

## The automation of subtraction schemes for next-to-leading order calculations in QCD

---

**Rikkert Frederix\***

*Institute for Theoretical Physics, Universität Zürich*

*E-mail: frederix@physik.uzh.ch*

In this talk we give an overview of the existing codes that aim for the automation of the subtraction terms in next-to-leading order calculations in perturbative QCD.

*13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research  
February 22-27, 2010  
Jaipur, India*

---

\*Speaker.

## 1. Introduction

Next-to-leading order (NLO) in QCD predictions consist of three distinct contributions, which can schematically presented as

$$\sigma^{\text{NLO}} = \int_m d\sigma^{\text{Born}} + \int_m d\sigma^{\text{virt.}} + \int_{m+1} d\sigma^{\text{real}}. \quad (1.1)$$

The first term is the Born contribution, which is the leading order (LO) contribution to the process under consideration. The second term is the virtual contribution that consists of the same  $m$ -particle final state as the Born, but has one more power of the strong coupling. This means that there must be a closed loop of particles with a momentum that has to be integrated over in the diagrams contributing to this term. The third term in Eq. (1.1) is the real contribution. Also this contribution has one power of the strong coupling more than the Born contributions, but also one QCD parton extra in the final state, which may or may not be resolved. Hence, like the Born, this is a tree-level contribution.

The sum of the three contributions in Eq. (1.1) is finite, but the individual contributions can be divergent. In particular, the loop integral in the (ultraviolet renormalized) virtual corrections over the internal momentum is divergent. Also the phase-space integral over the (possibly) unresolved particle in the real emission is divergent. These divergences are of infrared origin and only after the virtual and the real-emission contributions are added together, they cancel against each other. In almost all practical calculations for NLO QCD predictions these integrations are performed using a dimensional regularization scheme in which the dimension of these integrals is shifted away from 4, conventionally by a factor  $2\epsilon$ . After integration the divergences will show up as explicit poles in  $1/\epsilon$  and their cancellation can be verified explicitly.

The calculation of an integral in a non-integer number of dimensions can only be done analytically. But experimental analyses require the possibility to make complicated phase-space cuts and therefore it is an impossible task to perform the phase-space integrals by analytic means. A Monte Carlo technique is favoured. However, this numerical approach implies that the phase-space integrals have to be performed in 4 dimensions, which leads to infrared divergences for the real-emission contributions.

There are two classes of solutions to this problem. They are the (approximate) phase-space slicing or the (exact) subtraction method; it is nowadays acknowledged that the slicing method is unsuited for describing complicated final states, such as those in multi-jet production. In the subtraction method a term which has the same singularity structure as the real-emission corrections is added to these contributions. This cancels the infrared divergences in the phase space integral. The subtraction terms should be simple enough such that the one-particle phase-space integral (of the unresolved particle) can be done analytically in dimensional regularization. Schematically we can write this as:

$$\sigma^{\text{NLO}} = \int_m \left[ d\sigma^{\text{Born}} + d\sigma^{\text{virt.}} + \int_1 d\sigma^{\text{subtr.}} \right] + \int_{m+1} \left[ d\sigma^{\text{real}} - d\sigma^{\text{subtr.}} \right]. \quad (1.2)$$

The remaining phase-space integrals can now be performed numerically and give finite results for infrared-safe observables.

Recently the generation of these subtraction terms has been completely automated. There are three different levels of automation that the available packages aim for. Firstly, there are the codes that simply take care of the bookkeeping and automate the generation of the subtraction terms for a given process. These can then be copied into a separate program to compute the phase-space integrals etc. The second possibility is to have the full NLO computation Eq. (1.2) automated. This includes the phase-space integration. Usually the virtual corrections are linked from an external package through the Binoth-Les Houches Accord interface [1]. Thirdly, there is a class of packages that aim for the automation of NLO computations and linking them with parton showers. In this process also the subtraction terms are needed.

The two most widely known and used subtraction methods are the Catani-Seymour dipole subtraction [2,3] and the FKS (or residue) subtraction by Frixione, Kunszt and Signer [4,5]. In the next section we shortly review these two methods, in section 3 we discuss the available packages that have these subtraction methods automated. We end with a short summary.

## 2. Automated subtraction

At the moment there are two subtraction schemes automated in computer codes: the FKS (or residue) subtraction and the Catani-Seymour dipole (CS) subtraction. These two subtraction schemes will be reviewed here in a schematic way.

For more detailed explanations we refer to the papers of Refs. [2,3] and Refs. [4–6] for the CS and FKS subtraction schemes, respectively.

### 2.1 FKS subtraction

The real-emission contribution in Eq. (1.1) can be written schematically as

$$d\sigma^{\text{real}} = |M^{m+1}|^2 d\phi_{m+1}, \quad (2.1)$$

where  $|M^{m+1}|^2$  is the matrix element squared and  $d\phi_{m+1}$  the phase-space measure. The matrix element squared blows up<sup>1</sup> like  $\frac{1}{\xi_i} \frac{1}{1-y_{ij}}$ , where  $\xi_i$  is the energy of final state particle  $i$  (in the partonic center-of-mass frame) over the total partonic energy,  $\xi_i = 2E_i/\sqrt{\hat{s}}$ , and  $y_{ij}$  is the cosine of the angle between particle  $i$  and  $j$ ,  $y_{ij} = \cos \theta_{ij}$ .

The crucial step in the FKS subtraction method is to realize that the phase-space can be partitioned into regions that have at most one collinear and/or one soft divergence. This can be achieved by multiplying Eq. (2.1) by so-called  $S$ -functions

$$d\sigma^{\text{real}} = \sum_{ij\text{-pairs}} S_{ij} |M^{m+1}|^2 d\phi_{m+1}, \quad (2.2)$$

where the  $S$ -functions have been defined in such a way that they vanish in all singular limits not related to particle  $i$  becoming soft or particles  $i$  and  $j$  collinear and that the sum over all pairs gives one,  $\sum_{ij\text{-pairs}} S_{ij} = 1$ . The precise definition of the  $S$ -functions used is not important. The result is

---

<sup>1</sup>The phase space measure includes an implicit factor  $\xi_i$ . It has been assumed that this has already canceled one of the  $\frac{1}{\xi_i}$  in the matrix element squared.

that each term of the sum in Eq. (2.2) is finite over all of phase space except if the energy of particle  $i$  goes to zero or particles  $i$  and  $j$  become collinear.

Because we know exactly where the infrared singularities are for a given partition, it is now relatively straight-forward to regularize these divergences. This amounts to replacing each of the terms in the sum of Eq. (2.2) by

$$\left(\frac{1}{\xi_i}\right)_{\xi_{cut}} \left(\frac{1}{1-y_{ij}}\right)_{\delta_0} \xi_i(1-y_{ij})S_{ij}|M^{m+1}|^2 d\phi_{m+1}, \quad (2.3)$$

where we have used generalized plus distributions defined, with a smooth test function  $f(x)$ , as

$$\int dx \left(\frac{1}{x}\right)_{x_{cut}} f(x) = \int dx \frac{f(x) - f(0)\Theta(x_{cut} - x)}{x}. \quad (2.4)$$

This leads to (maximally) three counter terms for a single  $(m+1)$ -particle event in a given phase-space partition:

- the soft counter event:  $\xi_i = 0$ ;
- the collinear counter event:  $y_{ij} = 1$ ;
- and the soft-collinear counter event:  $\xi_i = 0$  and  $y_{ij} = 1$ ;

Of course, for numerical evaluation the explicit  $\xi_i$  and  $(1 - y_{ij})$  in Eq. (2.3) should be canceled analytically against the divergences in the matrix element squared. But this yields no problem, because it is appreciated that in the collinear limit the real-emission matrix element squared is equal to the Born times the Altarelli-Parisi splitting functions and in the soft limit by the color-linked Borns times the eikonals. This defines the last two terms of Eq. (1.2).

In order not to change the NLO prediction, the “integrated subtraction terms” need to be added to the virtual corrections, *i.e.*, the third term in Eq. (1.2). The integral over the unresolved particle can be performed analytically and is process independent; it results in terms proportional to the (color-linked) Borns. For the explicit formulas of these integrated subtraction terms we refer to the paper of Ref. [6].

## 2.2 Catani-Seymour dipole subtraction

The fundamental building blocks of the subtraction terms in the dipole formalism [2, 3] 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  $\tilde{i}, \tilde{k}$ . The full soft behavior is recovered after summing all dipole factors.

Here, we use the notation introduced in Refs. [2, 3]. Independent of whether we have initial or final state particles we can write an arbitrary dipole in the form

$$\mathcal{D}_{ij,k} \sim {}_m\langle 1, \dots, \tilde{i}, \dots, \tilde{k}, \dots, m+1 | \frac{\mathbf{T}_k \cdot \mathbf{T}_{ij}}{\mathbf{T}_{ij}^2} \mathbf{V}_{ij,k} | 1, \dots, \tilde{i}, \dots, \tilde{k}, \dots, m+1 \rangle_m. \quad (2.5)$$

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 are summed over after contraction with the dipole splitting function.

In the limit of a large number of external particles  $N$ , the number of subtraction terms needed scales like  $N^3$ . Furthermore, due to different momentum mappings for each of the dipoles, the Born matrix elements, *i.e.*, the bra's and ket's in Eq. (2.5), have to be computed many times for a given real-emission phase-space point. Therefore the calculation of the sum of all subtraction terms can be more time consuming than the real emission itself. An clever trick was introduced by Nagy to restrict the phase space of the dipoles and therefore the average number of subtraction terms per real-emission phase-space point is greatly reduced [7].

### 2.2.1 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 principal 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 the introduction of a parameter usually labelled  $\alpha$  with  $\alpha \in [0, 1]$ , which was introduced in Ref. [7] 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. [8].

Using the notation of Ref. [8], the contribution from the subtraction term to the differential cross section can be written as

$$\begin{aligned}
 d\sigma_{ab}^A = & \sum_{\{n+1\}} d\Gamma^{(n+1)}(p_a, p_b, p_1, \dots, p_{n+1}) \frac{1}{S_{\{n+1\}}} \\
 & \times \left\{ \sum_{\substack{\text{pairs} \\ i,j}} \sum_{k \neq i,j} \mathcal{D}_{ij,k}(p_a, p_b, p_1, \dots, p_{n+1}) F_J^{(n)}(p_a, p_b, p_1, \dots, \tilde{p}_{ij}, \tilde{p}_k, \dots, p_{n+1}) \Theta(y_{ij,k} < \alpha) \right. \\
 & + \sum_{\substack{\text{pairs} \\ i,j}} \left[ \mathcal{D}_{ij}^a(p_a, p_b, p_1, \dots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, p_1, \dots, \tilde{p}_{ij}, \dots, p_{n+1}) \Theta(1 - x_{ij,a} < \alpha) \right. \\
 & \quad \left. \left. + (a \leftrightarrow b) \right] \right. \\
 & + \sum_{i \neq k} \left[ \mathcal{D}_k^{ai}(p_a, p_b, p_1, \dots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, p_1, \dots, \tilde{p}_k, \dots, p_{n+1}) \Theta(u_i < \alpha) + (a \leftrightarrow b) \right] \\
 & \left. + \sum_i \left[ \mathcal{D}^{ai,b}(p_a, p_b, p_1, \dots, p_{n+1}) F_J^{(n)}(\tilde{p}_a, p_b, \tilde{p}_1, \dots, \tilde{p}_{n+1}) \Theta(\tilde{v}_i < \alpha) + (a \leftrightarrow b) \right] \right\}. \tag{2.6}
 \end{aligned}$$

The functions  $\mathcal{D}_{ij,k}$ ,  $\mathcal{D}_{ij}^a$ ,  $\mathcal{D}_k^{ai}$  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. In general, different numerical values for  $\alpha$  can be chosen for the final-final, final-initial, initial-final and initial-initial dipoles.

It has to be kept in mind that the integrated dipole factors, which are to be added to the virtual  $n$ -parton contribution, will also depend on  $\alpha$ . For the case of massless partons, the  $\alpha$ -dependence of the integrated terms is stated in [7, 8] while for massive partons results for most cases can be found in [9–11].

### 3. Available packages

As explained in the introduction, there are three directions of automation. For each direction there are packages available, which we will shortly describe here.

#### 3.1 Automation of the subtraction terms

The codes describe here are for the automation of the bookkeeping and generation of the subtraction terms only. In particular, they do not consider the phase-space integration. The two packages described here are using the Catani-Seymour dipole subtraction method and rely on MadGraph [12] to generate the matrix elements.

**MadDipole** Given any real emission process, MadDipole [13] generates that process and all the subtraction terms needed for it. Both the massless and the massive dipoles have been implemented, including the phase-space restriction according to the  $\alpha$  parameter. The integrated subtraction terms are all implemented and will be available soon. It is available from any of the MadGraph/MadEvent websites, *e.g.*, <http://madgraph.phys.ucl.ac.be/>.

**AutoDipole** Using external Mathematica routines for the bookkeeping of the subtraction terms and the generation of the color matrices of the subtraction terms, AutoDipole [14] uses the MadGraph matrix elements for the generation of the Born and real-emission contributions. It has the (integrated) dipoles for both the massless and the massive particles, but lacks the  $\alpha$  dependence to restrict the phase space of the dipoles. The code is available for download at the webpage <http://www-zeuthen.desy.de/~moch/autodipole/index.html>.

#### 3.2 Full NLO

Compared to the previous subsection, the packages that will be described here aim at the full automation of NLO computations, excluding the finite part of the virtual corrections: they can do the same as the codes in the previous section but on top of that the main addition is the automation of the phase space generation and integration. There are three packages publicly available that have the CS dipoles automated, Sherpa [15], TevJet [16] and Helac-Dipoles [17], and one package that will be publicly available in the near future that has the FKS subtraction implemented [6].

**TevJet** The TevJet package, written by Seymour and Tevlin in C++, has the massless CS subtraction terms automated. The user must provide the ingredients for a NLO computation, *i.e.*, the Born and real emission matrix elements, the color-linked and off-diagonal helicity Borns, and the virtual corrections. The TevJet package takes care of the bookkeeping and generating the subtraction terms (for massless particles only). General purpose 2-, 3- and 4-body phase spaces are available within the package to allow for a phase-space integration. The code can be downloaded from the webpage <http://www.hep.man.ac.uk/u/chris/tevjet/>.

**Sherpa** The first automation of the NLO subtraction terms was within the Sherpa package in 2007 [15]. It is based on the CS dipoles, including the massive contributions and the  $\alpha$ -parameter restriction. It has a phase-space generator specially designed to have an efficient integration of the matrix elements minus subtraction terms. It has been proven to work for complicated final states, in particular, together with the BlackHat collaboration [18] has given predictions for  $W + 3$  jets production at the Tevatron and LHC [19]. Recently, it has become publicly available and can be downloaded from the Sherpa webpage, <http://www.sherpa-mc.de/>.

**Helac-Dipoles** The third package that has an implementation of the CS dipole subtraction terms is the Helac-Dipoles package [17], based upon the Helac Monte Carlo generator [20]. Like the Sherpa implementation it also has the massless and massive dipoles implemented as well as the  $\alpha$ -parameter restriction of the dipole phase space. Furthermore, it has the possibility to have the subtraction terms for a given helicity configuration which allows for a Monte-Carlo sum over helicities [17]. Also this code has been proven to work: *e.g.*, it has given predictions for the most impressive  $tt + 2$  jets predictions at hadron colliders [21], using the Helac-1Loop package [22] for the virtual corrections. The code can be downloaded from the Helac webpage, <http://helac-phegas.web.cern.ch/helac-phegas/>.

**MadFKS** Contrary to the Sherpa, TevJet and the Helac-Dipoles packages, the MadFKS code [6] has the FKS subtraction implemented within the MadGraph/MadEvent MC event generator [12]. Also in this code both the massless and massive subtraction terms are implemented, including the possible phase-space restriction for the subtraction terms. The possibility for a Monte-Carlo sum over helicities is available as well. Furthermore, due to the nature of the FKS subtraction, it is also suited for subtraction for colored BSM particles. The most impressive result obtained with the code, is the calculation of 5 jets production at the LEP collider at NLO [23]. For this calculation the finite part of the virtual corrections are provided by the Rocket code [24], and have been checked using the BlackHat program [18]. The MadFKS code is not yet publicly available.

### 3.3 NLO with possible link to Parton Shower

Although in principle linking NLO calculations to parton showers does not depend on the subtraction method used, in practice, the only method proven to work (in hadron collisions) is the FKS subtraction.

**Powheg Box** The recently written package ‘Powheg Box’ [25] pursues exactly this: the automation of the linking between the ingredients of a NLO computation and the parton shower. It is a framework where the user should put in the various contributions to any NLO computation, *i.e.*, the Born, the real emission, the virtual corrections, the color-linked Borns, the off-diagonal helicity Borns and the Born phase space. From these ingredients the Powheg Box builds a full NLO computation linked to a parton shower in an automated way. In particular it has also the FKS subtraction implemented to deal with the soft and collinear singularities. The Powheg Box can be downloaded from <http://moby.mib.infn.it/~nason/POWHEG/>.

## 4. Summary

For NLO computation in QCD, there are various packages available that have the CS or FKS subtraction methods automated. There are two public packages that focus on the automation of the subtraction terms without having a specific phase-space generator: MadDipole and AutoDipole. There are also four packages that aim in automating the full NLO computation: there are implementations of the CS dipoles in TevJet, Sherpa, and Helac-dipoles, and there is an implementation of the FKS subtraction in the MadGraph/MadEvent framework, MadFKS. Furthermore there is the Powheg Box package that focusses on the automation of the Powheg method of linking fixed order NLO computations to parton showers. In its internal workings, there is an automated FKS subtraction implemented.

## References

- [1] T. Binoth *et. al.*, *Comput. Phys. Commun.* **181** (2010) 1612–1622, [1001.1307].
- [2] S. Catani and M. H. Seymour, *Nucl. Phys.* **B485** (1997) 291–419, [hep-ph/9605323].
- [3] S. Catani, S. Dittmaier, M. H. Seymour, and Z. Trocsanyi, *Nucl. Phys.* **B627** (2002) 189–265, [hep-ph/0201036].
- [4] S. Frixione, Z. Kunszt, and A. Signer, *Nucl. Phys.* **B467** (1996) 399–442, [hep-ph/9512328].
- [5] S. Frixione, *Nucl. Phys.* **B507** (1997) 295–314, [hep-ph/9706545].
- [6] R. Frederix, S. Frixione, F. Maltoni, and T. Stelzer, *JHEP* **10** (2009) 003, [0908.4272].
- [7] Z. Nagy and Z. Trocsanyi, *Phys. Rev.* **D59** (1999) 014020, [hep-ph/9806317].
- [8] Z. Nagy, *Phys. Rev.* **D68** (2003) 094002, [hep-ph/0307268].
- [9] J. M. Campbell, R. K. Ellis, and F. Tramontano, *Phys. Rev.* **D70** (2004) 094012, [hep-ph/0408158].
- [10] J. M. Campbell and F. Tramontano, *Nucl. Phys.* **B726** (2005) 109–130, [hep-ph/0506289].
- [11] G. Bevilacqua, M. Czakon, C. G. Papadopoulos, R. Pittau, and M. Worek, *JHEP* **09** (2009) 109, [0907.4723].
- [12] J. Alwall *et. al.*, *JHEP* **09** (2007) 028, [0706.2334].
- [13] R. Frederix, T. Gehrmann, and N. Greiner, *JHEP* **09** (2008) 122, [0808.2128].
- [14] K. Hasegawa, S. Moch, and P. Uwer, *Comput. Phys. Commun.* **181** (2010) 1802–1817, [0911.4371].
- [15] T. Gleisberg and F. Krauss, *Eur. Phys. J.* **C53** (2008) 501–523, [0709.2881].
- [16] M. H. Seymour and C. Tevlin, 0803.2231.
- [17] M. Czakon, C. G. Papadopoulos, and M. Worek, *JHEP* **08** (2009) 085, [0905.0883].
- [18] C. F. Berger *et. al.*, *Phys. Rev.* **D78** (2008) 036003, [0803.4180].
- [19] C. F. Berger *et. al.*, *Phys. Rev. Lett.* **102** (2009) 222001, [0902.2760].
- [20] A. Kanaki and C. G. Papadopoulos, *Comput. Phys. Commun.* **132** (2000) 306–315, [hep-ph/0002082].



- [21] G. Bevilacqua, M. Czakon, C. G. Papadopoulos, and M. Worek, *Phys. Rev. Lett.* **104** (2010) 162002, [1002.4009].
- [22] A. van Hameren, C. G. Papadopoulos, and R. Pittau, *JHEP* **09** (2009) 106, [0903.4665].
- [23] R. Frederix, S. Frixione, K. Melnikov, and G. Zanderighi, 1008.5313.
- [24] R. K. Ellis, K. Melnikov, and G. Zanderighi, *Phys. Rev.* **D80** (2009) 094002, [0906.1445].
- [25] S. Alioli, P. Nason, C. Oleari, and E. Re, *JHEP* **06** (2010) 043, [1002.2581].