An automatized tool for multileg NLO computations

Benedikt Biedermann
September, 29th 2009

Content:

1. NLO Computations for the LHC
2. Unitarity Methods
3. Tree-level recursion
4. Outlook

NLO Computations for the LHC

LHC = QCD-Machine:

•Data analysis requires precise knowledge of the standard model signal
$\longrightarrow$ relevant for signals of new physics

High energy pp-scattering gives rise to multi-leg parton ampliudes!

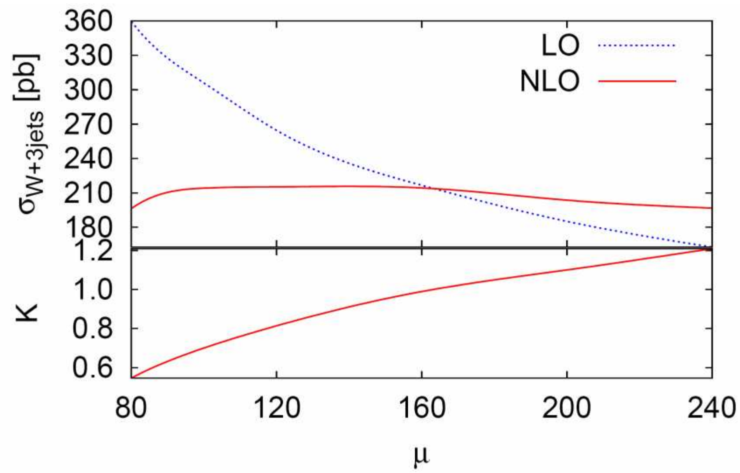
- Various automatized tools for tree-level amplitudes exist (e.g. MadGraph, HELAC)
- Often recursive techniques encountered in the packages (e.g. Berends-Giele recursion)

Why NLO amplitudes?

QCD is an asymptotically free theory:
Necessary to consider the running of the coupling

Tree-level approximation is the classical approximation i.e. no quantum corrections

→ Cross section cannot depend on renormalization scale



Inclusive W^++3 jet cross-section at the LHC and the K-factor defined as $K = \sigma_{NLO}/\sigma_{LO}$

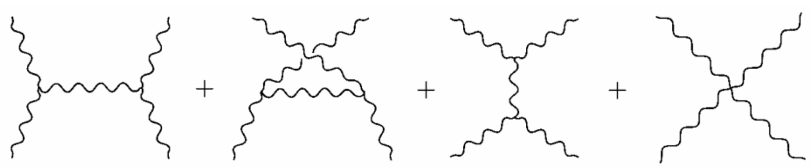
Feynman-Diagrams

Diagrammatic traditional approach
to handle perturbation theory

Recipe:

- compute the Feynman diagrams
- sum up all the diagrams in order to get the amplitude

Example: Tree-level gluon scattering

$\mathcal{A} =$ 

Disadvantages of Diagrams

- Number of diagrams grows rapidly with number of external legs
- Most of the parts in the calculation cancel each other

# external gluons	# diagrams
4	4
5	25
6	220
7	~ 2‘500
8	~ 35‘000
9	~ 500‘000
10	~ 10‘000‘000

Very simple analytic result for MHV amplitudes:
MHV = **M**aximally **H**elicity **V**iolating

$$A_{jk}^{\text{tree MHV}} \equiv A_n^{\text{tree}}(1^+,\ldots,j^-,\ldots,k^-,\ldots,n^+)$$

$$= i\,\frac{\langle j\,k\rangle^4}{\langle 1\,2\rangle\cdots\langle n\,1\rangle}$$ (Parke-Taylor formula)

Spinor Products: $\langle j\,k\rangle = \overline{u}_-(p_j)u_+(p_k)$

Unitarity based Methods

Traditional way: Passarino-Veltman Reduction [Passarino,Veltman1979]

Different ansatz:

- Work directly with the amplitude instead of diagrams
- Write amplitude in terms of a basis of integrals $\mathcal{J}_j$

[Bern,Dixon, Kosower1995]

$$\mathcal{A} = \sum_{j \in B} c_j \mathcal{J}_j$$

Difficulty:
Determination of the coefficients c_j

New deal: Unitarity, OPP [Ossola,Papadopoulos, Pittau2007]

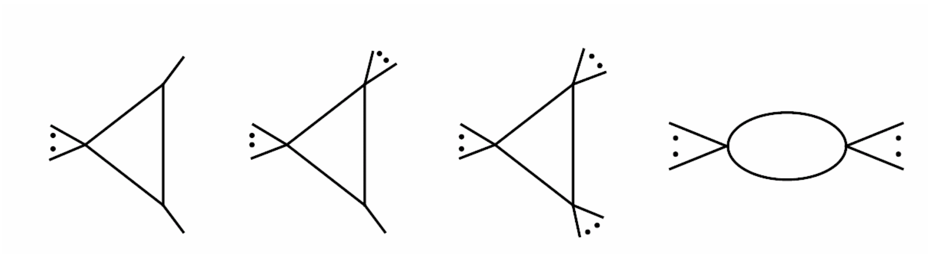
Example: scalar box integral

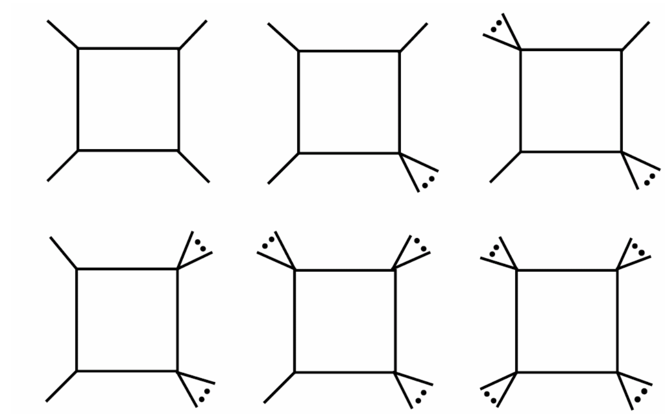
$$\mathcal{J}_j = \int d^n \overline{q} \frac{1}{\overline{D}_1 \overline{D}_2 \overline{D}_3 \overline{D}_4}$$

$$\overline{D}_i = (\overline{p}_i + \overline{q})^2 - m_i^2$$

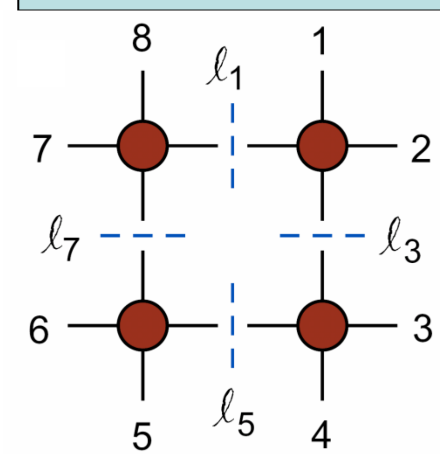
Unitarity based Methods

Different types of Feynman integrals:





- Use “artificial” tree-amplitudes to glue together one-loop ampliudes
- Lines connecting different tree amplitudes are **on-shell**



$$A_N = A_N^{CC} + A_N^R$$

One loop amplitudes as sum of cut-constructible and rational parts:

Present Aim

Use these techniques in order to compute LHC relevant processes in a rather automatized way.

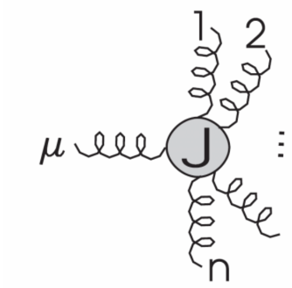
Berends-Giele recursion relation

[Berends,Giele1987]

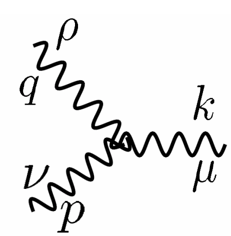
Recursion relation to compute color-ordered tree-level amplitudes:

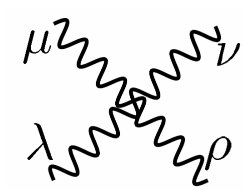
$$A(1,...,n) = g^{n-2} \sum_{\mathcal{P}(2,3,...,n)} \mathrm{Tr}[a_1 a_2 ... a_n] A^{co}(1,2,...,n)$$

1. Compute (n+1)-gluon current $J(1,...,n)$ with
   - n external legs on-shell
   - (n+1)th external leg off-shell
2. Extract the (n+1)-gluon amplitude from $J(1,...,n)$

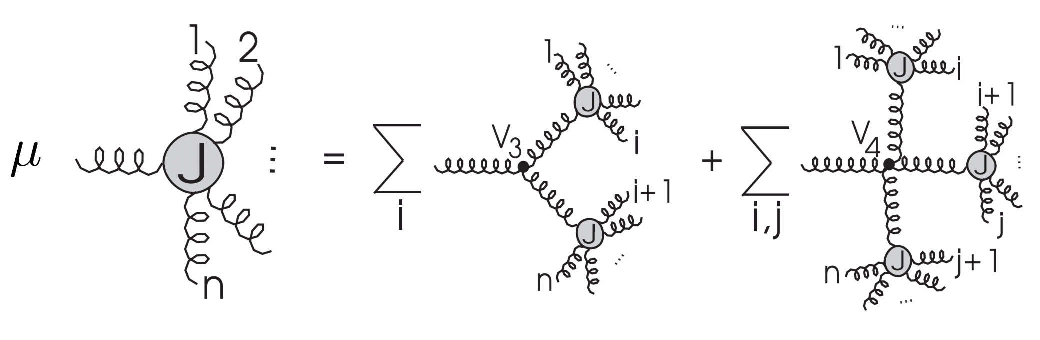


Vertices of pure Yang-Mills gauge theory:





Berends-Giele recursion relation



$$J(1,...,n) = \frac{1}{\kappa(1,n)^2}\left(\sum_{i=1}^{n-1}\Big[J(1,...,i),J(i+1,...,n)\Big]\right.$$
$$\left. + \sum_{k=1}^{n-2}\sum_{l=k+1}^{n-1}\Big\{J(1,...,k),J(k+1,...,l),J(l+1,...,n)\Big\}\right)$$

Momentum sum: $\kappa(1,n) = p_1 + p_2 + ... + p_n$

Color orderd amplitude:

$$\mathcal{A}(1,...,n+1) = J(n+1)\cdot J(1,...,n)\, i\, \kappa(1,n)^2|_{\kappa(1,n+1)=0}$$

First Implementation

direct recursion$\rightarrow$ function calls itself again until a termination condition is reached (i.e. J is a one-point current).

Program works correctly:

- comparison with MHV amplitudes
- gauge test: $J_\mu k^\mu = 0$

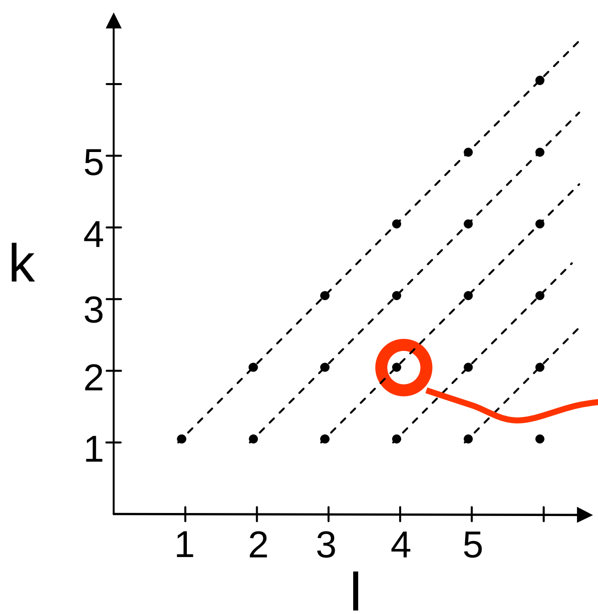
Disadvantage: Most currents are
computed several times

$\longrightarrow$ cpu-time $\sim 4^n$ 18 gluons ~ 4100 seconds

Implementation with cache

Store all the currents that one has used once and recycle them.

Idea: Start with the 1-pt currents and construct successively
all higher i-pt currents.



- Every dot represents one current
- diagonal dots = i-pt currents

$J(2,3,4): \; k=2, \; l=4$

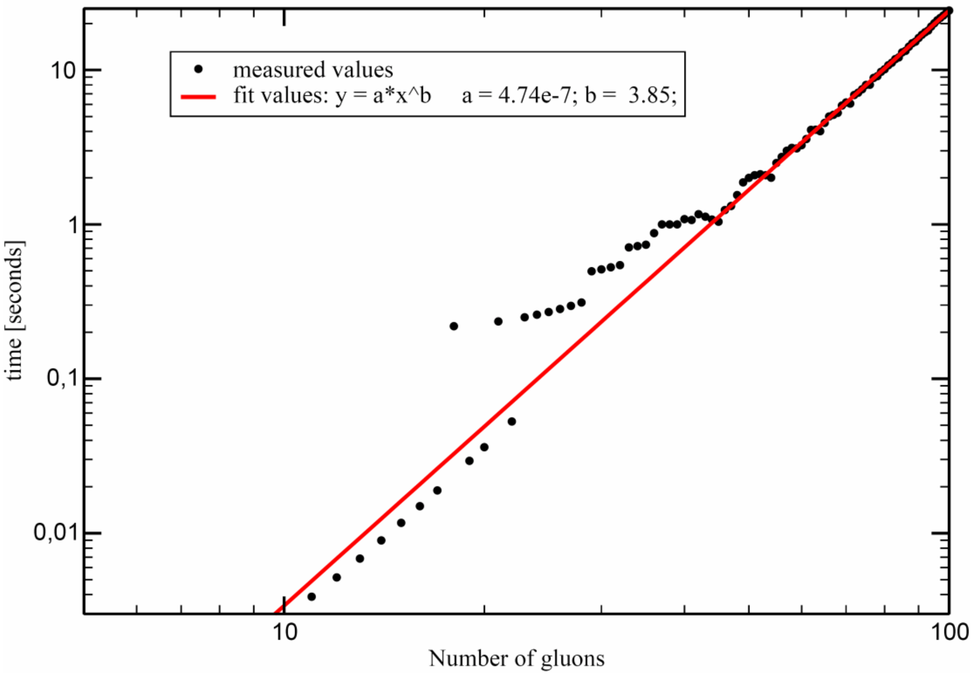
$$\#currents = \sum_{i=1}^{n} i = \frac{n(n+1)}{2}$$

Performance

polynomial scaling:
cpu-time $\sim n^4$

Checks:

- MHV amplitudes
- gauge test



```
 void Amplitude::direct_recursion(){
 for (int i=1; i<N_max; i++){
    int block_0 = block(i-1);
    int block_1 = block(i);
    int block_2 = block(i+1);
    int count = 0;
    int p = block_0+1;

    for (int kl=block_1; kl<block_2; kl++){
      for (int s=0; s<4; s++){
K[kl][s] = K[count][s] + K[p][s];
      }
      count++;
      p++;

      int k = kl - block_1;
      int l = k + i;

      for (int m=k; m<l; m++){
bracket(pos(k,m), pos(m+1,l), kl);
      }

      for (int m=k; m<l-1; m++){
  for (int n=m+1; n<l; n++){
    curly_bracket(pos(k,m), pos(m+1,n), pos(n+1,l), kl);
  }
      }

      if (kl != (N_max-1)*(N_max+2)/2){
for (int s=0; s<4; s++){
  J[kl][s] = J[kl][s]/lprod(K[kl],K[kl]);
  }
      }
    }
  }
}
```

Outlook

So far reached:

- Tree-level machinery works as building blocks for the unitarity methods

Work in progress:

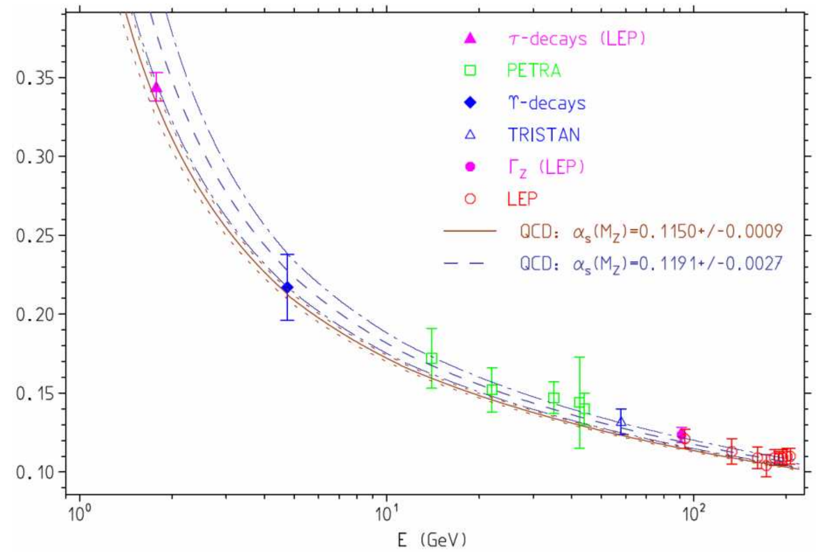
- Implementation of the one-loop unitarity method

Future perspectives:

- Inclusion of quarks
- Phenomenological calculations

Tree-level approximation is
the classical approximation
i.e. no quantum corrections

First quantum corrections
(and scale dependence) at NLO



Present experimental status of the QCD running coupling. Shown are RG solutions with αs(Mτ) extracted from τ–decays and the one with αs(MZ) measured at LEP as start values. The corresponding central values agree at the 1 σ level.

For the discovering machine the focus lies on NLO, while NNLO is for precision measurements

Optical Theorem

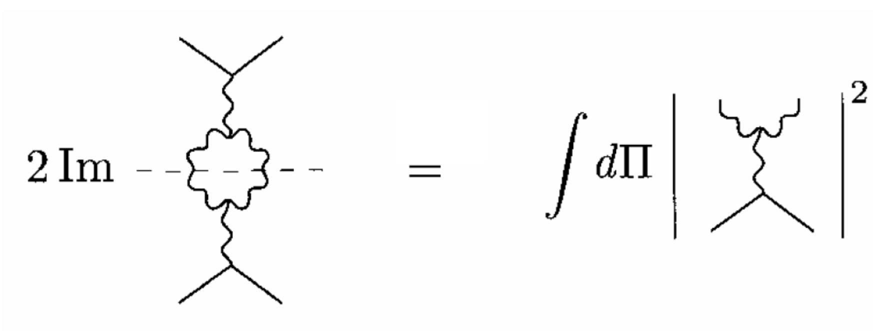
Based on the unitarity of the S-Matrix:

$$S = 1 + iT \longrightarrow -i(T - T^\dagger) = T^\dagger T$$

$$-i(T_{if} - T^*_{fi}) = \sum_k T^*_{fk} T_{ki}$$

$$\xrightarrow{\text{set i = f}} \quad \boxed{\operatorname{Im} A \sim \sigma_{\text{tot}}}$$

Relates imaginary part of the amplitude with the total cross section.



Unitarity based Methods

Master Integral Basis

$$A^{\text{1-loop}} = \sum_j c_j I_j$$

Unitarity

$$\mathrm{Im}\left(A^{\text{1-loop}}\right) \sim \int \mathrm{d}\Pi \, \left|A^{\text{tree}}\right|^2$$

$$\sum_j c_j \, \mathrm{Im}\left(I_j\right) \sim \int \mathrm{d}\Pi \, \left|A^{\text{tree}}\right|^2$$