Numerical implementation of integrand level reduction

Based on work with Costas Papadopoulos, Giuseppe Bevilacqua, Dhimiter Canko and Nikos Dokmetzoglou

Aris Spourdalakis

NCSR Demokritos

HOCTOOLS-II mini-workshop 27/10/2025

Monday 27th of October

Amplitude Construction and Reduction options

- Qgraph → symbolic manipulation, dimensionally regularized amplitudes → IBP: FIRE Kira or numerical pySecDec
- Numerical unitarity → dimensionally regularized amplitudes by gluing tree amplitudes in different integer dimensions → D_s (Abreu, Cordero, Ita, Page and Sotnikov, 2021)
- OpenLoops → Feynman graph → openning the loops → amplitudes in $d = 4$ → coefficients of tensor integrals (Pozzorini, Schär and Zoller, 2022)

What do we need for HELAC 2-loop?

3 steps:

- 1. Amplitude Construction (*Giuseppe B., Costas P. and Dhimiter "Jim" C.*)
- **2. Amplitude Reduction at 2-loops (My talk today)**
- 3. Do the integrals

Outline

- OPP Amplitude reduction at 1-loop
- 2-loop reduction analytic methodology
- Numerical Implementation
- Outlook

Amplitude reduction OPP @ 1-loop

Reduction is done at the *integrand* level. OPP (Ossola, Papadopoulos and Pittau, 2007) master formula a numerator at 1 loop:

$$
\begin{aligned}
N(q) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} d\left(i_0 i_1 i_2 i_3\right) + \tilde{d}\left(q; i_0 i_1 i_2 i_3\right) \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} c\left(i_0 i_1 i_2\right) + \tilde{c}\left(q; i_0 i_1 i_2\right) \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} b\left(i_0 i_1\right) + \tilde{b}\left(q; i_0 i_1\right) \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} a\left(i_0\right) + \tilde{a}\left(q; i_0\right) \prod_{i \neq i_0}^{m-1} D_i
\end{aligned}
$$

where $\tilde{d}$, $\tilde{c}$, $\tilde{b}$, $\tilde{a}$ are terms which vanish upon integration.

Amplitude reduction OPP: Fitting

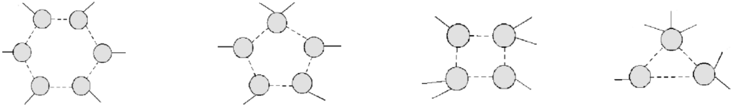
The system is solved by iteratively:

- Evaluate the numerator on values of k for which $D_i(k) = 0$, starting from the first line of the previous equation, where 4 propagators are put on shell
- This condition by default sets to zero all the rest of the terms, allowing us to calculate d and $\tilde{d}$
- Repeat for c, $\tilde{c}$ and so on until we fit all the coefficients

1 Loop Example

Example of a 6 point amplitude

$$\mathcal{A}(q) = \sum \underbrace{\frac{N_i^{(6)}(q)}{\bar{D}_{i_0}\bar{D}_{i_1}\cdots\bar{D}_{i_5}}} + \underbrace{\frac{N_i^{(5)}(q)}{\bar{D}_{i_0}\bar{D}_{i_1}\cdots\bar{D}_{i_4}}} + \underbrace{\frac{N_i^{(4)}(q)}{\bar{D}_{i_0}\bar{D}_{i_1}\cdots\bar{D}_{i_3}}} + \underbrace{\frac{N_i^{(3)}(q)}{\bar{D}_{i_0}\bar{D}_{i_1}\bar{D}_{i_2}}} + \cdots$$



- Calculate the coefficients of the OPP formula for each numerator
- The value of the loop momentum on the cut is given by CutTools, through hardcoded solutions (Ossola, Papadopoulos and Pittau, 2008)
- **This has been implemented!** $\longrightarrow$ **HELAC One Loop!**(Bevilacqua, Czakon, Garzelli, van Hameren, Kardos, Papadopoulos, Pittau and Worek, 2013)

2-Loop Amplitude: Algebraic Reduction

The fit will be more complex!
A generic 2-loop integrand can be written using the following scalar product set:

$$\{p_i \cdot p_j, k_i \cdot k_j, k_i \cdot p_j, k_i \cdot \eta_j\}$$

as well as any masses inside the loops. For the rest of the discussion we will ignore the case of massive loops.
The integrand can be written in the general form

$$\mathcal{R} = \frac{\mathcal{N}}{\mathcal{D}} = \frac{\sum_a c_a (z_1^{(a)})^{\beta_1} \ldots (z_{n_a}^{(a)})^{\beta_N}}{D_1 \ldots D_{N_p}} \tag{1}$$

where the z_i are any of the scalar products in the set.

2-Loop Amplitude reduction

Using $\bar{z}_i$ which are only the scalar products which cannot be eliminated by being written as linear combinations of D_i, know as irreducible scalar products (ISPs) or the transverse $k_i \cdot \eta_j$ and σ is any subset of $\{1, \ldots, N_p\}$ with m elements.

Write the numerator of the integrand level amplitude as follows:

Numerator Formula

$$\mathcal{N} = \sum_{m=0}^{N_p} \sum_{\sigma} \sum_{a} \bar{c}_a \prod_{i}^{N_{T+ISP}} (\bar{z}_i)^{\alpha_i} \prod_{j} D_j \tag{2}$$

where N_{T+ISP} is the number of ISP and transverse scalar products and $j \neq \sigma_i$ for all i (Bevilacqua, Canko, Papadopoulos and Spourdalakis, 2025)

2-Loop Amplitude reduction

Note that in the previous formula, the inverse propagators D_i do not nesserily have to appear in the numerator we are reducing, as propagators. We include them as long as the scalar products present in the numerator can be written as linear combinations of D_i. This broader set of is name *family*.

For each numerator, we

- Set all propagators of the family to zero and solve the equations that put all of them on shell simultaneously, AKA find *cut solutions*.
- Write a linear system for the coefficients $\mathbf{M} \cdot \vec{\bar{c}} = \vec{\mathcal{N}}$ where $\mathbf{M}$ is a matrix of all monomials $\bar{z}_i$ evaluated on different values of cut-solutions, $\vec{c}$ is all the $\bar{c}_a$ and $\vec{\mathcal{N}}$ is a vector of equal length with values of the numerator evaluated on the cut-solutions
- Solve the system of equations
- Subtract the result from the numerator

2-Loop Amplitude reduction

- Move to the next cut, where one less propagator is put on shell
- Do this till until fits can be performed with trivial polynomials
- Algebraic test that the reduced amplitude is equal to the original amplitude, known as the **N=N test**.
- The algebraic reduction is complete!

$$\mathcal{N} = P_{maxcut} + \sum_{i} P_{maxcut-1} D_i + \sum_{ij} P_{maxcut-2} D_i D_j + \ldots \qquad (3)$$

2-Loop Amplitude: Final Form

Final Form:

$$\mathcal{A} = \sum_i \bar{c}_i F_i \tag{4}$$

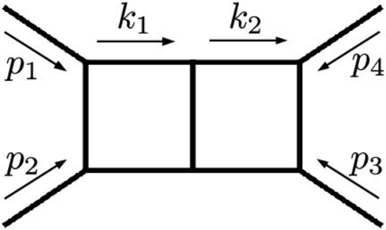
where:

$$F_i \equiv F_{a_1 \dots a_N} = \int d^d k \frac{\overbrace{(D_{m+1})^{a_{m+1}} \cdots (D_N)^{a_N}}^{ISP}}{\underbrace{(D_1)^{a_1} \cdots (D_m)^{a_m}}_{RSP}}$$

Use Integration by Parts (IBP) and/or other methods to further simplify the integral basis.

2-Loop reduction example

$2 \to 2$ gluon-topology



Maximal cut: 9 propagators on shell.

2-Loop reduction example: D-dimensions

Begin with 11 free parameters: 8 from the components of the two 4-momenta, and 3 $\mu_{11}, \mu_{22}, \mu_{12}$ the ϵ components of k_1^2, k_2^2 and $k_1 \cdot k_2$ respectively.
The maximal cut has 9 cut equations, we have a remainder of 2 free parameters, enough to find solutions which can constuct a full rank **M**

Results

We have completed an *analytic* Mathematica simulation of this fit. Agreement with known results: Caravel (Abreu, Cordero, Ita, Page and Sotnikov, 2021). Completed cut+fit for all subtologies with gluons and ghosts

How do we make the polynomials?

$$\mathcal{N} = P_{maxcut} + \sum_i P_{maxcut-1} D_i + \sum_{ij} P_{maxcut-2} D_i D_j + \ldots \qquad (5)$$

What's inside the P_i's? For this process, use (Zhang, 2012) with a twist:

- Use BasisDet with an educated guess for the monomial power.
- Obtain a basis containing the ISPs and μ_{ij}
- Use the on shell conditions at scalar product level (i.e $k_1.p_1$, $k_1.k_2$ etc, no specific representation for external or loop momenta) to solve for the μ_{ij} when possible and replace them
- Get a generic basis for this family containing a smaller basis, so called "true ISPs"

Example from double-box topology

- BasisDet Yields:

$$\mu_{11}^2,\ \mu_{11},\ \mu_{12}\mu_{11},$$
$$k_1\eta\,\mu_{11},\ k_2\eta\,\mu_{11},$$
$$\mu_{22},\ \mu_{22}^2,\ \mu_{12},$$
$$\mu_{12}\mu_{22},\ \mu_{12}^2,\ k_1\eta,$$
$$k_2\eta,\ \mu_{22}k_2\eta,\ \mu_{12}k_2\eta$$

- Cut Equations are:

$$\mu_{11} \to -\frac{4s\,(k_1\eta)^2 + 4t\,(k_1\eta)^2 + s\,t}{4\,(s+t)},$$
$$\mu_{22} \to -\frac{4s\,(k_2\eta)^2 + 4t\,(k_2\eta)^2 + s\,t}{4\,(s+t)},$$
$$\mu_{12} \to -\frac{4s\,k_1\eta\,k_2\eta + 4t\,k_1\eta\,k_2\eta - s\,t}{4\,(s+t)}$$

- True ISPs:

$$(k_1\eta)^4,\ (k_1\eta)^3,\ k_2\eta\,(k_1\eta)^3,$$
$$(k_2\eta)^2\,(k_1\eta)^2,\ (k_1\eta)^2,$$
$$k_2\eta\,(k_1\eta)^2,\ (k_2\eta)^3\,k_1\eta,$$
$$(k_2\eta)^2\,k_1\eta,\ k_1\eta,\ k_2\eta\,k_1\eta,$$
$$(k_2\eta)^4,\ (k_2\eta)^3,\ (k_2\eta)^2,\ k_2\eta$$

Example from double-box topology

In principle, the polynomials should be equivalent, since we are simply using the on-shell conditions of the cut to eliminate variables.

- Using the "true ISP" polynomials work for all the tested examples, full N=N completed ✓
- When using the "raw" BasisDet polynomials, preliminary results show that for some of the numerators in the 7-cut the N=N test fails.

Numerical Reduction Implementation

- Started with a hardcoded set of cut solutions, dependent on the form of the external momenta.
- Moved to cut equations obtained for general external momenta using the iterative Levenberg–Marquardt method
- Used Lapack (Anderson, Bai, Bischof, Blackford, Demmel, Dongarra, Du Croz, Greenbaum, Hammarling, McKenney and Sorensen, 1999) compiled in quadraple precision both for cut-equation solving and for the polynomial fitting
- Using a Hybrid approach 4*d*-terms from HELAC 2 Loop and term proportional to μ_{ij} exported to f90 from Mathematica

Results

- Algebraic reduction completed analytically for all $2 \to 2$ topologies (both in $d = 4$ and in $d = 4 - 2\epsilon$) ✓
- Numerical proof of concept results for Leading Color $gg \to gg$ (with gluons and ghosts) ✓
- Working in Quadruple Precision due to loss of precision at each stage in the reduction
- Numerical (algebraic) Reduction of a single numerator takes $\sim$ seconds
- Much more optimization is possible (and necessary)

Outlook

- Implement the fully D-dimensional fit @ 1 and 2 loops numerically
- Numerical D-dimensional reduction at 2-loops for all $2 \to 2$ topologies
- $2 \to 3$ topology reduction

Outlook-Further Future

Generalize for further topologies!

Thank you for listening!

The research project was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the 2nd Call for H.F.R.I. Research Projects to support Faculty Members and Researchers (Project Number: 2674).

H.F.R.I.
Hellenic Foundation for Research & Innovation

Abreu S, Cordero FF, Ita H, Page B, Sotnikov V. 2021. Leading-color two-loop qcd corrections for three-jet production at hadron colliders. *Journal of High Energy Physics* **2021**. ISSN 1029-8479. URL http://dx.doi.org/10.1007/JHEP07(2021)095

Anderson E, Bai Z, Bischof C, Blackford S, Demmel J, Dongarra J, Du Croz J, Greenbaum A, Hammarling S, McKenney A, Sorensen D. 1999. *LAPACK Users' Guide*. Philadelphia, PA: Society for Industrial and Applied Mathematics, third edition. ISBN 0-89871-447-8 (paperback).

Bevilacqua G, Canko D, Papadopoulos C, Spourdalakis A. 2025. Towards numerical two-loop integrand reduction. URL https://arxiv.org/abs/2506.07231

Bevilacqua G, Czakon M, Garzelli MV, van Hameren A, Kardos A, Papadopoulos CG, Pittau R, Worek M. 2013. HELAC-NLO. *Comput. Phys. Commun.* **184**: 986–997.

Frellesvig HA. 2014. *Generalized Unitarity Cuts and Integrand Reduction at Higher Loop Orders*. Ph.D. thesis, Copenhagen U.

Ossola G, Papadopoulos CG, Pittau R. 2007. Reducing full one-loop amplitudes to scalar integrals at the integrand level. *Nucl. Phys. B* **763**: 147–169.

Ossola G, Papadopoulos CG, Pittau R. 2008. CutTools: A Program implementing the OPP reduction method to compute one-loop amplitudes. *JHEP* **03**: 042.

Pozzorini S, Schär N, Zoller MF. 2022. Two-loop tensor integral coefficients in openloops. *Journal of High Energy Physics* **2022**. ISSN 1029-8479.
URL http://dx.doi.org/10.1007/JHEP05(2022)161

Zhang Y. 2012. Integrand-level reduction of loop amplitudes by computational algebraic geometry methods. *Journal of High Energy Physics* **2012**. ISSN 1029-8479.
URL http://dx.doi.org/10.1007/JHEP09(2012)042

2-Loop reduction example: 4-dimensions

- Some monomials which form the matrix **M** have the same values for different cut solution sets. Danger that we could get $Rank[\mathbf{M}] < Lenght[\vec{c}]$
  $\longrightarrow$ **Need to use full set of solutions, treating each of them independently.**
- How can we derive all the gram determinant relations that hold at each step of the recursion?
  $\longrightarrow$ Use BasisDet for the Ansatz
  $\longrightarrow$ **Write directly in terms of 4d** k_i **components**.
  Not unique, different basis lead to a different form of the cut-solutions!

2 loop reduction interesting questions

- What goes into to constructing the polynomial Ansatz? One option is BasisDet (Zhang, 2012). Is there some apriori way of determining it for each topology/subtopology apriori? What is the "correct" power for each monomial at each subtopology?
- Structure of the solution space for the cut solutions (Algebraic Geometry question) (Frellesvig, 2014)
- Are there any integrand level symmetries on the cuts which we can take into account to further simplify?

More specific 4-d questions

But Frellesvig (2014) mention issues with lower topology cuts:

- A) Some sub-topologies having divergences that require careful subtraction
- B) Other sub-topologies where $\mathrm{Rank}[\mathbf{M}] < \mathrm{Length}[\vec{c}]$, i.e not enough independent equations to fit the polynomial Ansatz.

Both currently under investigation with our approach

More specific question: Ansatz

The polynomial we use needs to properly capture the functional behaviour of the numerator

Can have

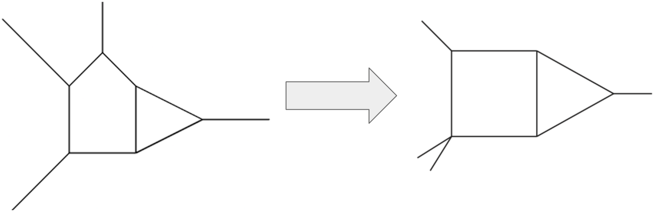
- Full rank $\mathrm{Rank}[\mathbf{M}] = \mathrm{Length}[\overrightarrow{c}]$ and no solution (i.e. incomplete Ansatz)

Depends on the relation between the Ansatz and the numerator $\rightarrow$ **Not just a matter of the rank of the matrix!**

Open Question: Can we know the Ansatz apriori from the topological family? Seems difficult...

Too much Freedom?

Penta-triangle and subtologies



Examining the penta-triangle and its 6-cut subtologies

- Do not get a full rank for **M** for *any* of the 6-cut subtopologies in 4-d ...
- We can still fit all of them!

More specific answers

- A) Don't seem to appear (so far) in the way we construct the amplitude
- B) Some sub-topologies do have $\mathrm{Rank}[\mathbf{M}] < \mathrm{Length}[\vec{c}]$. This does not lead to an unsolvable system!

Need to have an Ansatz that properly describes the Numerator in question!

More specific (partial) Answers: Ansatz

Can have

- $\mathrm{Rank}[\mathbf{M}] < \mathrm{Length}[\vec{c}]$ and get a correct solution!

Depends on the relation between the Ansatz and the numerator.
The polynomial we use needs to properly capture the functional behaviour of the numerator.
Open Question: Can we know the Ansatz apriori from the topological family?
We don't need to be exact, *just know enough to get a solution!*

Work is ongoing for this more difficult topology

Still need to do the lower subtopology cuts numerically