ELSEVIER

Available online at www.sciencedirect.com

ScienceDirect

Nuclear Physics B (Proc. Suppl.) 205–206 (2010) 86–91

NUCLEAR PHYSICS B
PROCEEDINGS
SUPPLEMENTS

www.elsevier.com/locate/npbps

Calculating one-loop multileg processes

Th. Diakonidis^a, B. Tausk^a

^aDeutsches Elektronen-Synchrotron DESY, Platanenallee 6, 15738 Zeuthen, Germany

This study is targeted to the calculation of NLO QCD corrections to multileg processes that are important for the LHC. Starting from the construction of Feynman diagrams, the analytical reduction of general one-loop integrals to scalar master ones, the calculation of colour structures, and finally phase space point selection are reviewed, in a program producing Fortran code for numerical calculation of one-loop corrections for processes like $gg \rightarrow t\bar{t}$, $gg \rightarrow t\bar{t}g$ and $gg \rightarrow t\bar{t}gg$.

1. Introduction

The era of LHC has finally arrived, after many years of waiting and anticipating, and all the particle physics community is waiting for the first results. There has been a lot of activity the last two years in the branch of NLO calculation. With almost all interesting cases of $2 \rightarrow 3$ processes having been calculated in QCD, as an example we can refer to the case of $pp \rightarrow t\bar{t} + jet$ [1], and a series of papers referring to W and Z production like $WW + jet$ [2–5], $ZZ + jet$ [6] and $WW + \gamma$ [7], the interest is moving towards $2 \rightarrow 4$. Processes like $pp \rightarrow t\bar{t} + b\bar{b}$ [8–10], $V + 3jet$ production [11–14], $qq \rightarrow b\bar{b}b\bar{b}$ [15] and most recently the case of $pp \rightarrow t\bar{t} + 2jets$ [16] have been studied in the last three years, providing the background for future discoveries. This study is also towards this direction.

2. The method

Our method is based on the Feynman diagrammatic approach. Which means that we calculate the total amplitude by taking into consideration the contributions of each of the one-loop Feynman diagrams that exist for the NLO calculation. The final program calculating the amplitude is in Fortran, although a variety of tools are used in combination (see Figure 5): Diana [17] for constructing the diagrams, a program based on Qgraf [18], Form [19] for manipulation of the Diana output (the steps of calculation in Form are coloured dark-grey - red), Maple for optimization of the produced Fortran code (medium-grey - green colour) and finally Fortran for the numerical evalua-

tion (light-grey - blue colour). In the next section, we will explain in more detail the complete calculation, which ends by providing the amplitude from an initial phase space point.

2.1. The steps of the calculation

As mentioned above, the first step is implemented by using Diana. We provide the program with the appropriate Feynman rules and ask for the construction of the diagrams. We move on gradually starting from the case of $gg \rightarrow t\bar{t}$, where we have 31 one-loop diagrams in total and stepping forward for $gg \rightarrow t\bar{t}g$ the total number is 354, and 4510 for the case of $gg \rightarrow t\bar{t}gg$, see Table 1.

Diana provides us with an output in Form which we further manipulate. The next step is to strip the one-loop amplitude from the colour structures. In a few words using Form to manipulate the output of Diana we strip off and manipulate the colour structures using colour algebra. We translate the Form result to Fortran code and end up with a Fortran array containing all the possible colour information. For the case of $gg \rightarrow t\bar{t}$ we only have 3 colour structures, divided in 2 groups, 8 cases for $gg \rightarrow t\bar{t} + g$ in 2 groups too, and finally for $gg \rightarrow t\bar{t}gg$ there are 50 colour structures, which can be further separated in 4 main colour groups. Representatives of each group for all the above processes are shown in Table 2. All other colour structures are produced from these ones through permutations of the external gluons.

The next step is the manipulation of the stripped amplitude. What is left after the colour bracketing is: the spinor line connecting the two external fermions $t\bar{t}$ in all possible ways and the rest are the tensor in-

Figure 1. Representative set of box diagrams for the case of $gg \rightarrow t\bar{t}$

Table 1

Tree and Loop diagrams for the cases of $gg \rightarrow t\bar{t}$, $gg \rightarrow t\bar{t} + g$ and $gg \rightarrow t\bar{t} + gg$.

Processes	$gg \rightarrow t\bar{t}$	$gg \rightarrow t\bar{t} + g$	$gg \rightarrow t\bar{t} + gg$
Tree diagrams	3	16	123
Tadpoles	1	13	169
Bubbles	8	96	1196
Triangles	16	143	1579
Boxes	6	78	978
Pentagons	–	24	468
Hexagons	–	–	120
Total (loop diagrams)	31	354	4510

tegrals and scalar products of external momenta produced by propagators outside the loop, and finally the polarization vectors of the gluons.

The spinor line contribution is calculated by Fortran. After manipulating it using all the possible contracting equations and simplifying the result by moving the appropriate momenta to the corners of the spinors and applying the Dirac equation, we end up with the simplest structure. Then we glue the two spinors left and right and calculate numerically the result in Fortran.

The tensor integrals are identified and flagged by Form. There is a Fortran program that calculates them according to the New Recursive Reduction Method of [20]. The program called OLOTIC (One LOP Tensor Integral Calculator) applies the formulas of the paper expressing the tensor integrals in terms of scalar master integrals. More explicitly, what Form does is to extract the tensor components from the initial structures and then calls OLOTIC. OLOTIC calculates those numerically, using the analytical structures of the Recursive Method above. In a few words, the idea of the reduction is that one-loop n -point rank R tensor Feynman integrals [in short: (n, R) -integrals]

for $n \leq 6$ with $R \leq n$ are represented in terms of $(n, R - 1)$ - and $(n - 1, R - 1)$ -integrals (see Figure 4). The recursive scheme is very convenient for explicit calculations and contains the complete calculational chain of tensor reduction, for both massive and massless propagators, and works with dimensional regularization. The general case of tensor integrals using dimensional regularization was treated in a series of papers [21–25], thereby allowing also for massless particles. The only necessary package to be added is one for the evaluation of 1-point to 4-point scalar integrals. These are calculated by QCDloop [26]. This is a very interesting property of the reduction, because using Fortran and switching off the relative scalar master integrals we have all the information that is needed to determine the coefficients of the scalar master integrals that arise in calculations by unitarity methods. In the same way, by switching off all the scalar master integrals we can also extract the rational term. Finally, due to the capability of QCDloop to give the $1/\epsilon$ and $1/\epsilon^2$ terms, we can cross check the correctness of our loop amplitude as these terms are proportional to the tree level amplitude.

The stripped amplitude in total is optimized by

Figure 2. Representative set of pentagon diagrams for the case of $gg \rightarrow t\bar{t} + g$

Figure 3. Representative set of hexagon diagrams for the case of $gg \rightarrow t\bar{t} + gg$

Maple. We divide the total number of diagrams to groups of hexagons, pentagons, boxes, triangles, and bubbles, and then we translate the output to Maple. Maple optimizes it and feeds it to a Fortran subroutine calculating the amplitude contributions of the above groups. As mentioned above, these structures are single coloured ones. So each one of them is multiplied with the appropriate colour structure stored in the colour subroutine.

Connecting all the pieces together, the stripped amplitudes including tensor integrals and spinor lines separately calculated in specific subroutines, the colour structures precalculated and stored in an array, we calculate the total loop amplitude. In the same

way, using much simpler routines, as there is no need of calling OLOTIC to calculate loop integrals and the numbers of diagrams are much smaller, we calculate the tree level case.

3. Conclusions and outlook

The biggest advantage of the program is the fact that it is flexible. Meaning that with minor changes, we can implement it for calculating different processes. We just have to provide Diana with the process that we are interested in, fix the Feynman rules and be careful with manipulation of possible spinor lines and colours. All the rest, tensor reduction and

Figure 4. Graphical presentation of the tensor reduction procedure

Figure 5. Graphical presentation of the case of $gg \rightarrow t\bar{t}gg$

Table 2

Color groups for the cases of $gg \rightarrow t\bar{t}$, $gg \rightarrow t\bar{t} + g$ and $gg \rightarrow t\bar{t} + gg$. The number of cases refers to all the possible permutations of the gluons

$gg \rightarrow t\bar{t}$	NC	$gg \rightarrow t\bar{t} + g$	NC	$gg \rightarrow t\bar{t} + gg$	NC
$(T^{\alpha_1} T^{\alpha_2})_{f_1 f_2}$	2	$(T^{\alpha_1} T^{\alpha_2} T^{\alpha_3})_{f_1 f_2}$	6	$(T^{\alpha_1} T^{\alpha_2} T^{\alpha_3} T^{\alpha_4})_{f_1 f_2}$	24
$Tr\{T^{\alpha_1} T^{\alpha_2}\} \delta_{f_1 f_2}$	1	$Tr\{T^{\alpha_1} T^{\alpha_2} T^{\alpha_3}\} \delta_{f_1 f_2}$	2	$Tr\{T^{\alpha_1} T^{\alpha_2}\} (T^{\alpha_3} T^{\alpha_4})_{f_1 f_2}$	12
–	–	–	–	$Tr\{T^{\alpha_1} T^{\alpha_2} T^{\alpha_3}\} (T^{\alpha_4})_{f_1 f_2}$	8
–	–	–	–	$Tr\{T^{\alpha_1} T^{\alpha_2} T^{\alpha_3} T^{\alpha_4}\} \delta_{f_1 f_2}$	6

optimization of the code are done automatically by the program. Further optimizing the code in spinor structures, caching the results of tensor integrals and extending OLOTIC for the cases of very small non vanishing Gram determinants so as to make accurate predictions for difficult phase space points are some of the future plans, applying of course the core of the program to other related processes.

4. Acknowledgments

Work supported in part by Sonderforschungsbereich/Transregio SFB/TR9 of DFG “Computergetrtztzte Theoretische Teilchenphysik” and by the European Community’s Marie-Curie Research Training Network MRTN-CT-2006-035505 “HEPTOOLS”. We would also like to thank our collaborators Jochem Fleischer and Tord Riemann for useful discussions.

REFERENCES

1. S. Dittmaier, P. Uwer and S. Weinzierl, *Nuovo Cim.* **123B** (2008) 1028.
2. S. Dittmaier, S. Kallweit, P. Uwer, *Phys. Rev. Lett.* **100** (2008) 062003 [arXiv:0710.1577 [hep-ph]].
3. J. M. Campbell, R. Keith Ellis, G. Zanderighi, *JHEP* **0712** (2007) 056 [arXiv:0710.1832 [hep-ph]].
4. S. Dittmaier, S. Kallweit, P. Uwer, *Nucl. Phys.* **B826** (2010) 18–70. [arXiv:0908.4124 [hep-ph]].
5. S. Dittmaier, S. Kallweit and P. Uwer, arXiv:1001.2427 [hep-ph].
6. T. Binoth, T. Gleisberg, S. Karg, N. Kauer and G. Sanguinetti, *Phys. Lett. B* **683** (2010) 154 [arXiv:0911.3181 [hep-ph]].
7. G. Bozzi, F. Campanario, V. Hankele and D. Zeppenfeld, *Phys. Rev. D* **81** (2010) 094030 [arXiv:0911.0438 [hep-ph]].
8. A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorini, *Phys. Rev. Lett.* **103** (2009) 012002 [arXiv:0905.0110 [hep-ph]].
9. A. Bredenstein, A. Denner, S. Dittmaier and S. Pozzorini, *JHEP* **1003** (2010) 021 [arXiv:1001.4006 [hep-ph]].
10. G. Bevilacqua, M. Czakon, C. G. Papadopoulos, R. Pittau and M. Worek, *JHEP* **0909** (2009) 109 [arXiv:0907.4723 [hep-ph]].
11. C. F. Berger, Z. Bern, L. J. Dixon *et al.*, *Phys. Rev. Lett.* **102** (2009) 222001 [arXiv:0902.2760 [hep-ph]].
12. R. Keith Ellis, K. Melnikov and G. Zanderighi, *Phys. Rev. D* **80** (2009) 094002 [arXiv:0906.1445 [hep-ph]].
13. C. F. Berger, Z. Bern, L. J. Dixon *et al.*, *Phys. Rev.* **D80** (2009) 074036. [arXiv:0907.1984 [hep-ph]].
14. C. F. Berger *et al.*, arXiv:1004.1659 [hep-ph]
15. T. Binoth, N. Greiner, A. Guffanti, J. Reuter, J. P. Guillet and T. Reiter, *Phys. Lett. B* **685** (2010) 293 [arXiv:0910.4379 [hep-ph]].
16. G. Bevilacqua, M. Czakon, C. G. Papadopoulos and M. Worek, arXiv:1002.4009 [hep-ph].
17. M. Tentyukov and J. Fleischer, *Comput. Phys. Commun.* **132** (2000) 124 [arXiv:hep-ph/9904258].
18. P. Nogueira, *J. Comput. Phys.* **105** (1993) 279–289.
19. J. A. M. Vermaseren, arXiv:math-ph/0010025.
20. T. Diakonidis, J. Fleischer, T. Riemann and J. B. Tausk, *Phys. Lett. B* **683** (2010) 69 [arXiv:0907.2115 [hep-ph]].
21. Z. Bern, L. J. Dixon and D. A. Kosower, *Phys. Lett. B* **302**, 299 (1993) [Erratum-ibid. **B 318**, 649 (1993)] [arXiv:hep-ph/9212308].
22. Z. Bern, L. J. Dixon and D. A. Kosower, *Nucl. Phys. B* **412**, 751 (1994) [arXiv:hep-

- ph/9306240].
23. T. Binoth, J. P. Guillet and G. Heinrich, Nucl. Phys. B **572**, 361 (2000) [arXiv:hep-ph/9911342].
 24. G. Duplancic and B. Nizic, Eur. Phys. J. C **35**, 105 (2004) [arXiv:hep-ph/0303184].
 25. T. Diakonidis, J. Fleischer, J. Gluza *et al.*, Phys. Rev. **D80** (2009) 036003 [arXiv:0812.2134 [hep-ph]].
 26. R. K. Ellis, G. Zanderighi, JHEP **0802** (2008) 002 [arXiv:0712.1851 [hep-ph]].