

A new method for one-loop amplitude generation and reduction in OpenLoops

Federico Buccioni

University of Zurich, Zurich, SWITZERLAND

E-mail: buccioni@physik.uzh.ch

Stefano Pozzorini

University of Zurich, Zurich, SWITZERLAND

E-mail: pozzorin@physik.uzh.ch

Max Zoller*

University of Zurich, Zurich, SWITZERLAND

E-mail: zoller@physik.uzh.ch

We describe a new method [1] for the automated construction of one-loop amplitudes based on the open-loop algorithm, where various operations are performed on-the-fly while constructing the integrand. In particular, an on-the-fly reduction interleaved with the construction steps of the amplitude keeps the maximum tensor rank in the loop momentum at two throughout the algorithm, thus drastically reducing the complexity of the calculation. The full reduction to scalar integrals is unified with the amplitude construction in a single recursion within the OPENLOOPS framework. This approach strongly exploits the factorisation of one-loop integrands in a product of loop segments. The on-the-fly approach, which is also applied to helicity summation and the merging of different diagrams, increases the speed of the original open-loop algorithm in a very significant way. A remarkably high level of numerical stability is achieved by exploiting freedoms in reduction identities and through simple expansions in rank-two Gram determinants. These features are particularly attractive for NLO multi-leg and NNLO real-virtual calculations. The new algorithm will be made public in a forthcoming release of the OPENLOOPS program.

13th International Symposium on Radiative Corrections (Applications of Quantum Field Theory to Phenomenology)

25-29 September, 2017

St. Gilgen, Austria

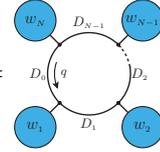
*Speaker.

1. Automated amplitude generation in OpenLoops

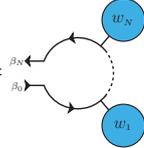
In the last decade, powerful methods for the calculation of one-loop scattering amplitudes have been developed. Highly automated one-loop tools, such as OPENLOOPS [2,3], provide the key to achieving NLO precision in multi-purpose Monte Carlo generators. The helicity- and colour-summed scattering probability densities for an n -particle process

$$\mathcal{W}_{\text{LO}} = \sum_{\text{hel,col}} |\mathcal{M}_0|^2, \quad \mathcal{W}_{\text{NLO}}^{\text{virtual}} = \sum_{\text{hel,col}} 2\text{Re}[\mathcal{M}_0^* \mathcal{M}_1] \quad \text{with } \mathcal{M}_l = \sum_d \mathcal{M}_l^{(d)} \quad (1.1)$$

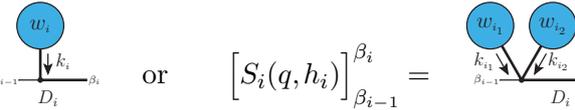
are computed as sums of Feynman diagrams d with $l = 0, 1$ loops and n external particles. A one-loop Feynman diagram amplitude can be written as

$$\mathcal{M}_1^{(d)} = \mathcal{C}_1^{(d)} \int d\bar{q} \frac{\text{Tr}[\mathcal{N}(q)]}{\bar{D}_0(\bar{q}) \cdots \bar{D}_{N-1}(\bar{q})} = \text{Diagram} \quad (1.2)$$


with a colour factor $\mathcal{C}_1^{(d)}$ and scalar propagators $\bar{D}_i(\bar{q}) = (\bar{q} + p_i)^2 - m_i^2$, which contain the loop momentum \bar{q} , the mass m_i and the external momentum p_i .¹ This is computed by cutting the loop open at one propagator \bar{D}_0 and numerically constructing the numerator²

$$[\mathcal{N}(q, h)]_{\beta_0}^{\beta_N} = \text{Diagram} = [S_1(q, h_1)]_{\beta_0}^{\beta_1} [S_2(q, h_2)]_{\beta_1}^{\beta_2} \cdots [S_N(q, h_N)]_{\beta_{N-1}}^{\beta_N}, \quad (1.3)$$


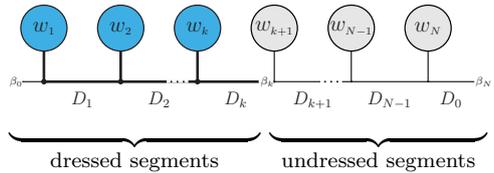
where $\beta_{0,N}$ are the Lorentz or spinor indices of the cut propagator. Finally, the trace is taken by multiplying with $\delta_{\beta_0}^{\beta_N}$ and the tensor integrals are evaluated. A key feature exploited in our approach is the factorisation of the numerator into segments $S_i(q, h_i)$, each consisting of a loop vertex and propagator and one or two external subtrees $w_i(h_i)$,

$$[S_i(q, h_i)]_{\beta_{i-1}}^{\beta_i} = \text{Diagram} \quad \text{or} \quad [S_i(q, h_i)]_{\beta_{i-1}}^{\beta_i} = \text{Diagram} \quad (1.4)$$


We construct the numerator recursively through so-called *dressing* steps

$$\mathcal{N}_k(q) = \mathcal{N}_{k-1}(q) S_k(q), \quad k \leq N, \quad (1.5)$$

starting from the initial condition $\mathcal{N}_0 = \mathbb{1}$. The partially dressed numerator, which we call an open loop, is a q -polynomial³,

$$\mathcal{N}_k(q) = \text{Diagram} = \prod_{i=1}^k S_i(q) = \sum_{r=0}^R \mathcal{N}_{\mu_1 \dots \mu_r}^{(k)} q^{\mu_1} \cdots q^{\mu_r}. \quad (1.6)$$


¹The bar marks D -dimensional quantities as opposed to four-dimensional ones, where $D = 4 - 2\epsilon$.

²Here we explicitly include the dependence on the global helicity configuration h , or equivalently the helicity configurations h_1, \dots, h_N of the external subtrees in the individual segments.

³Dressing steps are performed numerically at the level of the coefficients $\mathcal{N}_{\mu_1 \dots \mu_r}^{(n)}$

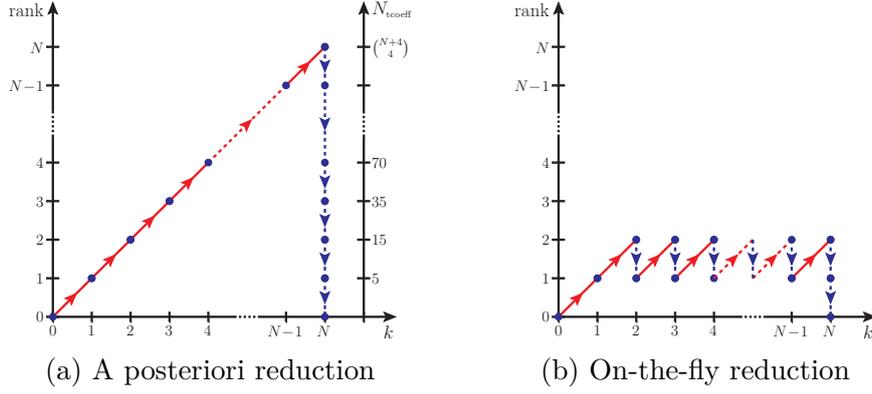


Figure 1: Evolution of the tensor rank R and number $N_{\text{tcoeff}}(R) = \binom{R+4}{4}$ of open-loop tensor coefficients with the number k of dressed segments (if each dressing step increases the rank by 1).

The rank R is potentially increased by one in each step (see Fig. 1 (a)). In the first version of OPENLOOPS [2,3] the tensor reduction was performed a posteriori with external libraries, such as COLLIER [4] or CUTTOOLS [5]. The basic idea of the on-the-fly reduction is to perform reduction steps at integrand level interleaved with the dressing steps (see Fig. 1 (b)), keeping the rank and hence the complexity of the tensor coefficients low throughout the calculation.

2. The on-the-fly method

The on-the-fly reduction formulas are based on [6] and have the form⁴

$$q^\mu q^\nu = \sum_{i=-1}^3 \left(A_i^{\mu\nu} + B_{i,\lambda}^{\mu\nu} q^\lambda \right) D_i(q), \quad D_i(q) = \begin{cases} 1, & i = -1 \\ (q + p_i)^2 - m_i^2, & i \geq 0 \end{cases} \quad (2.1)$$

where the coefficients $A_i^{\mu\nu}$ and $B_{i,\lambda}^{\mu\nu}$ are q -independent. The loop momentum dependence resides in the reconstructed denominators $D_i = \bar{D}_i - \tilde{q}^2$ which cancel denominators in the full integrand.⁵ Hence in each reduction step from rank-2 to rank-1 open loops new topologies with pinched propagators are created. Due to the factorisation (1.3) of the open loop (2.1) can be applied to a partial integrand, e.g. after two dressing steps (see Fig. 2 (a)), irrespective of segments to be dressed in subsequent steps and further \bar{D}_i ,

$$\left[\frac{N^{\mu\nu} q_\mu q_\nu}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \bar{D}_3} \right] \prod_{i=k+1}^N \frac{S_i(q)}{\bar{D}_{i-1}} = \left[\frac{N_{-1}^\mu q_\mu + N_{-1} + \tilde{N}_{-1} \tilde{q}^2}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \bar{D}_3} + \sum_{i=0}^3 \frac{N_i^\mu q_\mu + N_i}{\bar{D}_0 \cdots \bar{D}_i \cdots \bar{D}_3} \right] \prod_{i=k+1}^N \frac{S_i(q)}{\bar{D}_{i-1}}. \quad (2.2)$$

The reduction step is then followed by the next dressing steps and further reduction steps for each resulting topology until the loop is fully dressed and final reduction steps are performed at integral level (for details see [1]). The freedom to choose $\bar{D}_0, \dots, \bar{D}_3$ from the

⁴This is also valid for triangles in renormalisable theories [1,6] if we set terms involving \bar{D}_3, p_3 to zero.

⁵The terms $\propto \tilde{q}^2$, where $\tilde{q} = \bar{q} - q$ is $(D-4)$ -dimensional, lead to rational terms of type R_1 [6].

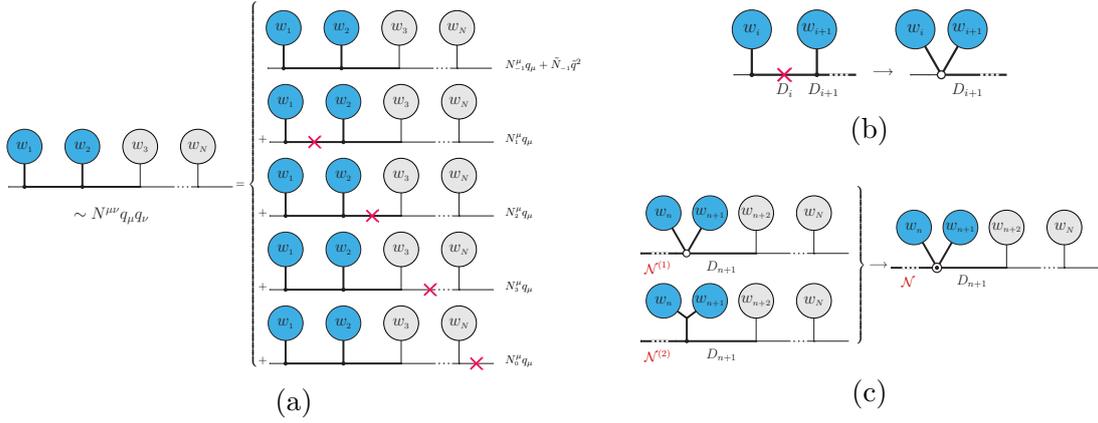


Figure 2: Creation of pinched topologies in on-the-fly reduction steps (a). Two dressed segments with a pinched propagator in between can be treated as one effective segment (b) of an $(N - 1)$ -point open loop. In many cases such an open loop can be absorbed into another $(N - 1)$ -point open loop stemming from a genuine $(N - 1)$ -point Feynman diagram (c).

set of available scalar propagators allows us to avoid numerical instabilities due to vanishing Gram determinants in four- and higher-point integrands and effectively shift them to triangles. Here we identified a simple kinematical configuration responsible for numerical instabilities due to a vanishing rank-two Gram determinant Δ and perform an analytical expansion in Δ [1]. In this way we completely avoid Gram determinant instabilities in our algorithm, which leads to excellent numerical stability (see section 3).

An important issue is the proliferation of pinched topologies, especially with many subsequent on-the-fly reductions steps. The solution we implement is an on-the-fly merging of open loops, which have the same topology⁶, and the same segments to be dressed in subsequent steps. This means that we sum the β_0 , β_k , helicity and q -tensor components of the open loops to be merged into a single open loop. All future dressing and reduction steps are then performed on the merged open loop which in general contains a set of contributions from different Feynman diagrams. In this way we can absorb most of the pinched contributions into lower-point Feynman diagrams (see Fig. 2 (c)) with the same topology and undressed segments. A requirement is that the segments left and right of the pinched propagator are already dressed, in which case the two can be treated as one effective segment (see Fig. 2 (b)).⁷ The on-the-fly merging requires the multiplication of each Feynman diagram with its colour factor contracted with the full Born amplitude in the beginning. In fact, we initialize each diagram d as

$$\mathcal{U}_0^{(d)}(h) = \mathcal{U}_0^{(d)}(\{h_1, \dots, h_N\}) = 2 \left(\sum_{\text{col}} \mathcal{M}_0^*(h) \mathcal{C}_1^{(d)} \right), \quad (2.3)$$

and perform the dressing, reduction and merging steps on these extended open loops. In

⁶A topology is defined as an ordered set $\{\bar{D}_0, \dots, \bar{D}_{N-1}\}$ of propagator denominators.

⁷The on-the-fly merging is also applied to open loops stemming from different Feynman diagrams, which have the same topology and last $N - k$ segments, but differ in the first n segments, after k dressing steps, giving a speed-up of about a factor 2 already without any on-the-fly reduction.

this way diagrams with different original colour factors can be merged. The initial interference with the Born, which is computed for all global helicity configurations h and hence all helicity configurations h_i of the individual segments, allows for an on-the-fly helicity summation, which significantly improves the CPU efficiency. We define the partially dressed and helicity summed open loop as well as an extended dressing step

$$\mathcal{U}_k(q, \{h_{k+1}, \dots, h_N\}) = \sum_{h_k} \mathcal{U}_{k-1}(q, \{h_k, \dots, h_N\}) S_k(q, h_k) \quad (2.4)$$

In this way, we reduce the number of dof of an open loop in the k -th dressing step by a factor equal to the number of helicity dof of the k -th segment. This new helicity treatment leads to an additional gain of a factor two–three in speed, depending on the process (see [1]).

3. Numerical stability

In this section we present numerical stability studies for the on-the-fly algorithm⁸ in OPENLOOPS (OL2) compared to the previous version of OPENLOOPS (OL1) using COLLIER or CUTTOOLS for the tensor integral reduction. The numerical accuracy of the double precision (DP) results is defined w.r.t. a benchmark derived with OL1+CUTTOOLS in quadruple precision (QP), $\mathcal{A} = \log_{10} |(\mathcal{W}_{\text{DP}} - \mathcal{W}_{\text{QP}}) / \min\{|\mathcal{W}_{\text{DP}}|, |\mathcal{W}_{\text{QP}}|\}|$. To estimate the intrinsic accuracy of the QP benchmark we use a so-called rescaling test [2].

In Fig. 3 and Fig. 4 we show the fraction of points with an accuracy $\mathcal{A} < \mathcal{A}_{\text{min}}$ plotted against \mathcal{A}_{min} for sample $2 \rightarrow 3$ and $2 \rightarrow 4$ processes respectively.⁹ While the results of OL1+CUTTOOLS feature the highest instability tails for all considered processes, we find that using COLLIER the probability of finding only a few correct digits goes down by one to three orders of magnitude, depending on the process. Using OL2 we observe improvements of one–two orders of magnitude w.r.t. OL1+COLLIER in many cases. For $2 \rightarrow 3$ processes, the stability of the on-the-fly algorithm is remarkably close to the QP benchmark and even superior for $\bar{t}\bar{t}g$ production. In the case of $2 \rightarrow 4$ processes OL1+COLLIER and OL2 are very close in the tail, both achieving excellent numerical stability.

4. Conclusion

We have presented a new approach for the automated calculation of scattering amplitudes at one loop. The key idea is that various operations, such as tensor reduction, helicity summation and diagram merging, can be performed *on-the-fly* during the open-loop recursion, exploiting the factorised structure of open loops in a systematic way. This reduces the complexity of certain operations in a very significant way. The employed integrand reduction method allows us to isolate Gram determinant instabilities in triangle topologies with a particular kinematic configuration and to cure them by means of simple analytic expansions, leading to an unprecedented level of numerical stability. This feature

⁸The final scalar integrals are evaluated with COLLIER.

⁹For each process a sample of 10^6 homogeneously distributed phase space points at $\sqrt{s} = 1$ TeV is taken. Infrared regions are excluded through cuts, $p_{i,T} > 50$ GeV and $\Delta R_{ij} > 0.5$, for massless final-state partons.

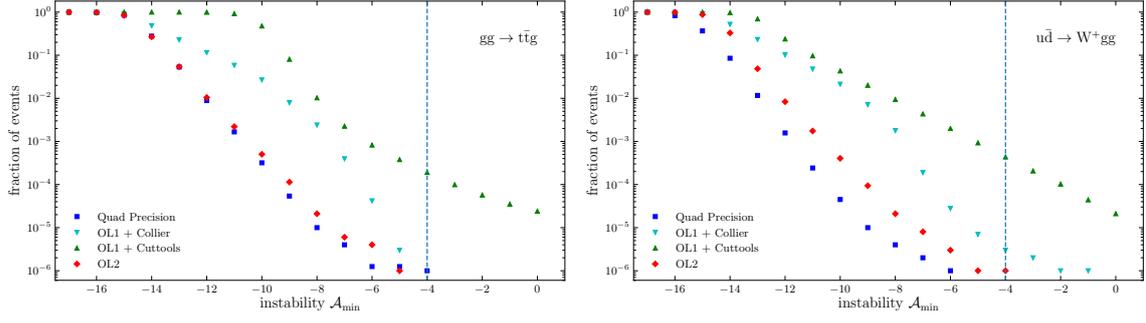


Figure 3: Stability distributions for sample 2 \rightarrow 3 processes.

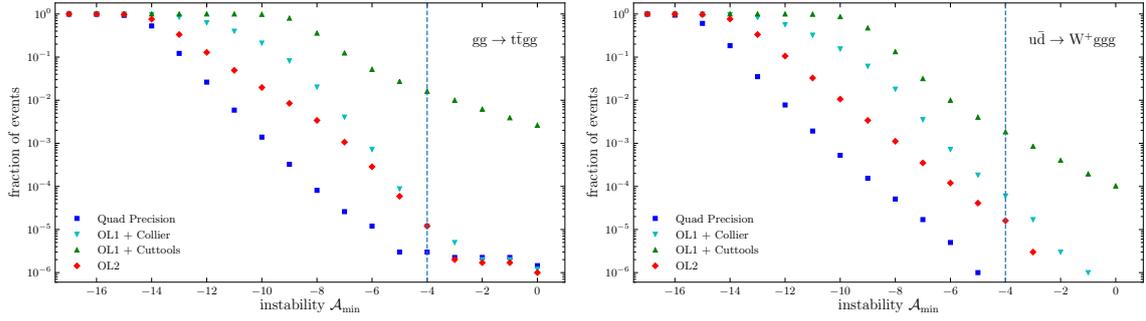


Figure 4: Stability distributions for sample 2 \rightarrow 4 processes.

is particularly attractive for the calculation of real-virtual contributions at NNLO. The new algorithm is fully automated and validated at NLO QCD and will become publicly available in the upcoming release of OPENLOOPS 2.

Acknowledgments

This research was supported in part by the Swiss National Science Foundation (SNF) under contracts PP00P2-128552 and BSCGI0-157722.

References

- [1] F. Buccioni, S. Pozzorini and M. Zoller, *On-the-fly reduction of open loops*, [1710.11452](https://arxiv.org/abs/1710.11452).
- [2] F. Cascioli, P. Maierhöfer and S. Pozzorini, *Scattering Amplitudes with Open Loops*, *Phys.Rev.Lett.* **108** (2012) 111601, [[1111.5206](https://arxiv.org/abs/1111.5206)].
- [3] F. Cascioli, J. M. Lindert, P. Maierhöfer and S. Pozzorini, *The OPENLOOPS one-loop generator*, Publicly available at <http://openloops.hepforge.org>.
- [4] A. Denner, S. Dittmaier and L. Hofer, *Collier: a fortran-based Complex One-Loop Library in Extended Regularizations*, *Comput. Phys. Commun.* **212** (2017) 220–238, [[1604.06792](https://arxiv.org/abs/1604.06792)].
- [5] 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](https://arxiv.org/abs/0711.3596)].
- [6] F. del Aguila and R. Pittau, *Recursive numerical calculus of one-loop tensor integrals*, *JHEP* **07** (2004) 017, [[hep-ph/0404120](https://arxiv.org/abs/hep-ph/0404120)].