Calculating gluon one-loop amplitudes numerically

Jan-Christopher Winter and Walter T. Giele *

Fermi National Accelerator Laboratory
P.O. Box 500, Batavia, IL 60510 - U.S.A.

This note reports on an independent implementation of calculating one-loop amplitudes semi-numerically using generalized unitarity techniques. The algorithm implemented in form of a C++ code closely follows the method by Ellis, Giele, Kunszt and Melnikov [1, 2]. For the case of gluons, the algorithm is briefly reviewed. Double-precision results are presented documenting the accuracy and efficiency of this computation [3].

1 Introduction

An automated next-to-leading order generator for Standard Model processes is highly desirable [4]. With recent developments of generalized unitarity [5] and parametric integration methods [6] such a generator seems to be within reach [1, 2, 7, 8]. A first crucial step is the development of stable and fast algorithms for evaluating one-loop amplitudes through generalized unitarity cuts. A C++ code is especially of interest because of the ease with which it can be integrated in leading-order generators such as COMIX [9]. The leading-order code will then be used to compute the cut graphs. Eventually, such a retrofitted generator will be able to generate all necessary amplitudes for a next-to-leading order Monte Carlo program for any Standard Model process of interest to the collider experiments.

2 Construct one-loop by tree-level amplitudes – algorithm in brief

The full N-gluon one-loop amplitude can be constructed from the leading colour-ordered amplitudes [10], which can be calculated by $A_N^{[1]}(\{p_i, \kappa_i\}) = A_N^{\mathrm{cc}} + R_N$ depending on external momenta p_i and polarizations κ_i. The cut-constructible part reads

$$A_N^{\mathrm{cc}} = \sum_{[i_1|i_4]} d_{i_1 i_2 i_3 i_4}^{(0)} I_{i_1 i_2 i_3 i_4}^{(4-2\epsilon)} + \sum_{[i_1|i_3]} c_{i_1 i_2 i_3}^{(0)} I_{i_1 i_2 i_3}^{(4-2\epsilon)} + \sum_{[i_1|i_2]} b_{i_1 i_2}^{(0)} I_{i_1 i_2}^{(4-2\epsilon)} \qquad (1)$$

employing the short-hand notation $[i_1, i_M] = 1 \le i_1 < i_2 < \ldots < i_M \le N$, and M denotes the number of cuts. The master integrals are defined as $I_{i_1 \cdots i_M}^{(D)} = \int d^D \ell \, (i\pi^{D/2} d_{i_1} \cdots d_{i_M})^{-1}$ and the inverse propagators d_i are functions of the loop momentum: $d_i(\ell) = (\ell + q_i - q_{i_M})^2$ with $q_k = \sum_{j=1}^k p_j$. The rational part R_N is represented by

$$R_N = -\sum_{[i_1|i_4]} \frac{d_{i_1 i_2 i_3 i_4}^{(4)}}{6} + \sum_{[i_1|i_3]} \frac{c_{i_1 i_2 i_3}^{(7)}}{2} - \sum_{[i_1|i_2]} \left(\frac{(q_{i_1} - q_{i_2})^2}{6} \right) b_{i_1 i_2}^{(9)}, \qquad (2)$$

cf. [2, 7]. For each possible cut configuration $i_1 \cdots i_M$, the box ($d_{i_1 i_2 i_3 i_4}^{(n)}$), triangle ($c_{i_1 i_2 i_3}^{(n)}$) and bubble ($b_{i_1 i_2}^{(n)}$) coefficients appearing above are found as solutions to the parametric form

---

LCWS/ILC 2008

of the unintegrated ordered one-loop amplitude,

$$\mathcal{A}_N^{(D_s)}(\ell) \;=\; \sum_{[i_1|i_5]} \frac{\bar{e}_{i_1 i_2 i_3 i_4 i_5}^{(D_s)}(\ell)}{d_{i_1} d_{i_2} d_{i_3} d_{i_4} d_{i_5}} + \sum_{[i_1|i_4]} \frac{\bar{d}_{i_1 i_2 i_3 i_4}^{(D_s)}(\ell)}{d_{i_1} d_{i_2} d_{i_3} d_{i_4}} + \sum_{[i_1|i_3]} \frac{\bar{c}_{i_1 i_2 i_3}^{(D_s)}(\ell)}{d_{i_1} d_{i_2} d_{i_3}} + \sum_{[i_1|i_2]} \frac{\bar{b}_{i_1 i_2}^{(D_s)}(\ell)}{d_{i_1} d_{i_2}} \,. \quad (3)$$

This decomposition of the integrand $\mathcal{A}_N^{(D_s)}(\ell) = \mathcal{N}^{(D_s)}(\ell)/(d_1 d_2 \cdots d_N)$ has been generalized to higher (integer) dimensionality of the internal particles, i.e. spin-polarization states and loop momenta respectively have dimension D_s and $D \le D_s$ (as required by dimensional regularization). The D_s dependence of the integrand can be eliminated by taking into account that the numerator only linearly depends on the spin-space dimension: $\mathcal{N}^{(D_s)}(\ell) = \mathcal{N}_0(\ell) + (D_s - 4)\,\mathcal{N}_1(\ell)$. Moreover, only up to 5-point terms (i.e. $M \le 5$) need to be included in the parametrization, since the loop momentum effectively has $4 + 1$ components only: $\mathcal{A}_N^{(D_s)}(\ell) = \mathcal{A}_N^{(D_s)}(\ell_1, \ldots, \ell_4, [-\sum_{i=5}^{D} \ell^2]^{1/2})$. These two additions essentially are sufficient to disentangle the rational part in the same way as the cut-constructible part [2].

The numerator functions of the M-point terms are polynomials encoding the loop-momentum dependence residing in the space orthogonal to the physical space defined by the external momenta. The orthogonal space is spanned by the basis vectors n_i. The form of the polynomials is richer compared to the four-dimensional case. They now include the coefficients that determine the rational part, e.g. for the box numerator, one finds

$$\bar{d}_{i_1 \cdots i_4}(\ell) \;=\; d_{i_1 \cdots i_4}^{(0)} + \alpha_4\, d_{i_1 \cdots i_4}^{(1)} + s_e^2\, [d_{i_1 \cdots i_4}^{(2)} + \alpha_4\, d_{i_1 \cdots i_4}^{(3)}] + s_e^4\, d_{i_1 \cdots i_4}^{(4)} \,, \quad (4)$$

where $\alpha_i = \ell \cdot n_i$ and $s_e^2 = -\alpha_5^2 - \ldots - \alpha_D^2$. To solve for the coefficients of the numerator functions, loop momenta $\ell = \ell_{i_1 \cdots i_M}$ have to be constructed such that $d_j(\ell_{i_1 \cdots i_M}) = 0$ for $j = i_1, \ldots, i_M$. Then

$$\bar{e}_{i_1 \cdots i_5}^{(D_s)}(\ell) \;=\; \mathrm{Res}_{i_1 \cdots i_5}\left[\mathcal{A}_N^{(D_s)}(\ell)\right] \;\equiv\; d_{i_1}(\ell) \cdots d_{i_5}(\ell)\, \mathcal{A}_N^{(D_s)}(\ell)\Big|_{d_{i_1}(\ell) = \cdots = d_{i_5}(\ell) = 0}, \quad (5)$$

$$\bar{d}_{i_1 \cdots i_4}^{(D_s)}(\ell) \;=\; \mathrm{Res}_{i_1 \cdots i_4}\left[\mathcal{A}_N^{(D_s)}(\ell) - \sum_{[j_1|j_5]} \frac{\bar{e}_{j_1 \cdots j_5}^{(D_s)}(\ell)}{d_{j_1} \cdots d_{j_5}}\right], \quad \ldots \quad (6)$$

Using the n_i vectors, loop momenta fulfilling the constraints can be generated: $\ell_{i_1 \cdots i_M} = V_{i_1 \cdots i_M} + \sum_{i=M}^{D} \alpha_i n_i$ where $\alpha_M^2 = -V_{i_1 \cdots i_M}^2 - \sum_{i=M+1}^{D} \alpha_i^2$. In general, the solutions are found as complex momenta. The vectors $V_{i_1 \cdots i_M}$ reside in the physical space; they are constructed from sums of external momenta as specified by the cuts, cf. [1]. By successively applying quintuple, quadruple, triple and double D_s-dimensional unitarity cuts, all coefficients can be determined. The cuts yield M onshell propagators factorizing the unintegrated one-loop amplitude into M tree-level amplitudes. Hence, the residues can be evaluated:

$$\mathrm{Res}_{i_1 \cdots i_M}\left[\mathcal{A}_N^{(D_s)}(\ell)\right] \;=\; \sum_{\{\lambda_1, \ldots, \lambda_M\} = 1}^{D_s - 2} \left\{\prod_{k=1}^{M} \mathcal{M}^{(0)}\left(\ell_{i_k}^{(\lambda_k)}; p_{i_k+1}, \ldots, p_{i_{k+1}}; -\ell_{i_{k+1}}^{(\lambda_{k+1})}\right)\right\} \quad (7)$$

where $\ell_{i_k} = \ell + q_{i_k} - q_{i_M}$ and the sum is over internal polarization states.

3 C++ implementation and results

The algorithm described above has been implemented in a new C++ code. The only interface is to link the QCDLoop package [11] for evaluating the master integrals. The program is ca-

pable of calculating the $A_N^{[1]}(\{p_i, \kappa_i\})$ amplitudes in double precision in the four-dimensional helicity scheme. This allows for crosschecks with the results obtained in Refs. [7, 12].

The construction of the orthonormal sets of the $D - M + 1$ basis vectors and the $D_s - 2$ polarization vectors follows the method outlined in [7]. In addition, the n_i vector generation $(i = M, \ldots, D)$ has been set up such that basis vectors obtained for large-M cuts can be re-used for suitable lower-M cuts. The basic strategy, which has been implemented to find the coefficients, is as follows: by using the freedom in choosing loop momenta, m solutions and, therefore, algebraic equations, such as eq. (4), can be generated to solve for $n \leq m$ coefficients.[a] First the dependence on D_s is eliminated by computing: $\mathrm{Res}_{i_1 \cdots i_M}[\mathcal{A}_N(\ell)] = (D_s - 3)\,\mathrm{Res}_{i_1 \cdots i_M}[\mathcal{A}_N^{(D_s)}(\ell)] - (D_s - 4)\,\mathrm{Res}_{i_1 \cdots i_M}[\mathcal{A}_N^{(D_s+1)}(\ell)]$. Then higher-point terms are subtracted yielding numerator factors $\bar{e}_{i_1 \cdots i_5}(\ell)$ etc. that are independent of D_s, i.e. eqs. (5) and (6) work without the (D_s) label. For the coefficients of the cut-constructible part, one can dispense with the determination of the D_s dependence of the residues and set $D = D_s = 4$. This, in addition, leads to smaller subsystems of equations, which can be solved separately, e.g. eq. (4) simplifies to $\bar{d}_{i_1 \cdots i_4}(\ell) = d^{(0)}_{i_1 \cdots i_4} + \alpha_4\, d^{(1)}_{i_1 \cdots i_4}$. The tree-level amplitudes needed to obtain the residues in eq. (7) are calculated with Berends–Giele recursion relations [13], adjusted to work for gluons in higher dimensions. For efficiency, currents, which involve external gluons only, are stored for re-use in evaluating other residues.

A number of consistency checks was carried out to verify the correctness of the implementation. The gauge invariance of the results and their independence of different choices for loop momenta and dimensionalities D and D_s were tested. Coefficients themselves, the pole structure of the amplitudes and, finally, the amplitudes themselves have been compared to analytic results for various N and different momentum and polarization configurations of the gluons. Agreement within the limits of double-precision calculations has been found with the numbers produced by Rocket for the fixed phase-space points given in [7].

In the following, studies are presented that have been conducted to examine the accuracy and time dependence of the numerical calculation.

Accuracy of the results. The quality of the numerical solutions can be estimated by analyzing the logarithmic relative deviations, which are defined as

$$\varepsilon_{\mathrm{dp,sp}} = \log_{10} \frac{|A_{N,\mathrm{C}++}^{[1](\mathrm{dp,sp})} - A_{N,\,\mathrm{anly}}^{[1](\mathrm{dp,sp})}|}{|A_{N,\,\mathrm{anly}}^{[1](\mathrm{dp,sp})}|}, \qquad \varepsilon_{\mathrm{fp}} = \log_{10} \frac{2\,|A_{N,\mathrm{C}++}^{[1](\mathrm{fp})}[1] - A_{N,\mathrm{C}++}^{[1](\mathrm{fp})}[2]|}{|A_{N,\mathrm{C}++}^{[1](\mathrm{fp})}[1]| + |A_{N,\mathrm{C}++}^{[1](\mathrm{fp})}[2]|}, \quad (8)$$

where the analytically known pole structures of the one-loop amplitudes are taken as reference for double (dp) and single (sp) poles, while for finite parts (fp), two independent solutions are compared with each other. Figure 1 shows the ε distributions together with the number of generated phase-space points for various N. The top row of numbers in the plots displays the means of the distributions. All results have been obtained for the same cuts on external gluons as reported in [7]; for the effect of tighter cuts, see Figure 3 (right).

For $N = 15$, the double-precision evaluation of the coefficients clearly is not sufficient to yield reliable finite-part result. The loss of precision as N increases is correlated with the more frequent appearance of small denominators and large numbers characteristic for the calculation. The two rightmost bottom panels of Figure 1 present two examples by depicting the range of magnitude taken by Gram determinants, used to evaluate the $V_{i_1 \cdots i_M}$ vectors of external gluons, and $e^{(0)}_{i_1 \cdots i_5}$ coefficients, which can be of $\mathcal{O}(10^{24})$ for $N = 15$. Double

---

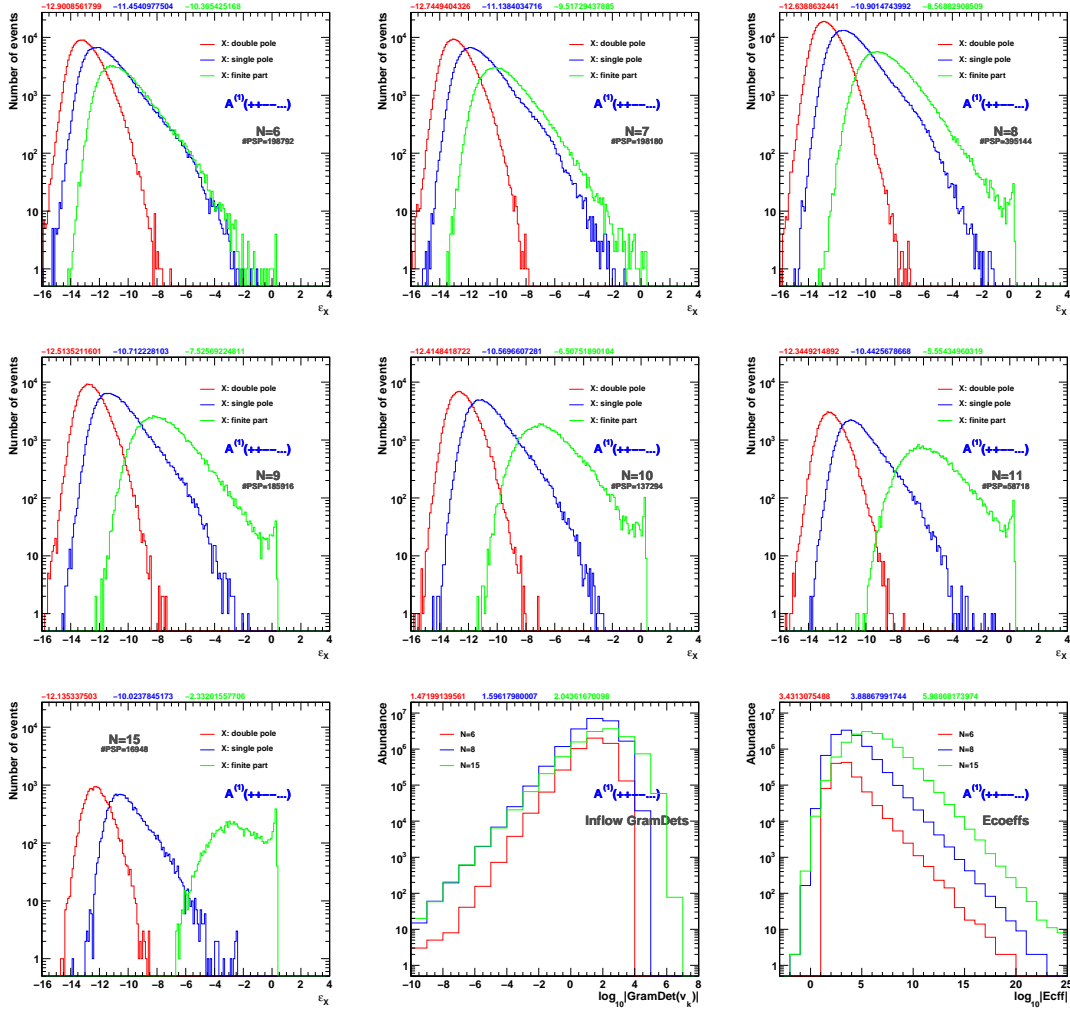[a]In principle, an infinite number of equations can be generated to fit a fixed number of unknowns.

Figure 1: Double-, single-pole and finite-part accuracy (in double precision) of the $++--\ldots$ one-loop amplitudes for $N = 6, \ldots, 11, 15$ gluons; see also text and right panel of Figure 3. Bottom row: center, double-logarithmic distributions of Gram determinants involving sets of external gluons and, right, $e_{i_1 \cdots i_5}^{(0)}$ coefficients for the $N = 6, 8, 15$ gluon setups.

precision will then be insufficient to make cancellations as they may occur e.g. in eq. (6) manifest. The scatter plots of Figure 2 visualize that (partial) correlations exist between the relative accuracy of the finite part and the smallest Gram determinant of external-gluon sets, the largest $e_{i_1 \cdots i_5}^{(0)}$ coefficient and the single-pole accuracy. In all cases, the areas of scatters shift with increasing N towards worse accuracy and more extreme values of Gram determinants and 5-point coefficients. The calculation may still involve other small denominators, such as the leftover d_j in the subtraction terms of e.g. eq. (6). This leads to
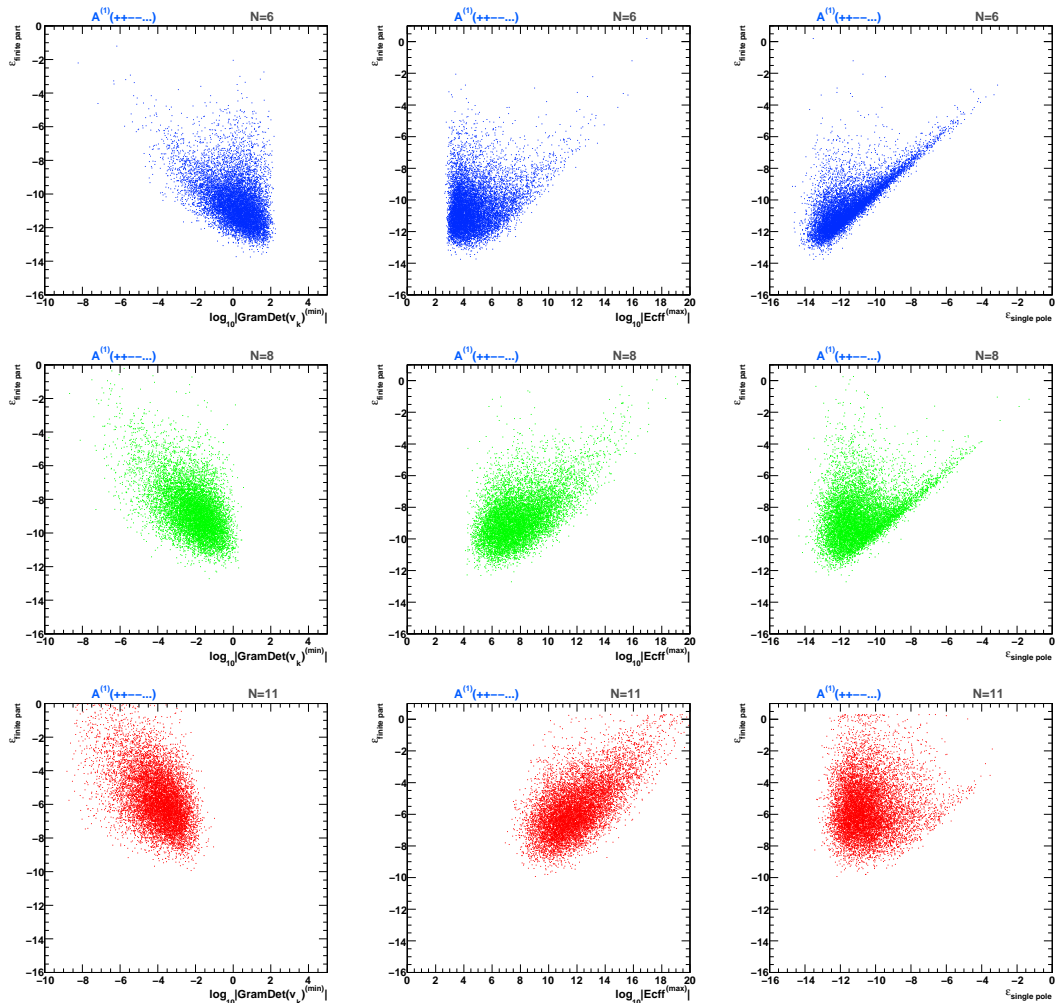
Figure 2: Finite-part accuracy versus minimal Gram determinant of external-gluon sets (left), maximal $e^{(0)}_{i_1 \cdots i_5}$ coefficient (center) and single-pole accuracy (all in double precision) for the $N = 6, 8, 11$ gluon setups of above.

instabilities (even for small coefficients) and uncorrelated areas in the plots are populated. Given these correlations, it can be seen that one way of achieving higher accuracy is to compute the coefficients in quadrupole precision. This has been pointed out in Ref. [7].

Efficiency of the calculation. As estimated in Ref. [4] (p. 31), the algorithm is expected to have polynomial complexity, see also [7]. The computing time τ_N to calculate an ordered N-gluon one-loop amplitude should scale as $N^x + \ldots$, with the leading term having $x = 9$ dominating the behaviour for large N. The results for τ_N with $N = 4, \ldots, 20$ are shown in Figure 3 together with the exponents $x_N = \ln \frac{\tau_N}{\tau_{N-1}} / \ln \frac{N}{N-1}$. The plots demonstrate that the C++ algorithm as implemented displays the predicted scaling.
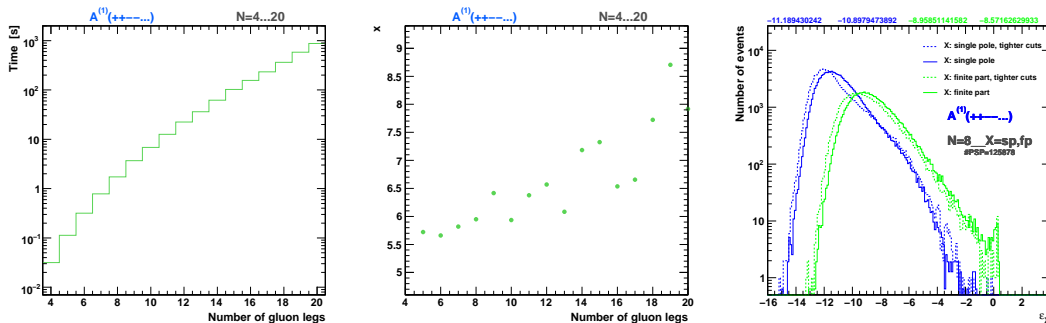
Figure 3: N dependence of the computing time; x_N-exponents (center), see text. Times refer to using a 2.20 GHz Intel Core2 Duo processor. Tighter gluon cuts were used: $|\eta_i| < 2$, $p_{\perp,i}\, s^{-0.5} > 0.1$, $R_{ij} > 0.7$, denoted as in [7]. The last plot shows the accuracy improvement.

4   Conclusions

It has been shown that the generalized unitarity method of Refs. [1, 2] can be implemented in a stable and fast C++ program. The one-loop N-gluon amplitude has been used as a testing ground. The next step is to integrate the C++ code in an existing leading-order generator. The resulting upgraded generator will be able to generate both virtual and bremsstrahlung contributions for arbitrarily complex Standard Model processes. The final step towards a full next-to-leading order Monte Carlo is to add the necessary phase-space integrations. As was shown in Ref. [14] the virtual matrix elements calculated with the generalized unitarity method can be used in next-to-leading order Monte Carlo programs.

References

[1] R.K. Ellis, W.T. Giele and Z. Kunszt, JHEP **0803**, 003 (2008) [arXiv:0708.2398 [hep-ph]].

[2] W.T. Giele, Z. Kunszt and K. Melnikov, JHEP **0804**, 049 (2008) [arXiv:0801.2237 [hep-ph]].

[3] Based on a presentation given by J. Winter:
http://ilcagenda.linearcollider.org/contributionDisplay.py?contribId=78&sessionId=18&confId=2628

[4] Z. Bern *et al.* [NLO Multileg Working Group], arXiv:0803.0494 [hep-ph].

[5] Z. Bern, L.J. Dixon, D.C. Dunbar and D.A. Kosower, Nucl. Phys. B **425**, 217 (1994) [arXiv:hep-ph/9403226]; Z. Bern, L.J. Dixon and D.A. Kosower, Nucl. Phys. B **513**, 3 (1998) [arXiv:hep-ph/9708239]; R. Britto, F. Cachazo and B. Feng, Phys. Rev. D **71**, 025012 (2005) [arXiv:hep-th/0410179]; R. Britto, F. Cachazo and B. Feng, Nucl. Phys. B **725**, 275 (2005) [arXiv:hep-th/0412103].

[6] G. Ossola, C.G. Papadopoulos and R. Pittau, Nucl. Phys. B **763**, 147 (2007) [arXiv:hep-ph/0609007].

[7] W.T. Giele and G. Zanderighi, JHEP **0806**, 038 (2008) [arXiv:0805.2152 [hep-ph]].

[8] G. Ossola, C.G. Papadopoulos and R. Pittau, JHEP **0803**, 042 (2008) [arXiv:0711.3596 [hep-ph]];
C.F. Berger *et al.*, Phys. Rev. D **78**, 036003 (2008) [arXiv:0803.4180 [hep-ph]].

[9] T. Gleisberg and S. Höche, JHEP **0812**, 039 (2008) [arXiv:0808.3674 [hep-ph]].

[10] Z. Bern and D.A. Kosower, Nucl. Phys. B **362**, 389 (1991).

[11] R.K. Ellis and G. Zanderighi, JHEP **0802**, 002 (2008) [arXiv:0712.1851 [hep-ph]].

[12] A. Lazopoulos, arXiv:0812.2998 [hep-ph].

[13] F.A. Berends and W.T. Giele, Nucl. Phys. B **306**, 759 (1988).

[14] R.K. Ellis, K. Melnikov and G. Zanderighi, arXiv:0901.4101 [hep-ph].