Preprint typeset in JHEP style - HYPER VERSION

CP3-08-39
ZU-TH 13/08

MadDipole: Automation of the Dipole Subtraction Method in MadGraph/MadEvent

R. Frederix

Center for Particle Physics and Phenomenology (CP3),
Université Catholique de Louvain,
Chemin du Cyclotron 2, 1348 Louvain-la-Neuve, Belgium

T. Gehrmann, N. Greiner

Institut für Theoretische Physik,
Universität Zürich,
Winterthurerstrasse 190, 8057 Zürich, Switzerland

Abstract: We present the implementation of the dipole subtraction formalism for the real radiation contributions to any next-to-leading order QCD process in the MadGraph/MadEvent framework. Both massless and massive dipoles are considered. Starting from a specific $(n+1)$-particle process the package provides a Fortran code for all possible dipoles to all Born processes that constitute the subtraction term to the $(n+1)$-particle process. The output files are given in the usual "MadGraph StandAlone" style using helicity amplitudes.

Contents

1. Introduction

Physics studies at the upcoming CERN LHC collider will frequently involve multi-particle final states. Especially searches for physics beyond the standard model rely on the reconstruction of new particles from their decay products, often through decay chains. Equally, requiring accompanying particles in the final state may serve to improve the ratio of signal to background processes, as done for example in the Higgs boson search through the vector boson fusion channel. Meaningful searches for these signals require not only a very good anticipation of the expected signal, but also of all standard model backgrounds yielding identical final state signatures. From the theoretical point of view, high precision implies that one has to go beyond the leading order in perturbation theory to be able to keep up with the precision of the measurements.

For leading order processes there have been many developments concerning event generation and simulation tools in the last two decades such as MadGraph/MadEvent [1–3] CompHEP/CalcHEP [4]/ [5], SHERPA [6] and WHIZARD [7] and also programs using different approaches such as ALPGEN [8] and HELAC [9]. All these programs are multi-purpose event generator tools, which are able to compute any process (up to technical restrictions in the multiplicity) within the standard model, or within alternative theories specified by their interaction Lagrangian or Feynman rules. They usually provide event information which can be interfaced into parton shower, hadronization and/or detector simulation.

Next-to-leading order (NLO) calculations are at present performed on a process-by-process basis. The widely-used programs MCFM [10,11], NLOJET++ [12], MC@NLO [13,14] and programs based on the POWHEG method [15–20] collect a variety of different processes

in a standardized framework, the latter two methods also match the NLO calculation onto a parton shower.

The NLO QCD corrections to a given process with a n-parton final state receive two types of contributions: the one-loop virtual correction to the $(2 \to n)$-parton scattering process, and the real emission correction from all possible $(2 \to n+1)$-parton scattering processes. For the numerical evaluation, one has to be able to compute both types of contributions separately.

The computation of one-loop corrections to multi-particle scattering amplitudes was performed on a case-by-case basis up-to-now, the calculational complexity increased considerably with increasing number of external partons. Since only a limited number of one-loop integrals can appear [21, 22] in the final result, the calculation of one-loop corrections can be reformulated as determination of the coefficients of these basis integrals, plus potential rational terms. Enormous progress [23–42] has been made in the recent past in the systematic determination of the one-loop integral coefficients and rational terms, and steps towards fully automated programs for the calculation of one-loop multi-parton amplitudes were made with the packages CutTools [43], BlackHat [44], Rocket [45] and GOLEM [46].

The real emission corrections contain soft and collinear singularities, which become explicit only after integration over the appropriate real radiation phase space yielding a hard n-parton final state. They are canceled by the singularities from the virtual one-loop contributions, thus yielding a finite NLO correction. To systematically extract the real radiation singularities from arbitrary processes, a variety of methods, based either on phase-space slicing [47] or on the introduction of process-independent subtraction terms [48] have been proposed. Several different algorithms to derive subtraction terms are available: residue subtraction [49], dipole subtraction [50, 51] and antenna subtraction [52–55].

Especially the dipole subtraction formalism, which provides local subtraction terms for all possible initial and final state configurations [50] and allows to account for radiation off massive partons [51], is used very widely in NLO calculations. It has recently also been automated in the SHERPA framework [56] and the TeVJet framework [57], and most recently in the form of independent libraries [58] interfaced to MadGraph. The dipole subtraction within the SHERPA framework is not available as a stand-alone tool, while within the TeVJet framework, the user needs to provide all the necessary process dependent information. Moreover both approaches have only included massless particles for the dipoles. There is still no general tool available which is able to produce the dipole terms for an arbitrary process and which can also deal with massive partons.

In this paper we present MadDipole, an automatic generation of the dipole subtraction terms using the MadGraph/MadEvent framework. The results are `Fortran` subroutines which return the squared amplitude for all possible dipole configurations in the usual MadGraph style. We describe the construction of the dipole subtraction terms and their implementation in Section 2. Results from various checks of the implementation are provided Section 3, and instructions on the practical usage of the package are contained in Section 4.

2. Construction of dipole terms

The fundamental building blocks of the subtraction terms in the dipole formalism [50, 51] are dipole splitting functions $\mathbf{V}_{ij,k}$, which involve only three partons: emitter i, unresolved parton j, spectator k. A dipole splitting function accounts for the collinear limit of j with i, and for part of the soft limit of j in between i and k. The dipole factors, which constitute the subtraction terms, are obtained by multiplication with reduced matrix elements, where partons i, j and k are replaced by recombined pseudo-partons $\widetilde{ij}$, $\widetilde{k}$. The full soft behavior is recovered after summing all dipole factors.
Throughout the whole paper we are using the notation introduced in Refs. [50] and [51]. Independent on whether we have initial or final state particles we can write an arbitrary dipole in the form

$$\mathcal{D}_{ij,k} \quad \sim \quad {}_m\langle 1,...\tilde{ij},...,\tilde{k},...,m+1|\frac{\mathbf{T}_k\cdot\mathbf{T}_{ij}}{\mathbf{T}_{ij}^2}\mathbf{V}_{ij,k}|1,...\tilde{ij},...,\tilde{k},...,m+1\rangle_m. \tag{2.1}$$

The amplitude factors $\langle\dots|$ ('bra') and $|\dots\rangle$ ('ket') on the right hand side are tensors in color space. The helicities of the external particles in them are a priori fixed (but can be summed over for unpolarized processes), while the helicities of the pseudo-partons have to be summed over after contraction with the dipole splitting function.
These Born-level amplitude factors are provided by the usual MadGraph code. The two elements that combine the ket with the bra are the additional color structure $\frac{\mathbf{T}_k\cdot\mathbf{T}_{ij}}{\mathbf{T}_{ij}^2}$ and the dipole splitting function $\mathbf{V}_{ij,k}$.

2.1 Color and helicity management

For the calculation of the color factors there already exist routines in the MadGraph program. Our intension was to use exactly these routines because this code is very well-confirmed and efficient. We have included the additional color operator, $\mathbf{T}_k\cdot\mathbf{T}_{ij}$, by rewriting the internal MadGraph color labelling for the ket-side only. After insertion of this color operator the color structure is no longer multiplied by its own complex conjugate and therefore the routine that squares the color needed to be altered, to multiply the modified ket by its original complex conjugate. We emphasize that due to the factorial growth of the color factors MadGraph can not handle more than seven colored particles.

For the insertion of the splitting function $\mathbf{V}_{ij,k}$ several changes with respect to the original code are required. One has to keep in mind that in general the splitting function is a tensor in helicity space, *i.e.*,

$$\mathbf{V}_{ij,k} \equiv \langle\mu|\mathbf{V}_{ij,k}|\nu\rangle = \mathbf{V}_{ij,k}^{\mu\nu}.$$

As MadGraph deals with helicity amplitudes, we have to write the dipole in a slightly different way to be able to include the calculation of the splitting function in the code. Neglecting the color for a moment we start from the definition of the dipole in (2.1) and

by inserting a full set of helicity states $-g_{\mu\nu} = \sum_\lambda \epsilon^*_\mu(\lambda)\epsilon_\nu(\lambda)$ we get

$$\begin{aligned}
\mathcal{D}_{ij,k} &\sim {}_m\langle 1,...\tilde{ij},...,\tilde{k},...,m+1|_\mu \mathbf{V}^{\mu\nu}_{ij,k}\ {}_\nu|1,...\tilde{ij},...,\tilde{k},...,m+1\rangle_m \qquad (2.2)\\
&= {}_m\langle 1,...\tilde{ij},...,\tilde{k},...,m+1|_{\mu'} \left(-g^{\mu'}_{\ \mu}\right) \mathbf{V}^{\mu\nu}_{ij,k} \left(-g_\nu^{\ \nu'}\right) {}_{\nu'}|1,...\tilde{ij},...,\tilde{k},...,m+1\rangle_m \\
&= \sum_{\lambda_a,\lambda_b} {}_m\langle \dots|_{\mu'}\ \epsilon^{*\mu'}(\lambda_b)\epsilon_\mu(\lambda_b)\mathbf{V}^{\mu\nu}_{ij,k}\ \epsilon^*_\nu(\lambda_a)\epsilon^{\nu'}(\lambda_a)\ {}_{\nu'}|\dots\rangle_m \\
&= \sum_{\lambda_a,\lambda_b} {}_m\langle \dots|_{\lambda_b}\ V(\lambda_b,\lambda_a)\ {}_{\lambda_a}|\dots\rangle_m
\end{aligned}$$

with $V(\lambda_b,\lambda_a) = \epsilon_\mu(\lambda_b)\ \mathbf{V}^{\mu\nu}_{ij,k}\ \epsilon^*_\nu(\lambda_a)$ and $\epsilon^\mu(\lambda)\,{}_\mu|\dots\rangle_m = {}_\lambda|\dots\rangle_m$.
The polarization vectors are calculated using the HELAS routines [59] already available in the MadGraph code. The parts that are diagonal in helicity space are trivial to calculate in that sense that one only has to multiply the MadGraph output for the squared amplitude for a given helicity combination with the splitting function. To calculate the off-diagonal helicity terms, the amplitude for each helicity combination is stored and then combined with the according amplitude with opposite helicity.
For the calculation of the splitting functions and for the remaping of the momenta we use modified versions of the routines used in MCFM [10, 11].

2.2 Massive particles

If some of the masses of the external particles are non-zero, in particular for processes involving top and/or bottom quarks, there are dipoles for which the unresolved parton is massive. In this case the collinear singularities are regulated by the mass of the unresolved parton and the unsubtracted matrix element does therefore no longer diverge in these collinear limits, but only develops potentially large logarithms. Our code still calculates all possible dipoles, also in which the unresolved parton is massive, but puts them in a separate subroutine, `dipolsumfinite(...)`, that is not evaluated by default. In the limit of large center of momentum energy or, similarly, small external masses, the user can easily include the non-divergent dipoles to subtract the associated large logarithms, which can then be included analytically through the integrated subtraction terms. In the limit of zero external masses we have checked that the results obtained after summing all dipoles are the same as obtained by generating the code with massless particles from the start.

2.3 Phase space restriction

The calculation of the subtraction terms is only necessary in the vicinity of a soft and/or collinear limit. Away from these limits the amplitude is finite and there is in principle no need to calculate the computationally heavy subtraction terms. The distinction between regions near to a singularity from regions without need for a subtraction can be parameterized by a parameter usually labelled α with $\alpha \in [0, 1]$, which was introduced in Ref. [60] for processes involving partons only in the final state. The case with incoming hadrons, *i.e.*, with partons in the initial state, is described in Ref. [12].

Using the notation of Ref. [12], the contribution from the subtraction term to the differential cross section can be written as

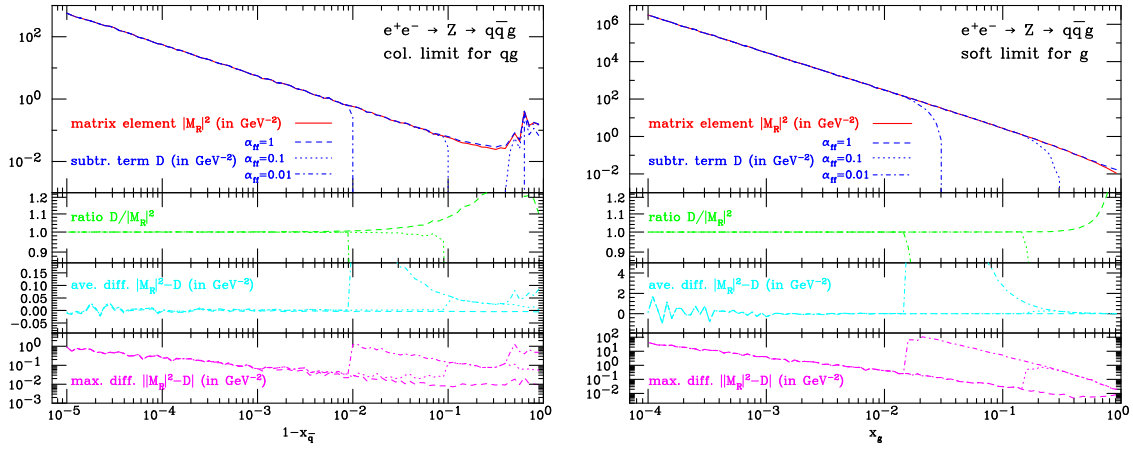
$$
\begin{aligned}
d\sigma^A_{ab} = & \sum_{\{n+1\}} d\Gamma^{(n+1)}(p_a,p_b,p_1,...,p_n+1)\frac{1}{S_{\{n+1\}}} \\
& \times\Bigg\{ \sum_{\substack{\text{pairs}\\ i,j}} \sum_{k\neq i,j} \mathcal{D}_{ij,k}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(p_a,p_b,p_1,..,\tilde{p}_{ij},\tilde{p}_k,..,p_{n+1})\Theta(y_{ij,k}<\alpha) \\
& \quad + \sum_{\substack{\text{pairs}\\ i,j}} \Big[\mathcal{D}^a_{ij}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,p_1,..,\tilde{p}_{ij},..,p_{n+1})\Theta(1-x_{ij,a}<\alpha) \\
& \qquad\qquad + (a\leftrightarrow b)\Big] \\
& \quad + \sum_{i\neq k} \Big[\mathcal{D}^{ai}_{k}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,p_1,..,\tilde{p}_{k},..,p_{n+1})\Theta(u_i<\alpha) + (a\leftrightarrow b)\Big] \\
& \quad + \sum_{i} \Big[\mathcal{D}^{ai,b}(p_a,p_b,p_1,...,p_{n+1}) F_J^{(n)}(\tilde{p}_a,p_b,\tilde{p}_1,...,\tilde{p}_{n+1})\Theta(\tilde{v}_i<\alpha) + (a\leftrightarrow b)\Big] \Bigg\} \ .
\end{aligned}
\tag{2.3}
$$

The functions $\mathcal{D}_{ij,k}$, $\mathcal{D}^a_{ij}$, $\mathcal{D}^{ai}_k$ and $\mathcal{D}^{ai,b}$ are the dipole terms for the various combinations for emitter and spectator. $\sum_{\{n+1\}}$ denotes the summation over all possible configurations for this $(n+1)$-particle phase space which is labelled as $d\Gamma^{(n+1)}$ and the factor $S_{\{n+1\}}$ is the symmetry factor for identical particles. We have introduced four different α-parameters, one for each type of dipoles. In our code they are called `alpha_ff`, `alpha_fi`, `alpha_if` and `alpha_ii` for the final-final, finial-initial, initial-final and initial-initial dipoles, respectively. The actual values for these parameters are by default set to unity, corresponding to the original formulation of the dipole subtraction method [50, 51], but can be changed by the user in the file `dipolsum.f`.

It has to be kept in mind that the integrated dipole factors, which are to be added with the virtual n-parton contribution, will also depend on α. For case of massless partons, the α-dependence of the integrated terms is stated in [12, 60] while for massive partons results for most cases can be found in [61, 62].

3. Checks

The MadDipole package provides a code, `check_dip.f`, which allows the user to test the limits of the $(n+1)$-particle matrix element and the dipole subtraction terms. This code builds up a trajectory of randomly selected phase space point approaching a given soft or collinear limit of the $(n+1)$-parton matrix element and yields the values of matrix element, sum of all the dipoles, and their ratio along this trajectory. The result is printed to the screen in a small table for which each successive row is closer to the singularity. The ratio between matrix element and the sum of the dipoles should go to unity. We have tested our code in all possible limits, both for massless as well as massive dipoles and found no



(a) (b)

Figure 1: Matrix element squared $|M_R|^2$ (upper plots, solid line) and the subtraction terms D (upper plots, dashed/dotted/dot-dashed lines) for the process $e^+(p_1)e^-(p_2) \to Z \to q(p_3)\bar{q}(p_4)g(p_5)$ as a function of $1 - x_{\bar{q}} = (p_3.p_5)/(p_1.p_2)$ and $x_g = 1 - (p_3.p_4)/(p_1.p_2)$ in figures (a) and (b), respectively. Also plotted are the ratio $D/|M_R|^2$, the difference $|M_R|^2 - D$ (averaged over 100 random points per bin) and the maximal difference $\max(||M_R|^2 - D|)$ per bin. The dashed lines include the dipoles for each point in phase space, $\alpha_{ff} = 1$, while for the dotted $\alpha_{ff} = 0.1$ and dot-dashed $\alpha_{ff} = 0.01$ the phase space for the dipoles has been restricted to the collinear/soft regions.

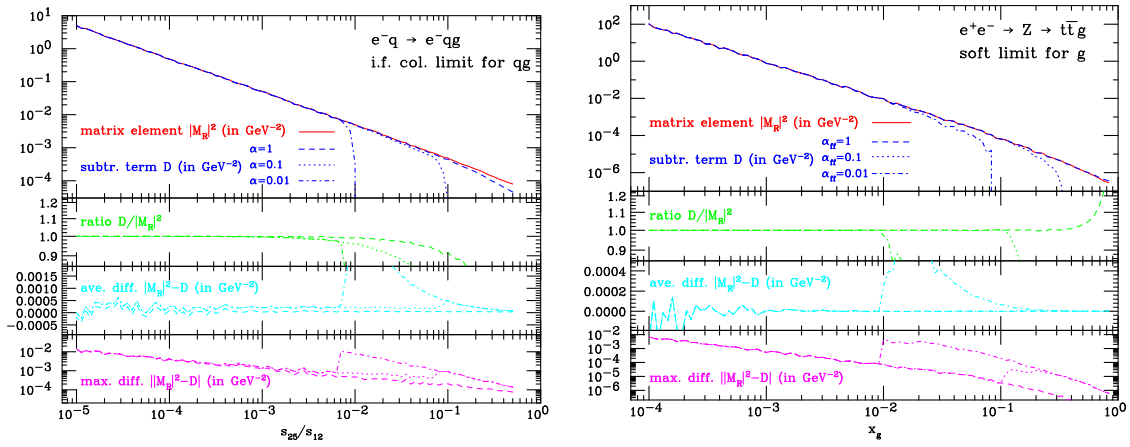
inconsistencies. Choosing small values for α-parameters, *e.g.*, $\alpha = 0.1$, improves the computation time and the convergence of the subtraction procedure.

To show that the subtraction terms are implemented correctly we provide a couple of examples in the form of plots and argue that the cancellation between the matrix element squared and the subtraction term is as expected. In the figures 1 and 2 we show the matrix element squared and the subtraction term as a function of a variable that represents a soft and/or collinear limit of the process specified. For these figures we have binned the x-axis (equally sized bins for the logarithmic scale) and generated random points in phase space to fill each of the bins with exactly 100 events. In the upper plot, $|M_R|^2$ and D are the per bin averages of the matrix element squared and the subtraction term, respectively. The second to upper plot shows the per bin average of the ratio of the matrix element squared and the subtraction term, while the third plot from the top shows the per bin average of the difference. The lowest plot shows the absolute value of the maximal difference among the 100 points in a bin. To show the effects of the phase space restriction for the dipoles, see section 2.3, all the plots are given for $\alpha = 1$ (dashed lines), $\alpha = 0.1$ (dotted lines) and $\alpha = 0.01$ (dot-dashed lines).

In figure 1(a) we show the matrix element squared and the subtraction term as a function of $1-x_{\bar{q}}$, where $x_{\bar{q}}$ is the fraction of the energy carried by the anti-quark, $x_{\bar{q}} = \frac{s_{34}+s_{45}}{s_{12}} = 1-\frac{s_{35}}{s_{12}}$, with $s_{ij} = p_i.p_j$. For this process, $e^+(p_1)e^-(p_2) \to Z \to q(p_3)\bar{q}(p_4)g(p_5)$, there are only final-final state dipoles contributing to the subtraction term. The center of mass energy is set equal to the Z boson mass $\sqrt{s} = m_Z$. To restrict the discussion to the collinear

divergence only, points close to the soft divergence ($x_{\bar{q}} = x_{\bar{q}} = 1$) have been removed by forcing $x_q + x_{\bar{q}} < 1.5$ in the generation of the phase space points.
From the upper plot it is clear that both the matrix element squared and the subtraction term diverge in the collinear limit $x_{\bar{q}} \to 1$, as $1/x_{\bar{q}}$. The ratio $D/|M_R|^2$ goes to 1 and the average values of the differences fluctuate close to 0 as can be seen in the second and third plots from the top. The numerical fluctuations for small $1-x_{\bar{q}}$ can be completely explained by statistical fluctuations. They are of the order of 1% of the maximal difference given in the lower plot. As can be expected, the cancellations are not exact, which is shown by the lower plot. The maximal difference between $|M_R|^2$ and D rises like $1/\sqrt{1-x_{\bar{q}}}$, which does not lead to a divergent phase space integral, because the integration measure is proportional to $x_{\bar{q}}$. The small peaks/fluctuations in the region for small $x_{\bar{q}}$ are due to the fact that we are approaching the other collinear limit, *i.e.*, for which the gluon is collinear to the anti-quark $x_q \to 1$, where the matrix elements squared and the subtraction term also diverge.
In figure 1(b) the same matrix elements and subtraction terms are presented, but as a function of the fraction of the energy carried away by the gluon $x_g = 2 - x_q - x_{\bar{q}}$. The limit for which x_g goes to zero represents the soft divergence of this process, while the collinear divergences for this process are removed by excluding phase space points for which $x_{\bar{q}} > (1 + x_q)/2$ or $x_q > (1 + x_{\bar{q}})/2$. The same conclusion as for figure 1(a) can be drawn here: the matrix element squared and the subtraction term diverge in the soft limit, their ratio goes to one and the average difference to zero, while the absolute value of the maximal difference still rises when approaching the soft limit, but does not lead to a divergent phase space integral.

An example for a collinear limit between final state and initial state particles is given in figure 2(a). In this plot the matrix element squared and the subtraction term for the process $e^-(p_1)q(p_2) \to e^-(p_3)q(p_4)g(p_5)$ are given as a function of the invariant mass of the initial state quark and the final state gluon s_{25}/s_{12}. As this invariant mass goes to zero, the matrix element squared and the subtraction term diverge like $1/s_{25}$, and their ratio goes to one. To remove the other possible divergences a cut on the momentum transfered $s_{13}/s_{12} > 0.5$ and on the invariant mass of the final state quark and gluon $s_{45}/s_{12} > 0.2$ have been imposed. Like for the final-final state dipoles the average difference goes to a constant, as can be seen from the second plot from the bottom, but the dipoles have a sizeable constant contribution. Therefore the normalization of the average value for the difference $|M_R|^2 - D$ depends on the number of dipoles included for the phase space point. If all the dipoles are included for all points the difference goes to a smaller constant than if we restrict the phase space of the subtraction term to be close to the singularities by setting $\alpha < 1$. Due to this restriction only the dipoles to cancel that divergence are included in the subtraction term and therefore give a smaller constant contribution, hence the difference $|M_R|^2 - D$ is larger. Also here the maximal left-over difference, the lowest plot, increases for small invariant masses but does not lead to a divergent phase space integral.
In figure 2(b) an example with massive final state particles is shown. The process is $t\bar{t}$ production at a linear collider, $e^+(p_1)e^-(p_2) \to t(p_3)\bar{t}(p_4)g(p_5)$ at 1 TeV center of mass energy. The plot shows a behavior very similar to the massless case, fig. 1, and the conclu-



(a) (b)

Figure 2: Matrix element squared $|M_R|^2$ (upper plots, solid line) and the subtraction term D (upper plots, dashed/dotted/dot-dashed lines) for (a) the process $e^-(p_1)q(p_2) \to e^-(p_3)q(p_4)g(p_5)$ as a function of $s_{25}/s_{12} = (p_2.p_5)/(p_1.p_2)$ and (b) the process $e^+(p_1)e^-(p_2) \to Z \to t(p_3)\bar{t}(p_4)g(p_5)$ as a function of $x_g = 1 - (p_3.p_4)/(p_1.p_2)$. Also plotted are the ratio $D/|M_R|^2$, the difference $|M_R|^2 - D$ (averaged over 100 random points per bin) and the maximal difference $\max(||M_R|^2 - D|)$ per bin. The dashed lines include the dipoles for each point in phase space, $\alpha = 1$, while for the dotted $\alpha = 0.1$ and dot-dashed $\alpha = 0.01$ the phase space for the dipoles has been restricted to the collinear/soft regions.

sions drawn there apply to this plot as well.

As a further check we have tested the code extensively against MCFM [10, 11]. We have generated random points in phase space and compared the subtraction terms calculated by MCFM with the subtraction terms calculated by our code. See table 1 for a list of processes that have been checked. We observed differences only in the case where dipoles were introduced entirely to cancel collinear limits, which can be made independently of the spectator particle. In our code all possible dipoles are calculated, which implies a sum over all spectator particles. However, if there is only a collinear divergence, *i.e.*, the unresolved parton cannot go soft, this sum is redundant and one dipole with the appropriate coefficient is enough to cancel the singularity. In MCFM, these special limits are implemented using a single spectator momentum, while MadDipole sums over all spectator momenta, thereby yielding a different subtraction term. We have checked in the relevant cases that close to the singularities the MCFM subtraction terms behave identical to the subtraction terms calculated by our code.

We also tested the CPU time which is needed to produce the squared matrix element and the dipoles for a given phase space point. These checks were performed with an Intel Pentium 4 processor with 3.20Ghz. As an example we picked out three different processes:

1) $gg \to gggg$: $|\mathcal{M}|^2$: 26ms, $\sum_{dipoles}$: 68ms

2) $u\bar{u} \to d\bar{d}ggg$: $|\mathcal{M}|^2$: 10ms, $\sum_{dipoles}$: 45ms

3) $u\bar{u} \to u\bar{u}ggg$: $|\mathcal{M}|^2$: 34ms, $\sum_{dipoles}$: 0.15s

process	subprocesses
Drell-Yan (W)	$q\bar{q}' \to W^+(\to e^+\nu_e)g$ $qg \to W^+(\to e^+\nu_e)q'$
Drell-Yan (Z)	$q\bar{q} \to Z(\to e^+e^-)g$ $qg \to Z(\to e^+e^-)q$
Drell-Yan (Z+jet)	$q\bar{q} \to Z(\to e^+e^-)q'\bar{q}'$ $q\bar{q} \to Z(\to e^+e^-)q\bar{q}$ $q\bar{q} \to Z(\to e^+e^-)gg$ $q\bar{g} \to Z(\to e^+e^-)qg$ $g\bar{g} \to Z(\to e^+e^-)q\bar{q}$
top quark pair ($t\bar{t}$)	$q\bar{q} \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)g$ $qg \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)q$ $gg \to t(\to bl^+\nu_l)\bar{t}(\to \bar{b}l^-\bar{\nu}_l)g$
t-channel single top with massive b-quark [63]	$gg \to t\bar{b}q\bar{q}'$ $qq' \to t\bar{b}q'q''$ $qq' \to t\bar{b}q'q''$ $qg \to t\bar{b}q'g$

Table 1: Set of processes for which the MadDipole code has been tested against MCFM for random points in phase space. All the possible initial-initial, initial-final and final-initial, dipoles for massless and massive final state particles have been checked with this set of subprocesses. No inconsistencies were found.

The time which is needed to produce the `Fortran` code is strongly dependent on the process and ranges from a few seconds to at most a few minutes. The process $gg \to 5g$ is currently not yet feasible within MadGraph because of the size of the color factors. Once MadGraph has been adjusted to handle this process, it will equally become accessible for MadDipole.

4. How to use the package

The installation and running of the MadDipole package is very similar to the usual Stand-Alone version of the MadGraph code. In this section, we only provide a brief description for MadDipole, while more information can also be found on the MadGraph wiki page, `http://cp3wks05.fynu.ucl.ac.be/twiki/bin/view/Software/MadDipole`.

1. Download and extract the MadDipole package, `MG_ME_DIP_V4.4.3.tar.gz`, from one of the MadGraph websites, *e.g.*, `http://madgraph.hep.uiuc.edu/`.

2. Run `make` in the `MadGraphII` directory.

3. Copy the `Template` directory into a new directory, *e.g.*, `MyProcDir` to ensure that you always have a clean copy of the Template directory.

4. Go to the new `MyProcDir` directory and specify your process in the file `./Cards/proc_card.dat`. This is the $(n+1)$-particle process you require the subtraction term for.

5. Running `./bin/new_process` generates the code for the $(n+1)$-particle matrix element and for all dipole terms. After running this you will find a newly generated

directory `./Subprocesses/P0_yourprocess` (*e.g.*, `./Subprocesses/P0_e+e-_uuxg`) which contains all required files.

6. For running the check program change to this directory and run `make` and `./check`.

The directory `./Subprocesses/P0_yourprocess` contains all necessary files needed for further calculations. As in the usual MadGraph code the $(n+1)$-particle matrix element is included in the file `matrix.f`. For the dipoles there exist several files called `dipol???.f` where `???` stands for a number starting from `001`. Each file contains only one dipole. Note that the syntax for calling the dipoles is exactly the same as calling the $(n+1)$-particle matrix element. In particular, also the dipoles have to be called with $(n+1)$ momenta rather than with only n momenta. The remapping of the momenta going from the $(n+1)$-particle phase space to a n-parton phase space is done within the code for the dipoles.
The file `dipolsum.f` calculates the sum over all dipoles for a given $(n+1)$-particle phase space point. It contains two subroutines called `DIPOLSUM` and `DIPOLSUMFINITE`. As discussed in Section 2.2 above, the subroutine `DIPOLSUMFINITE` contains the dipoles which only contribute potentially large logarithms but not a real singularity.
In both subroutines the contribution of a certain dipole is only added to the sum if the phase space restriction specified by the α parameter is fulfilled. The value of the four α parameters can be changed in these two subroutines. They are all set to unity by default.

5. Conclusions

In this paper we have presented MadDipole, an implementation to fully automatize the calculation of the dipole subtraction formalism for massless and massive partons in the MadGraph/MadEvent framework. The implementation is done in such a way that the user only needs to specify the desired $(n+1)$-particle process and our code returns a Fortran code for all dipoles combined with possible Born processes which can lead to the $(n+1)$ process specified by the user.
For the calculation of the new color factors we have used as far as possible the routines already provided by the original MadGraph code. We inserted the two additional operators for emitter and spectator and modified the evaluation of the squared color factors.
For the contributions that are not diagonal in helicity space we again used the already available routines for calculating amplitudes for a given helicity combination and combined amplitudes with different helicity combinations to yield the off-diagonal helicity contributions to the subtraction terms.
We have validated the code on numerous different processes with massive and massless partons using two checking procedures. The ratio of $\mathcal{M}^2_{n+1}/\sum(dipoles)$ was confirmed to approach unity as one approaches any soft or collinear limit. The package includes a file `check_dip.f` which allows the user to reproduce this check for any process under consideration.
Secondly, we compared our code against the results of MCFM [10,11], where subtraction is performed using the dipole formalism, finding point-wise agreement wherever anticipated.

Differences with MCFM are understood to be due to different details in the implementation.
The MadDipole package allows the automated computation of real radiation dipole subtraction terms required for NLO calculations. Together with the fastly developing automation of one-loop calculations of multi-leg processes, it could lead to a full automation of NLO calculations for collider processes. In view of the large number of potentially relevant multi-particle production processes at LHC, such automation will be crucial for precision phenomenology, in order to establish and interpret potential deviations from standard model expectations.

Acknowledgments

We would like to thank Fabio Maltoni and Keith Hamilton for useful comments and discussions. NG wants to thank Gabor Somogyi for useful discussions and comparing results. He also would like to thank the University of Louvain for kind hospitality where part of this work was done. This work was partially supported by the Belgian Federal Science Policy (IAP 6/11) and by the Swiss National Science Foundation (SNF) under contract 200020-117602.

References

[1] T. Stelzer and W. F. Long, "Automatic generation of tree level helicity amplitudes," *Comput. Phys. Commun.* **81** (1994) 357–371, `hep-ph/9401258`.

[2] F. Maltoni and T. Stelzer, "MadEvent: Automatic event generation with MadGraph," *JHEP* **02** (2003) 027, `hep-ph/0208156`.

[3] J. Alwall *et al.*, "MadGraph/MadEvent v4: The New Web Generation," *JHEP* **09** (2007) 028, `0706.2334`.

[4] **CompHEP** Collaboration, E. Boos *et al.*, "CompHEP 4.4: Automatic computations from Lagrangians to events," *Nucl. Instrum. Meth.* **A534** (2004) 250–259, `hep-ph/0403113`.

[5] A. Pukhov, "CalcHEP 3.2: MSSM, structure functions, event generation, batchs, and generation of matrix elements for other packages," `hep-ph/0412191`.

[6] T. Gleisberg *et al.*, "SHERPA 1.α, a proof-of-concept version," *JHEP* **02** (2004) 056, `hep-ph/0311263`.

[7] W. Kilian, "WHIZARD 1.0: A generic Monte-Carlo integration and event generation package for multi-particle processes. Manual,". LC-TOOL-2001-039.

[8] M. L. Mangano, M. Moretti, F. Piccinini, R. Pittau, and A. D. Polosa, "ALPGEN, a generator for hard multiparton processes in hadronic collisions," *JHEP* **07** (2003) 001, `hep-ph/0206293`.

[9] A. Kanaki and C. G. Papadopoulos, "HELAC: A package to compute electroweak helicity amplitudes," *Comput. Phys. Commun.* **132** (2000) 306–315, `hep-ph/0002082`.

[10] J. M. Campbell and R. K. Ellis, "An update on vector boson pair production at hadron colliders," *Phys. Rev.* **D60** (1999) 113006, `hep-ph/9905386`.

[11] J. Campbell and R. K. Ellis, "Next-to-leading order corrections to W + 2jet and Z + 2jet production at hadron colliders," *Phys. Rev.* **D65** (2002) 113007, hep-ph/0202176.

[12] Z. Nagy, "Next-to-leading order calculation of three-jet observables in hadron hadron collision," *Phys. Rev.* **D68** (2003) 094002, hep-ph/0307268.

[13] S. Frixione and B. R. Webber, "Matching NLO QCD computations and parton shower simulations," *JHEP* **06** (2002) 029, hep-ph/0204244.

[14] S. Frixione and B. R. Webber, "The MC@NLO 3.3 event generator," hep-ph/0612272.

[15] P. Nason, "A new method for combining NLO QCD with shower Monte Carlo algorithms," *JHEP* **11** (2004) 040, hep-ph/0409146.

[16] P. Nason and G. Ridolfi, "A positive-weight next-to-leading-order Monte Carlo for Z pair hadroproduction," *JHEP* **08** (2006) 077, hep-ph/0606275.

[17] O. Latunde-Dada, S. Gieseke, and B. Webber, "A positive-weight next-to-leading-order Monte Carlo for e+ e- annihilation to hadrons," *JHEP* **02** (2007) 051, hep-ph/0612281.

[18] S. Frixione, P. Nason, and G. Ridolfi, "A Positive-Weight Next-to-Leading-Order Monte Carlo for Heavy Flavour Hadroproduction," *JHEP* **09** (2007) 126, 0707.3088.

[19] S. Alioli, P. Nason, C. Oleari, and E. Re, "NLO vector-boson production matched with shower in POWHEG," *JHEP* **07** (2008) 060, 0805.4802.

[20] K. Hamilton, P. Richardson, and J. Tully, "A Positive-Weight Next-to-Leading Order Monte Carlo Simulation of Drell-Yan Vector Boson Production," 0806.0290.

[21] D. B. Melrose, "Reduction of Feynman diagrams," *Nuovo Cim.* **40** (1965) 181–213.

[22] Z. Bern, L. J. Dixon, and D. A. Kosower, "Dimensionally regulated pentagon integrals," *Nucl. Phys.* **B412** (1994) 751–816, hep-ph/9306240.

[23] Z. Bern, L. J. Dixon, D. C. Dunbar, and D. A. Kosower, "One loop N point gauge theory amplitudes, unitarity and collinear limits," *Nucl. Phys.* **B425** (1994) 217–260, hep-ph/9403226.

[24] Z. Bern and A. G. Morgan, "Massive Loop Amplitudes from Unitarity," *Nucl. Phys.* **B467** (1996) 479–509, hep-ph/9511336.

[25] Z. Bern, L. J. Dixon, and D. A. Kosower, "Bootstrapping multi-parton loop amplitudes in QCD," *Phys. Rev.* **D73** (2006) 065013, hep-ph/0507005.

[26] C. F. Berger, Z. Bern, L. J. Dixon, D. Forde, and D. A. Kosower, "Bootstrapping one-loop QCD amplitudes with general helicities," *Phys. Rev.* **D74** (2006) 036009, hep-ph/0604195.

[27] D. Forde, "Direct extraction of one-loop integral coefficients," *Phys. Rev.* **D75** (2007) 125019, 0704.1835.

[28] A. Denner, S. Dittmaier, M. Roth, and L. H. Wieders, "Electroweak corrections to charged-current $e^+ e^- \to 4$ fermion processes: Technical details and further results," *Nucl. Phys.* **B724** (2005) 247–294, hep-ph/0505042.

[29] A. Denner and S. Dittmaier, "Reduction schemes for one-loop tensor integrals," *Nucl. Phys.* **B734** (2006) 62–115, hep-ph/0509141.

[30] T. Binoth, J. P. Guillet, G. Heinrich, E. Pilon, and C. Schubert, "An algebraic / numerical formalism for one-loop multi-leg amplitudes," *JHEP* **10** (2005) 015, hep-ph/0504267.

[31] T. Binoth, J. P. Guillet, and G. Heinrich, "Algebraic evaluation of rational polynomials in one-loop amplitudes," *JHEP* **02** (2007) 013, hep-ph/0609054.

[32] R. Britto, B. Feng, and P. Mastrolia, "The cut-constructible part of QCD amplitudes," *Phys. Rev.* **D73** (2006) 105004, hep-ph/0602178.

[33] P. Mastrolia, "On triple-cut of scattering amplitudes," *Phys. Lett.* **B644** (2007) 272–283, hep-th/0611091.

[34] R. Britto and B. Feng, "Integral Coefficients for One-Loop Amplitudes," *JHEP* **02** (2008) 095, 0711.4284.

[35] R. Britto, B. Feng, and P. Mastrolia, "Closed-Form Decomposition of One-Loop Massive Amplitudes," 0803.1989.

[36] C. Anastasiou, R. Britto, B. Feng, Z. Kunszt, and P. Mastrolia, "Unitarity cuts and reduction to master integrals in d dimensions for one-loop amplitudes," *JHEP* **03** (2007) 111, hep-ph/0612277.

[37] R. K. Ellis, W. T. Giele, and Z. Kunszt, "A Numerical Unitarity Formalism for Evaluating One-Loop Amplitudes," *JHEP* **03** (2008) 003, 0708.2398.

[38] W. T. Giele, Z. Kunszt, and K. Melnikov, "Full one-loop amplitudes from tree amplitudes," *JHEP* **04** (2008) 049, 0801.2237.

[39] R. K. Ellis, W. T. Giele, Z. Kunszt, and K. Melnikov, "Masses, fermions and generalized D-dimensional unitarity," 0806.3467.

[40] G. Ossola, C. G. Papadopoulos, and R. Pittau, "Reducing full one-loop amplitudes to scalar integrals at the integrand level," *Nucl. Phys.* **B763** (2007) 147–169, hep-ph/0609007.

[41] G. Ossola, C. G. Papadopoulos, and R. Pittau, "On the Rational Terms of the one-loop amplitudes," *JHEP* **05** (2008) 004, 0802.1876.

[42] P. Mastrolia, G. Ossola, C. G. Papadopoulos, and R. Pittau, "Optimizing the Reduction of One-Loop Amplitudes," *JHEP* **06** (2008) 030, 0803.3964.

[43] G. Ossola, C. G. Papadopoulos, and R. Pittau, "CutTools: a program implementing the OPP reduction method to compute one-loop amplitudes," *JHEP* **03** (2008) 042, 0711.3596.

[44] C. F. Berger *et al.*, "An Automated Implementation of On-Shell Methods for One- Loop Amplitudes," 0803.4180.

[45] W. T. Giele and G. Zanderighi, "On the Numerical Evaluation of One-Loop Amplitudes: the Gluonic Case," 0805.2152.

[46] T. Binoth *et al.*, "Precise predictions for LHC using a GOLEM," 0807.0605.

[47] W. T. Giele and E. W. N. Glover, "Higher order corrections to jet cross-sections in e^+ e^- annihilation," *Phys. Rev.* **D46** (1992) 1980–2010.

[48] Z. Kunszt and D. E. Soper, "Calculation of jet cross-sections in hadron collisions at order α_s^3," *Phys. Rev.* **D46** (1992) 192–221.

[49] S. Frixione, Z. Kunszt, and A. Signer, "Three-jet cross sections to next-to-leading order," *Nucl. Phys.* **B467** (1996) 399–442, hep-ph/9512328.

[50] S. Catani and M. H. Seymour, "A general algorithm for calculating jet cross sections in NLO QCD," *Nucl. Phys.* **B485** (1997) 291–419, hep-ph/9605323.

[51] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, "The dipole formalism for next-to-leading order QCD calculations with massive partons," *Nucl. Phys.* **B627** (2002) 189–265, hep-ph/0201036.

[52] D. A. Kosower, "Antenna factorization of gauge-theory amplitudes," *Phys. Rev.* **D57** (1998) 5410–5416, hep-ph/9710213.

[53] J. M. Campbell, M. A. Cullen, and E. W. N. Glover, "Four jet event shapes in electron positron annihilation," *Eur. Phys. J.* **C9** (1999) 245–265, hep-ph/9809429.

[54] A. Gehrmann-De Ridder, T. Gehrmann, and E. W. N. Glover, "Antenna subtraction at NNLO," *JHEP* **09** (2005) 056, hep-ph/0505111.

[55] A. Daleo, T. Gehrmann, and D. Maître, "Antenna subtraction with hadronic initial states," *JHEP* **04** (2007) 016, hep-ph/0612257.

[56] T. Gleisberg and F. Krauss, "Automating dipole subtraction for QCD NLO calculations," *Eur. Phys. J.* **C53** (2008) 501–523, 0709.2881.

[57] M. H. Seymour and C. Tevlin, "TeVJet: A general framework for the calculation of jet observables in NLO QCD," 0803.2231.

[58] K. Hasegawa, S. Moch, and P. Uwer, "Automating dipole subtraction," 0807.3701.

[59] H. Murayama, I. Watanabe, and K. Hagiwara, "HELAS: HELicity amplitude subroutines for Feynman diagram evaluations,". KEK-91-11.

[60] Z. Nagy and Z. Trocsanyi, "Next-to-leading order calculation of four-jet observables in electron positron annihilation," *Phys. Rev.* **D59** (1999) 014020, hep-ph/9806317.

[61] J. Campbell, R. K. Ellis, and F. Tramontano, "Single top production and decay at next-to-leading order," *Phys. Rev.* **D70** (2004) 094012, hep-ph/0408158.

[62] J. Campbell and F. Tramontano, "Next-to-leading order corrections to W t production and decay," *Nucl. Phys.* **B726** (2005) 109–130, hep-ph/0506289.

[63] J. Campbell, R. Frederix, F. Maltoni, and F. Tramontano, in preparation.