Computation of QCD cross sections of large multiplicity final states at NLO

Zoltán Trócsányi, Univ. of Debrecen

CERN, 10.08.09

- Consensus on
  - need for NLO corrections of high multiplicity final state processes at the LHC
  - theoretical basis is known and robust
  - need for automatization

- Consensus on
  - need for NLO corrections of high multiplicity final state processes at the LHC
  - theoretical basis is known and robust
  - need for automatization
- Competition in actual
  - implementations
  - applications

- Consensus on
  - need for NLO corrections of high multiplicity final state processes at the LHC
  - theoretical basis is known and robust
  - need for automatization
- Competition in actual
  - implementations
  - applications
- Latter
  - is good for making improvements
  - could lead to unnecessary repetitions (waste)

Workshops could be useful to reduce waste

(at the end efficient and correct codes are needed)

Cross sections at NLO

$$\sigma = \sigma^{\mathrm{LO}} + \sigma^{\mathrm{NLO}} + \ldots$$

Born:

$$\sigma^{\mathrm{LO}} = \int_m \mathrm{d}\sigma_m^{\mathrm{B}} = \int \mathrm{d}\Phi_m \left| M_m^{(0)} \right|^2 J_m$$

NLO:

$$\sigma^{\mathrm{NLO}} = \int_m \mathrm{d}\sigma_m^{\mathrm{V}} + \int_{m+1} \mathrm{d}\sigma_{m+1}^{\mathrm{R}}$$

$$= \int \mathrm{d}\Phi_m \, 2\,\mathrm{Re}\left(M_m^{(0)*} M_m^{(0)} \right) J_m + \int \mathrm{d}\Phi_{m+1} \left| M_{m+1}^{(0)} \right|^2 J_{m+1}$$

Born cross section

- Well understood
- Robust tools are available (too many?)

Born cross section

- Well understood
- Robust tools are available (too many?)
- Most important messages for tree level:
  - solutions to efficient PS generation (PHEGAS, MADEVENT)
  - recursive aproach to amplitudes: BG is most efficient for large *m*
  - MC summation over helicity is useful
  - MC summation over colour is the 'only practical' way for 'colourful' processes

Virtual corrections: one-loop amplitudes

- Long-standing bottleneck broken by on-shell methods
  - generalized unitarity
  - complex momenta
  - one-loop integral basis

Virtual corrections: one-loop amplitudes

- Long-standing bottleneck broken by on-shell methods
  - generalized unitarity
  - complex momenta
  - one-loop integral basis
- many (too many?) numerical implementations:
  - c++: BlackHat (2), Winter, Lazopoulos
  - f90: HELAC-1LOOP/CutTools, Rocket, SUSYloop90 (Kardos)

Virtual corrections: one-loop amplitudes

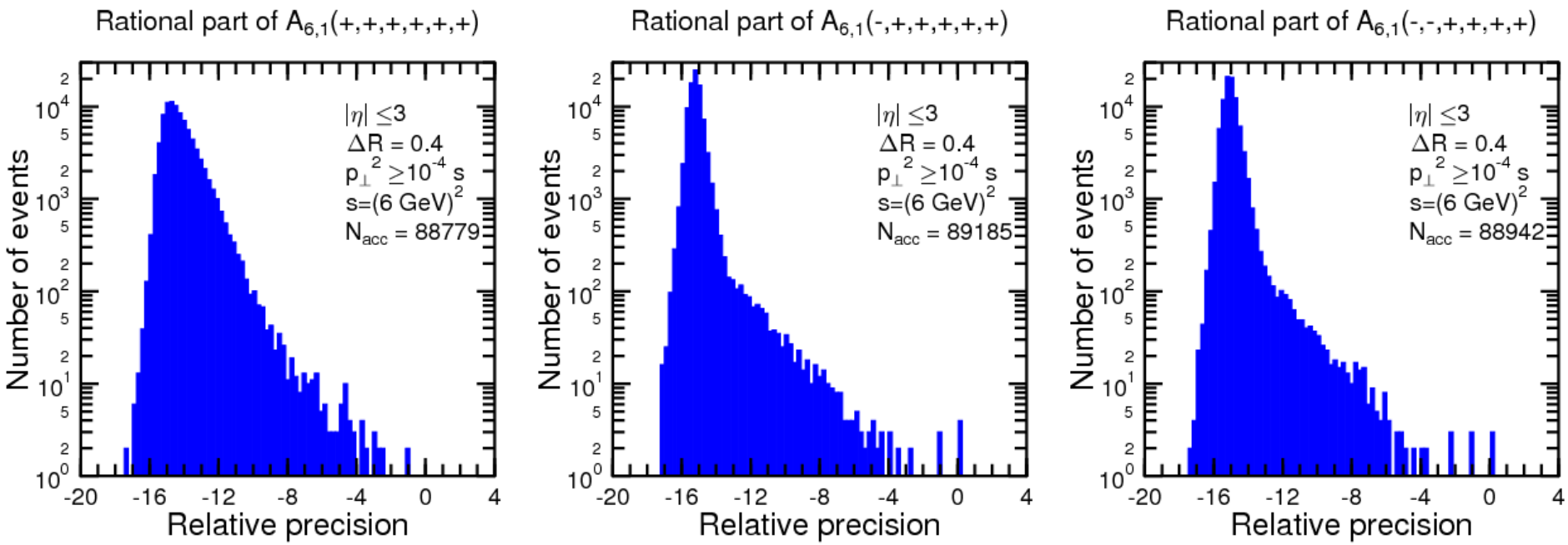
- Long-standing bottleneck broken by on-shell methods
  - generalized unitarity
  - complex momenta
  - one-loop integral basis
- many (too many?) numerical implementations:
  - c++: BlackHat (2), Winter, Lazopoulos
  - f90: HELAC-1LOOP/CutTools, Rocket, SUSYloop90 (Kardos)
- Essentially
  - same for cut-constructible (C) parts
  - different in computing the rational (R) parts

SUSYloop90

- BG recursion for tree amplitudes
- SUSY decomposition for loop ones
- Calcutaion is in d=4 for both C and R

SUSYloop90

- BG recursion for tree amplitudes
- SUSY decomposition for loop ones
- Calcutaion is in d=4 for both C and R



Precision plots for six-gluon amplitudes

Real corrections

- Theoretical basis well understood
- Most implementations use subtractions:
  - dipole (CS+DT)
  - antennae (GGG)
  - residuum (FKS)

Real corrections

- Theoretical basis well understood
- Most implementations use subtractions:
  - dipole (CS+DT)
  - antennae (GGG)
  - residuum (FKS)
- Majority of calculations used dipole subtractions in the last decade (except MC@NLO), but it
  - cannot directly be generalized to NNLO

    ($\Longrightarrow$ antennae?)
  - inconvenient for matching with PS
  - becomes inefficient for multileg processes due to priliferation of dipoles ($\Longrightarrow$ FKS?)

Results from studies at NNLO

- Easy separation of soft & collinear singularities (to all orders)
- General subtraction scheme at NNLO

Results from studies at NNLO

- Easy separation of soft & collinear singularities (to all orders)
- General subtraction scheme at NNLO
- New scheme at NLO that solves previous objections simultaneously
  - minimum number of subtraction terms at NLO (as FKS)
  - uses exact PS factorisations/convolutions (similarly to CS)
  - permutation symmetry of the SME (if any) is resepcted (as FKS)
  - works at any order in PT