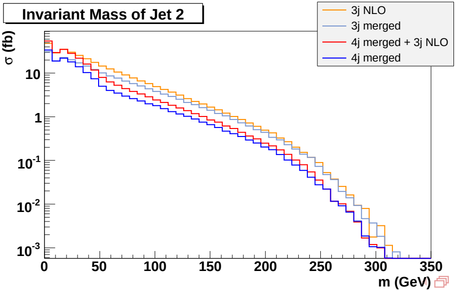
NLO with Additional Tree-level

Compare

- $3j$ matched at NLO
- $3j$ merged
- $4j$ merged with $3j$ at NLO
- $4j$ merged

$$E_{\mathrm{CM}} = 1000\ \mathrm{GeV}$$
$$\Lambda_{\mathrm{IR}} = 10\ \mathrm{GeV}$$

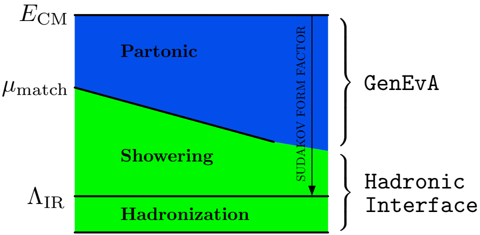




Conclusions and Outlook

GENEVA is a *framework* to turn theory calculations into exclusive events

- Analytic parton shower serves as efficient phase space generator
- Can use any partonic calculation
- Simple matching, independent of jet algorithm and parton shower
- Easy to combine different calculations (e.g. NLO + more tree-level ME)



Conclusions and Outlook

In the future

- Reweighting approach allows to
  - study uncertainties using single simulated event sample
  - update events to newest theory results even *after detector simulation*
- Extend philosophy to full range of interesting processes
  - pp, $W/Z/t$ production, SUSY, ...