PAUL SCHERRER INSTITUT

Swiss National
Science Foundation

Status of OpenLoops at Two Loops

Natalie Schär

27. January 2023

in collaboration with
S. Pozzorini and M. F. Zoller

based on
JHEP05(2022)161 (arXiv:2201.11615)

Theory Predictions in Particle Physics

In particle theory observables are computed by **Monte Carlo Tools** (e.g. SHERPA [Gleisberg, Hoeche, Krauss, Schonherr, Schumann, Siegert et al.], POWHEG [Alioli, Nason, Oleari, Re], HELAC-NLO [Bevilacqua, Czakon, Garzelli, van Hameren, Kardos, Papadopoulos et al.], MADGRAPH [Alwall, Frederix, Frixione, Hirschi, Maltoni, Mattelaer et al.], Herwig++ [Bellm, Gieseke, Grellscheid, Plätzer, Rauch], etc.)

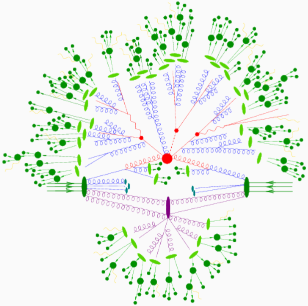
→ calculation **factorizes** into various **perturbative** and **non-perturbative** components

→ development and implementation of each component involves highly complex methods and algorithms

---

Components include:

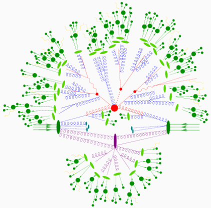
- PDFs ●
- hard scattering process ●
- parton showers ●
- hadronization ●



[Schälicke, Gleisberg, Höche, Schumann, Winter, Krauss, Soff]

OpenLoops

- OpenLoops is a numerical tool providing hard scattering amplitudes to Monte Carlo simulations.
- All components to NLO fully automated in OpenLoops for QCD and EW corrections to the SM.

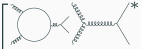


[Schälicke, Gleisberg, Höche, Schumann, Winter, Krauss, Soff]

OpenLoops constructs helicity and color summed scattering probability densities $\mathcal{W}_{LL} = \sum_h \sum_{\text{col}} |\bar{\mathcal{M}}_L(h)|^2$ for $L = 0, 1$ and $\mathcal{W}_{0L} = \sum_h \sum_{\text{col}} 2\,\text{Re}\left[\bar{\mathcal{M}}_L(h)\,\bar{\mathcal{M}}_0^*(h)\right]$ for $L = 1$ from L-loop matrix elements $\bar{\mathcal{M}}_L$.

Example:

$$\mathcal{W}_{01} = \sum_h \sum_{\text{col}} 2\text{Re}\left[\quad\right.$$

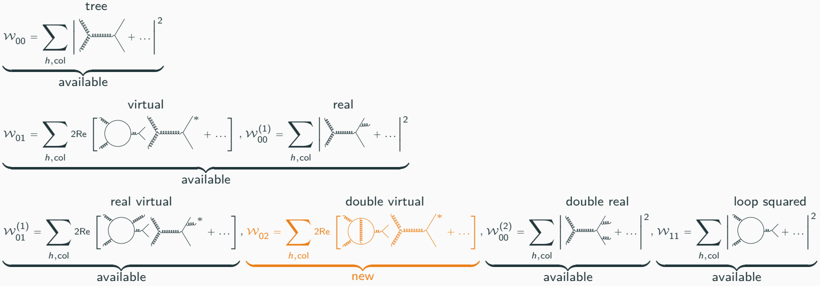


$$\left. + \ldots \right]$$

Goal: automation at NNLO

Automation at NNLO

The public OpenLoops [Buccioni, Lang, Lindert, Maierhöfer, Pozzorini, Zhang, Zoller] already delivers some components to NNLO:

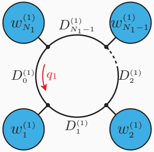


- OpenLoops is already being used in NNLO calculations in particular for the real virtual components in e.g. MATRIX [Grazzini, Kallweit, Wiesemann], NNLOJET [Gehrmann-De Ridder, Gehrmann, Glover, Huss, Walker], McMule [Banerjee, Engel, Signer, Ulrich].
- **NNLO in OpenLoops: require double virtual**

Components to NLO Calculations

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{1,\Gamma} =$



$$= \underbrace{C_{1,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \underbrace{\frac{\mathcal{N}(q_1)}{\mathcal{D}(\bar{q}_1)}}_{\substack{\text{4-dim numerator,}\\ \text{(D-dim denominator)}}} = C_{1,\Gamma} \sum_r \underbrace{\mathcal{N}_{\mu_1\cdots\mu_r}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \frac{q_1^{\mu_1}\cdots q_1^{\mu_r}}{\mathcal{D}(\bar{q}_1)}}_{\text{tensor integral}}$$

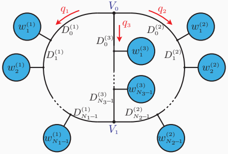
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim → OpenLoops algorithm
  [van Hameren; Cascioli, Maierhöfer, Pozzorini; Buccioni, Lang, Lindert, Maierhöfer, Pozzorini, Zhang, Zoller]
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms→
  $\mathbf{R}\bar{\mathcal{M}}_{1,\Gamma} = \mathcal{M}_{1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,1,\Gamma}$ [Ossola, Papadopoulos, Pittau]
- Reduction and evaluation of tensor integrals → On-the-fly reduction
  [Buccioni, Pozzorini, Zoller], Collier [Denner, Dittmaier, Hofer], OneLoop [van Hameren]

Components to NNLO Calculations

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$



$$= \underbrace{\mathcal{C}_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1, q_2)}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\substack{\text{4-dim numerator,}\\ \text{(D-dim denominator)}}} = \mathcal{C}_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \, \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \, q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\text{tensor integral}}$$

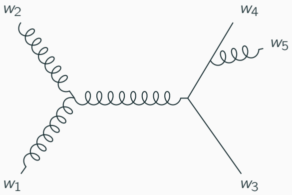
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim → fully general algorithm, implementation complete for QED and QCD
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms→
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
  currently working on implementation and validation
- Reduction and evaluation of tensor integrals → small in-house library for test purposes, general solution: future projects

Outline

Tree Level Algorithm

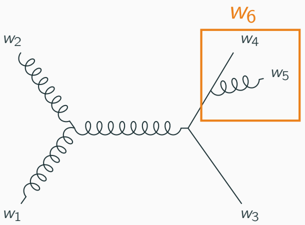
OpenLoops Tree Level Algorithm: Example



input: external wavefunctions

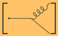
w_1, w_2, w_3, w_4, w_5

OpenLoops Tree Level Algorithm: Example

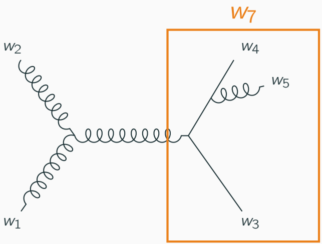


Combine w_4, w_5 into subtree w_6:

$w_6^{\gamma} =$  $^{\gamma}_{\alpha\beta}\, w_4^{\alpha} w_5^{\beta}$

 $^{\gamma}_{\alpha\beta} =$ vertex + propagator, universal process-independent Feynman rule

OpenLoops Tree Level Algorithm: Example



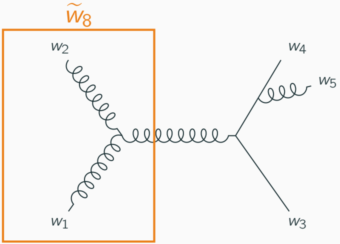
Add next external leg:

$$w_6^\gamma = [\;]^{\gamma}_{\alpha\beta}\, w_4^\alpha\, w_5^\beta$$

$$w_7^\gamma = [\;]^{\gamma}_{\alpha\beta}\, w_3^\alpha\, w_6^\beta$$

$[\;]^{\gamma}_{\alpha\beta} =$ vertex + propagator, universal process-independent Feynman rule

OpenLoops Tree Level Algorithm: Example



same on the other side:

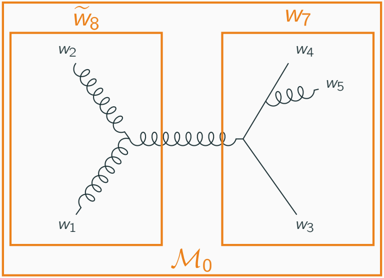
$$w_6^\gamma = \left[\;\;\right]^\gamma_{\alpha\beta} w_4^\alpha w_5^\beta$$

$$w_7^\gamma = \left[\;\;\right]^\gamma_{\alpha\beta} w_3^\alpha w_6^\beta$$

$$\widetilde{w}_8^\gamma = \left[\;\;\right]^\gamma_{\alpha\beta} w_1^\alpha w_2^\beta$$

$\left[\;\;\right]^\gamma_{\alpha\beta} =$ vertex, universal process-independent Feynman rule

OpenLoops Tree Level Algorithm: Example



combine to full diagram:

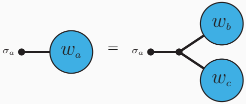
$$w_6^\gamma = \left[\quad \right]^\gamma_{\alpha\beta} w_4^\alpha w_5^\beta$$
$$w_7^\gamma = \left[\quad \right]^\gamma_{\alpha\beta} w_3^\alpha w_6^\beta$$
$$\widetilde{w}_8^\gamma = \left[\quad \right]^\gamma_{\alpha\beta} w_1^\alpha w_2^\beta$$
$$\mathcal{M}_0 = \left[\quad \right]_{\alpha\beta} w_7^\alpha w_8^\beta$$

$\left[\quad \right]_{\alpha\beta} =$
universal process-independent
Feynman rule

OpenLoops Tree Level Algorithm

Recursively construct subtrees starting from external wavefunctions:

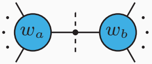
$$w_a^{\sigma_a}(k_a, h_a) \;=\;$$



$$= \underbrace{\frac{X^{\sigma_a}_{\sigma_b \sigma_c}(k_b, k_c)}{k_a^2 - m_a^2}}_{\text{model-dependent}} \underbrace{w_b^{\sigma_b}(k_b, h_b)\, w_c^{\sigma_c}(k_c, h_c)}_{\text{process-dependent}}$$

Then contract into full diagram:

$$\mathcal{M}_{0,\Gamma}(h) \;=\;$$



$$= C_{0,\Gamma} \cdot w_a^{\sigma_a}(k_a, h_a)\, \delta_{\sigma_a \sigma_b} \widetilde{w}_b^{\sigma_b}(k_b, h_b)$$

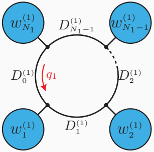
- diagrams constructed using **universal Feynman rules**
- identical subtrees are **recycled** in multiple tree and loop diagrams

One Loop Algorithm

OpenLoops Algorithm at One Loop

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{1,\Gamma} =$

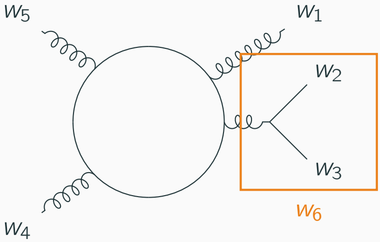


$$= \underbrace{C_{1,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \underbrace{\frac{\mathcal{N}(q_1)}{\mathcal{D}(\bar{q}_1)}}_{\substack{\text{4-dim numerator,}\\ \text{(D-dim denominator)}}} = C_{1,\Gamma} \sum_r \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r}}{\mathcal{D}(\bar{q}_1)}}_{\text{tensor integral}}$$

Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim → OpenLoops algorithm
  [van Hameren; Cascioli, Maierhöfer, Pozzorini; Buccioni, Lang, Lindert, Maierhöfer, Pozzorini, Zhang, Zoller]
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms→
  $\mathbf{R}\bar{\mathcal{M}}_{1,\Gamma} = \mathcal{M}_{1,\Gamma} + \mathcal{M}_{0,1,\Gamma}^{(\mathrm{CT})}$ [Ossola, Papadopoulos, Pittau]
- Reduction and evaluation of tensor integrals → On-the-fly reduction
  [Buccioni, Pozzorini, Zoller], Collier [Denner, Dittmaier, Hofer], OneLoop [van Hameren]

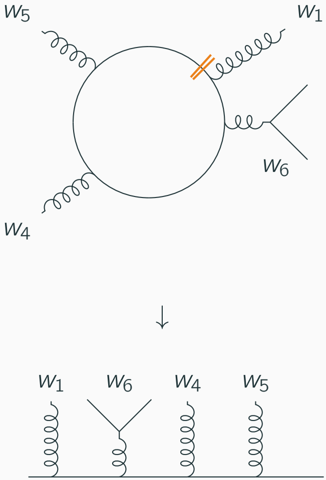
One Loop Algorithm: Example



External subtrees constructed in tree level algorithm (together with tree diagrams):

$w_2, w_3 \rightarrow w_6$

One Loop Algorithm: Example

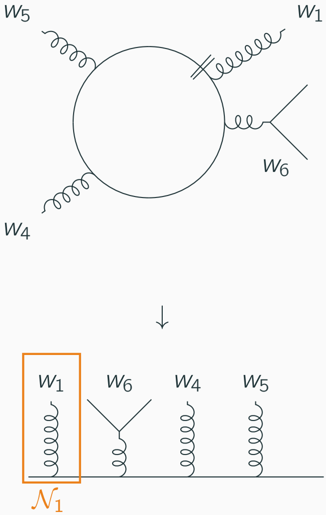


Open Loop:

Diagram factorizes into chain of segments: $\mathcal{N} = S_1 \cdots S_N$

segment = loop vertex + loop propagator + external subtree(s)

One Loop Algorithm: Example

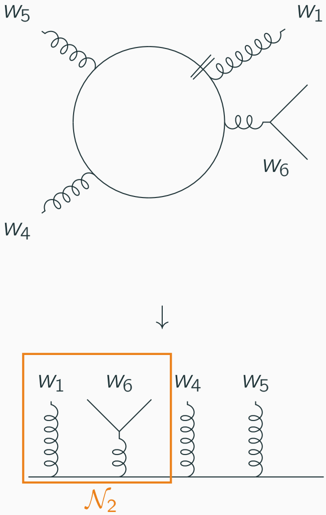


Construct first segment S_1 attaching the external subtree w_1.

$\mathcal{N}_0 = \mathbb{1}$
$\mathcal{N}_1 = \mathcal{N}_0 \cdot S_1(w_1)$

segment = loop vertex + loop propagator + external subtree(s)

One Loop Algorithm: Example

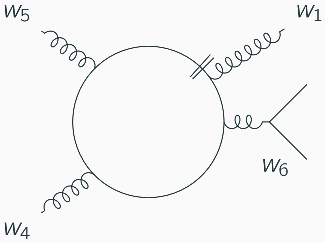


Add second segment attaching the subtree w_6.

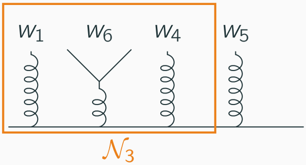
$$\mathcal{N}_0 = \mathbb{1}$$
$$\mathcal{N}_1 = \mathcal{N}_0 \cdot S_1(w_1)$$
$$\mathcal{N}_2 = \mathcal{N}_1 \cdot S_2(w_6)$$

segment = loop vertex + loop propagator + external subtree(s)

One Loop Algorithm: Example



$\downarrow$

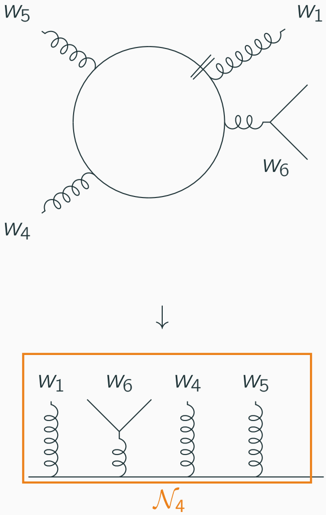


Add third segment.

$\mathcal{N}_0 = \mathbb{1}$
$\mathcal{N}_1 = \mathcal{N}_0 \cdot S_1(w_1)$
$\mathcal{N}_2 = \mathcal{N}_1 \cdot S_2(w_6)$
$\mathcal{N}_3 = \mathcal{N}_2 \cdot S_3(w_4)$

segment = loop vertex + loop propagator + external subtree(s)

One Loop Algorithm: Example

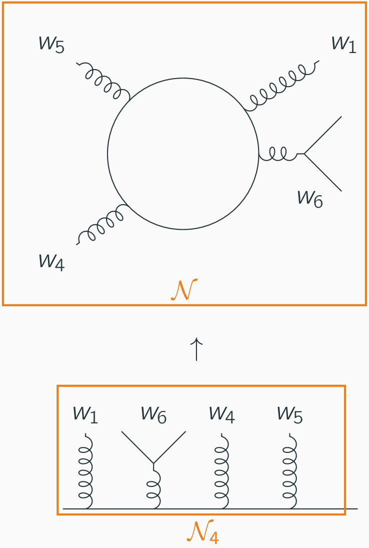


Add last segment.

$$\mathcal{N}_0 = \mathbb{1}$$
$$\mathcal{N}_1 = \mathcal{N}_0 \cdot S_1(w_1)$$
$$\mathcal{N}_2 = \mathcal{N}_1 \cdot S_2(w_6)$$
$$\mathcal{N}_3 = \mathcal{N}_2 \cdot S_3(w_4)$$
$$\mathcal{N}_4 = \mathcal{N}_3 \cdot S_4(w_5)$$

segment = loop vertex + loop propagator + external subtree(s)

One Loop Algorithm: Example



Close the loop (contract open Lorentz/spinor indices).

$$\mathcal{N}_0 = \mathbb{1}$$
$$\mathcal{N}_1 = \mathcal{N}_0 \cdot S_1(w_1)$$
$$\mathcal{N}_2 = \mathcal{N}_1 \cdot S_2(w_6)$$
$$\mathcal{N}_3 = \mathcal{N}_2 \cdot S_3(w_4)$$
$$\mathcal{N}_4 = \mathcal{N}_3 \cdot S_4(w_5) = {\mathcal{N}_4}_{\beta_0}^{\beta_N}$$

$$\mathcal{N} = Tr({\mathcal{N}_4}_{\beta_0}^{\beta_N})$$

OpenLoops One Loop Algorithm

One Loop Amplitude:

$$\mathcal{M}_{1,\Gamma} = C_{1,\Gamma} \int \mathrm{d}\bar{q} \, \frac{\mathrm{Tr}[\mathcal{N}(q)]}{D_0 D_1 \cdots D_{N_1-1}} =$$

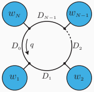
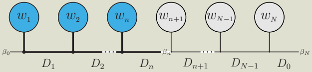


Diagram is cut open resulting in a chain, which **factorizes** into segments:

$$\mathcal{N}_n(q) = \prod_{a=1}^{n} S_a(q) =$$

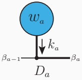


Chain is constructed recursively, recursion step: $\mathcal{N}_n = \mathcal{N}_{n-1} \cdot S_n$.

Implemented at level of tensor coefficients in $\mathcal{N} = \mathcal{N}_{\mu_1 \cdots \mu_r} q_1^{\mu_1} \cdots q_1^{\mu_r}$.

Segment = vertex + propagator + subtree(s)

$$\Big[S_a(q)\Big]^{\beta_a}_{\beta_{a-1}} =$$



$$= \Big[Y_{\sigma_a} + Z_{\sigma_a,\nu} q^{\nu}\Big]^{\beta_a}_{\beta_{a-1}} w_a^{\sigma_a}(k_a)$$

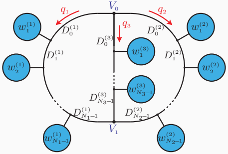
Exploit factorization to construct 1l diagrams from universal process-independent building blocks.

Two Loop Algorithm

OpenLoops Algorithm at Two Loops

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$



$$= \underbrace{\mathcal{C}_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1\,,\,q_2)}{\mathcal{D}(\bar{q}_1\,,\,\bar{q}_2)}}_{\substack{\text{4-dim numerator,}\\ \text{(D-dim denominator)}}} = \mathcal{C}_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1\cdots\mu_r\nu_1\cdots\nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \, \frac{q_1^{\mu_1}\cdots q_1^{\mu_r}\, q_2^{\nu_1}\cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1\,,\,\bar{q}_2)}}_{\text{tensor integral}}$$

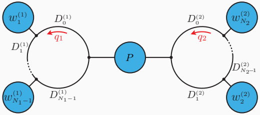
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
- Reduction and evaluation of tensor integrals

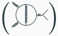
Two Loop Algorithm: Reducible Diagrams

Distinguish irreducible () and reducible (, ) diagrams.

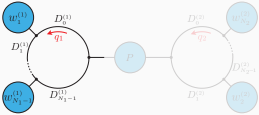
Exploit numerator factorization:

$\mathcal{N}(q_1, q_2) =$ 

$$= \underbrace{\left[\mathcal{N}^{(1)}(q_1)\right]^{\alpha_1}}_{\text{chain 1}} \underbrace{P_{\alpha_1 \alpha_2}}_{\text{bridge}} \underbrace{\left[\mathcal{N}^{(2)}(q_2)\right]^{\alpha_2}}_{\text{chain 2}}$$

Distinguish irreducible () and reducible () diagrams.

Exploit numerator factorization:

$\mathcal{N}(q_1, q_2) =$ 

$= \underbrace{\left[\mathcal{N}^{(1)}(q_1)\right]^{\alpha_1}}_{\text{chain 1}} \underbrace{P_{\alpha_1\alpha_2}}_{\text{bridge}} \underbrace{\left[\mathcal{N}^{(2)}(q_2)\right]^{\alpha_2}}_{\text{chain 2}}$

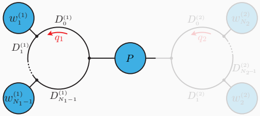
1. Construct chain 1 using extension of one-loop algorithm, perform first loop integration.

$$\mathcal{N}^{(1)}_n = \mathcal{N}^{(1)}_{n-1} S^{(1)}_n, \qquad \mathcal{N}^{(1)}_0 = \mathbb{1}, \qquad \left[\mathcal{M}^{(1)}\right]^{\alpha_1} = \int \mathrm{d}\bar{q}_1 \frac{\mathrm{Tr}\left[\mathcal{N}^{(1)}_{N_1}(q_1)\right]^{\alpha_1}}{\mathcal{D}^{(1)}(\bar{q}_1)}$$

Two Loop Algorithm: Reducible Diagrams

Distinguish irreducible () and reducible (, ) diagrams.

Exploit numerator factorization:

$\mathcal{N}(q_1, q_2) =$ 

$$= \underbrace{\left[\mathcal{N}^{(1)}(q_1)\right]^{\alpha_1}}_{\text{chain 1}} \underbrace{P_{\alpha_1 \alpha_2}}_{\text{bridge}} \underbrace{\left[\mathcal{N}^{(2)}(q_2)\right]^{\alpha_2}}_{\text{chain 2}}$$

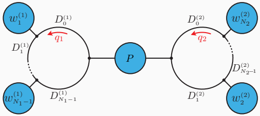
1. Construct chain 1 using extension of one-loop algorithm, perform first loop integration.
2. Connect bridge using tree algorithm
   → treat first loop as external "subtree".

$$P_n = P_{n-1} S_n^{(B)}(w_n^{(B)}), \qquad w_0^{(B)} = \left[\mathcal{M}^{(1)}\right]^{\alpha_1}, \qquad P_{-1} = \mathbb{1}$$

Two Loop Algorithm: Reducible Diagrams

Distinguish irreducible () and reducible () diagrams.

Exploit numerator factorization:

$\mathcal{N}(q_1, q_2) =$ 

$$= \underbrace{\left[\mathcal{N}^{(1)}(q_1)\right]^{\alpha_1}}_{\text{chain 1}} \underbrace{P_{\alpha_1 \alpha_2}}_{\text{bridge}} \underbrace{\left[\mathcal{N}^{(2)}(q_2)\right]^{\alpha_2}}_{\text{chain 2}}$$

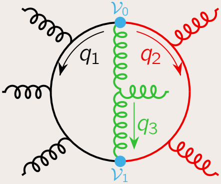
1. Construct chain 1 using extension of one-loop algorithm, perform first loop integration.
2. Connect bridge using tree algorithm
   $\rightarrow$ treat first loop as external "subtree".
3. Construct chain 2 using extension of one-loop algorithm
   $\rightarrow$ treat first loop + bridge as external "subtree".

$$\mathcal{N}_n^{(2)} = \mathcal{N}_{n-1} S_n^{(2)}(w_n^{(2)}), \qquad w_1^{(2)} = \left[\mathcal{M}^{(1)}\right]^{\alpha_1} P_{\alpha_1 \alpha_2}, \qquad \mathcal{N}_0^{(2)} = \mathbb{1}$$

Two Loop Algorithm: Irreducible Diagrams

Two-loop numerator factorizes:

$$\mathcal{N}(q_1, q_2) = \mathcal{N}^{(1)}(q_1)\ \mathcal{N}^{(2)}(q_2)\ \mathcal{N}^{(3)}(q_3)\ \mathcal{V}_0(q1, q2)\ \mathcal{V}_1(q1, q2)\big|_{q_3 \to -(q_1+q_2)}$$
$$\mathcal{N}^{(i)}(q_i) = S_0^{(i)}(q_i)\ S_1^{(i)}(q_i) \cdots S_{N_i-1}^{(i)}(q_i)$$



Building blocks $\mathcal{K}_n$ for algorithm:

- $\mathcal{N}^{(1)}, \mathcal{N}^{(2)}, \mathcal{N}^{(3)}$ 3 chains
- $s_a^{(1)}, s_a^{(2)}, s_a^{(3)}$ their segments
- $\mathcal{V}_0, \mathcal{V}_1$ vertices connecting chains
- $\mathcal{U}_0 = 2\sum_{\text{col}} \mathcal{C}\mathcal{M}_0^*$ Born and color

⇒ Construct Born-loop interference recursively from building blocks:

$$\mathcal{U}_n = \mathcal{U}_{n-1}\mathcal{K}_n, \quad \mathcal{K}_n \in \{\mathcal{U}_0, \mathcal{N}^{(i)}, s_a^{(i)}, \mathcal{V}_j\}$$

Factorization results in freedom of choice for two-loop algorithm.

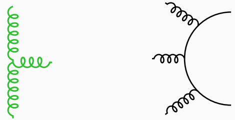
- CPU cost ∼ # multiplications
- determine most efficient variant through cost simulation

Two Loop Algorithm: Irreducible Diagrams

1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.

$$\mathcal{N}_n^{(3)}(q_3) = \mathcal{N}_{n-1}^{(3)} S_n^{(3)}, \qquad \mathcal{N}_0^{(3)} = \mathbb{1}$$

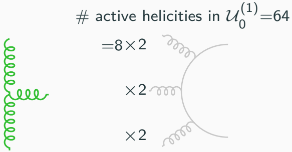
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition.

$$\mathcal{U}_n^{(1)} = \mathcal{U}_{n-1}^{(1)} S_n^{(1)}, \qquad \mathcal{U}_0^{(1)} = 2\sum_{col} C\mathcal{M}_0^*$$

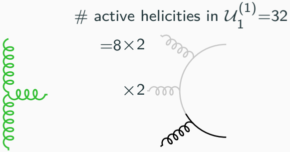
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0{=}2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex.

$$\mathcal{U}_n^{(1)}(h_{n+1}, h_{n+2}, \ldots) = \sum_{h_n} \mathcal{U}_{n-1}^{(1)}(h_n, h_{n+1}, h_{n+2} \ldots) S_n^{(1)}(h_n), \qquad \mathcal{U}_0^{(1)} = \mathcal{U}_0^{(1)}(h_1, h_2, \ldots, h_{N_1+N_2+N_3})$$

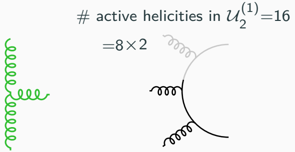
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0{=}2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex.

$$\mathcal{U}_n^{(1)}(h_{n+1}, h_{n+2}, \ldots) = \sum_{h_n} \mathcal{U}_{n-1}^{(1)}(h_n, h_{n+1}, h_{n+2} \ldots) S_n^{(1)}(h_n), \qquad \mathcal{U}_0^{(1)} = \mathcal{U}_0^{(1)}(h_1, h_2, \ldots, h_{N_1+N_2+N_3})$$

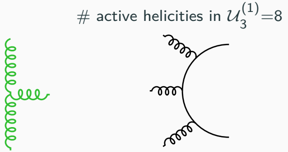
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex.

$$\mathcal{U}_n^{(1)}(h_{n+1}, h_{n+2}, \ldots) = \sum_{h_n} \mathcal{U}_{n-1}^{(1)}(h_n, h_{n+1}, h_{n+2} \ldots) S_n^{(1)}(h_n), \qquad \mathcal{U}_0^{(1)} = \mathcal{U}_0^{(1)}(h_1, h_2, \ldots, h_{N_1+N_2+N_3})$$

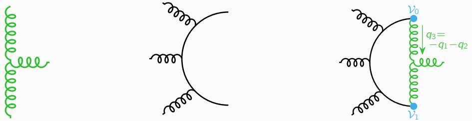
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex. **Large # of helicities summed in this step (one-loop complexity).**

$$\mathcal{U}_n^{(1)}(h_{n+1}, h_{n+2}, \ldots) = \sum_{h_n} \mathcal{U}_{n-1}^{(1)}(h_n, h_{n+1}, h_{n+2} \ldots) S_n^{(1)}(h_n), \qquad \mathcal{U}_0^{(1)} = \mathcal{U}_0^{(1)}(h_1, h_2, \ldots, h_{N_1+N_2+N_3})$$

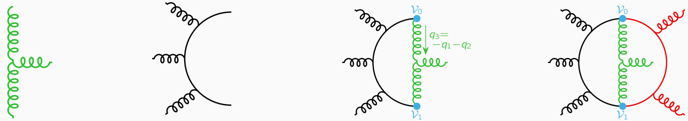
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex. **Large # of helicities summed in this step (one-loop complexity).**
3. Attach $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(3)}(q_3)$ first to $\mathcal{V}_1$, then to $\mathcal{V}_0$, sum helicities of $\mathcal{N}^{(3)}(q_3)$, $\mathcal{V}_1$, $\mathcal{V}_0$.

$$[\mathcal{U}^{(13)}]^{\beta^{(2)}_{N_2}}_{\beta^{(2)}_0} = [\mathcal{U}^{(1)}]^{\beta^{(1)}_{N_1}}_{\beta^{(1)}_0} [\mathcal{N}^{(3)}]^{\beta^{(3)}_{N_3}}_{\beta^{(3)}_0} \left[\mathcal{V}_0(q1,\, q2)\right]^{\beta^{(1)}_0 \beta^{(2)}_0 \beta^{(3)}_0} \left[\mathcal{V}_1(q1,\, q2)\right]_{\beta^{(1)}_{N_1} \beta^{(2)}_{N_2} \beta^{(3)}_{N_3}} \Big|_{q_3 \to -(q_1+q_2)}$$

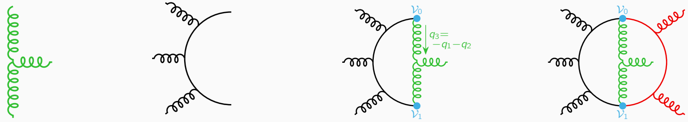
Two Loop Algorithm: Irreducible Diagrams



1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex. **Large # of helicities summed in this step (one-loop complexity).**
3. Attach $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(3)}(q_3)$ first to $\mathcal{V}_1$, then to $\mathcal{V}_0$, sum helicities of $\mathcal{N}^{(3)}(q_3)$, $\mathcal{V}_1$, $\mathcal{V}_0$.
4. Attach $\mathcal{N}^{(2)}(q_2)$ segments to previously constructed object, sum helicities on-the-fly.

$$\mathcal{U}_n^{(123)} = \mathcal{U}_{(n-1)}^{(123)} S_n^{(2)}, \qquad \mathcal{U}_0^{(123)} = \mathcal{U}^{(13)} = \mathcal{U}^{(1)}(q_1)\mathcal{N}^{(3)}(q_3)\mathcal{V}_1(q1, q2)\mathcal{V}_0(q1, q2)$$

Two Loop Algorithm: Irreducible Diagrams

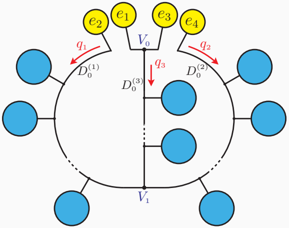


1. Construct shortest chain $\mathcal{N}^{(3)}(q_3)$.
2. Construct longest chain $\mathcal{N}^{(1)}(q_1)$ using $\mathcal{U}_0{=}2\sum_{col} C\mathcal{M}_0^*(h)$ as the initial condition. Perform on-the-fly helicity summation of ext. subtrees [Buccioni, Pozzorini, Zoller]: Begin with maximal # helicities in $\mathcal{U}_0$, sum helicities of ext. subtrees at each vertex. **Large # of helicities summed in this step (one-loop complexity).**
3. Attach $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(3)}(q_3)$ first to $\mathcal{V}_1$, then to $\mathcal{V}_0$, sum helicities of $\mathcal{N}^{(3)}(q_3)$,$\mathcal{V}_1$, $\mathcal{V}_0$.
4. Attach $\mathcal{N}^{(2)}(q_2)$ segments to previously constructed object, sum helicities on-the-fly.

Completely general and highly efficient algorithm.
Fully implemented for QED and QCD corrections to the SM.

Numerical Stability

Validate and measure numerical stability of two-loop algorithm without computing tensor integrals using **pseudotree test**.



- Cut two propagators of two-loop diagram
- Insert random wavefunctions e_1, e_2, e_3, e_4 saturating indices
- Set q_1, q_2 to random constant values, contract tensor coefficients $\mathcal{N}_{\mu_1 \ldots \mu_r \nu_1 \ldots \nu_s}$with fixed-value tensor integrand $\frac{q_1^{\mu_1} \cdots q_1^{\mu_r} q_2^{\nu_1} \cdots q_1^{\nu_s}}{\mathcal{D}(q_1, q_2)}$
- Compare to computation with well-tested tree level algorithm

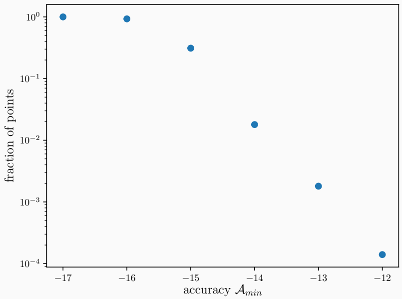
Typical accuracy around 10^{-15} in double (DP) and 10^{-30} in quad (QP) precision, always much better than 10^{-17} in QP $\Rightarrow$ **Establish QP as benchmark for DP**

Numerical Stability: Irreducible Diagrams

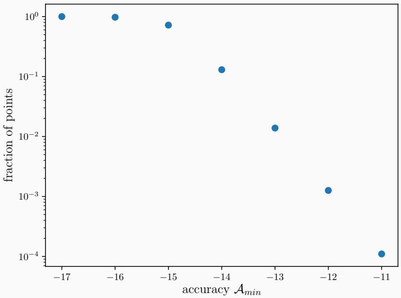
Numerical stability of scattering probability density $\mathcal{W}_{02}^{(2L,pr)}$ in double (pr=DP) vs quad (pr=QP) precision in pseudotree mode.

$$\mathcal{A}_{\mathrm{DP}} = \log_{10}\left(\frac{|\mathcal{W}_{02}^{(2\mathrm{L},\mathrm{DP})} - \mathcal{W}_{02}^{(2\mathrm{L},\mathrm{QP})}|}{\mathrm{Min}(|\mathcal{W}_{02}^{(2\mathrm{L},\mathrm{DP})}|, |\mathcal{W}_{02}^{(2\mathrm{L},\mathrm{QP})}|)}\right)$$

Process: $gg \to \bar{t}t$



Process: $d\bar{d} \to u\bar{u}g$



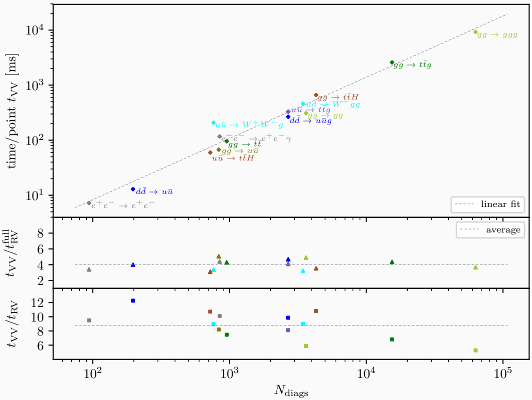
The plot shows the fraction of points with $\mathcal{A}_{\mathrm{DP}} > \mathcal{A}_{\mathrm{min}}$ for 10^5 uniform random points.

Excellent numerical stability. Essential for full calculation, tensor integrals will be main source of instabilities.

Efficiency: Irreducible Diagrams

Construction of tensor coefficients for QED, QCD and SM (NNLO QCD) processes

(single intel i7-6600U, 2.6 GHz, 16GB RAM, 1000 points)



- $2 \to 2$ process: 10-300ms/psp
- $2 \to 3$ process: 65-9200ms/psp

Runtime $\propto$ # diagrams

time/psp/diagram $\sim$ 150 μs

Constant ratios between NNLO double virtual (VV) and real-virtual (RV):

$\frac{t_{VV}}{t_{RV}^{full}} \approx 4 \pm 1$ (full RV)

$\frac{t_{VV}}{t_{RV}} \approx 9 \pm 3$ (tensor coefficients)

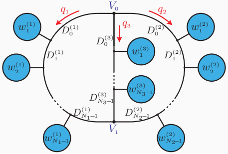
Strong CPU performance, comparable to real-virtual corrections in OpenLoops.

Implementation of Rational Terms

Renormalization and Rational Terms at NNLO

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$



$$= \underbrace{C_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1, q_2)}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\substack{\text{4-dim numerator,} \\ \text{(D-dim denominator)}}} = C_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \, q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\text{tensor integral}}$$

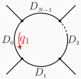
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
- Reduction and evaluation of tensor integrals

One-loop rational terms

Amputated one-loop diagram γ (bar denotes quantities in D dimensions):

$$\bar{\mathcal{M}}_{1,\gamma} = C_{1,\gamma}\int \mathrm{d}\bar{q}_1 \frac{\bar{\mathcal{N}}(q_1)}{\mathcal{D}(\bar{q}_1)} = C_{1,\gamma}\int \mathrm{d}\bar{q}_1 \frac{\overbrace{\mathcal{N}(q_1)}^{\text{4-dim}} + \overbrace{\tilde{\mathcal{N}}(\bar{q}_1)}^{\text{(D-4)-dim}}}{\mathcal{D}(\bar{q}_1)} =$$



$$\Rightarrow \delta\mathcal{R}_{1,\gamma} = C_{1,\gamma}\int \mathrm{d}\bar{q}_1 \frac{\tilde{\mathcal{N}}(\bar{q}_1)}{\mathcal{D}(\bar{q}_1)}$$

The ε-dim numerator parts $\tilde{\mathcal{N}}(\bar{q}_1) = \bar{\mathcal{N}}(\bar{q}_1) - \mathcal{N}(q_1)$ contribute only via interaction with $\frac{1}{\varepsilon}$ UV poles

$\Rightarrow$ Can be restored through **rational counterterm** $\delta\mathcal{R}_{1,\gamma}$ [Ossola, Papadopoulos, Pittau]

$$\underbrace{\mathbf{R}\,\bar{\mathcal{M}}_{1,\gamma}}_{D-\text{dim, renormalised}} = \underbrace{\mathcal{M}_{1,\gamma}}_{4-\text{dim numerator}} + \underbrace{\delta Z_{1,\gamma} + \delta\mathcal{R}_{1,\gamma}}_{\text{UV and rational counterterm}}$$

Finite set of process-independent rational terms in renormalisable models.

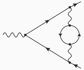
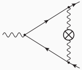
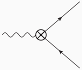
No rational terms of IR origin at one-loop [Bredenstein, Denner, Dittmaier, Pozzorini].

Two-loop rational terms

Renormalised D-dim amplitudes from amplitudes with 4-dim numerator [Pozzorini, Zhang, Zoller]

$$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \sum_{\gamma}\Big(\underbrace{\delta Z_{1,\gamma} + \delta\tilde{Z}_{1,\gamma}}_{\text{subtract subdivergences}} + \underbrace{\delta\mathcal{R}_{1,\gamma}}_{\text{restore } \tilde{\mathcal{N}}\text{-terms from subdiagrams}}\Big)\cdot\mathcal{M}_{1,\Gamma/\gamma} + \Big(\underbrace{\delta Z_{2,\Gamma}}_{\text{subtract remaining local divergence}} + \underbrace{\delta\mathcal{R}_{2,\Gamma}}_{\text{restore remaining } \tilde{\mathcal{N}}\text{-term}}\Big)$$

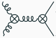
Example:

$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \Big[$  $+$  $\left(\delta Z_{1,\gamma} + \delta\tilde{Z}_{1,\gamma} + \delta\mathcal{R}_{1,\gamma}\right) +$  $\left(\delta Z_{2,\Gamma} + \delta\mathcal{R}_{2,\Gamma}\right)\Big]_{\text{4-dim numerators}}$

- Divergences from subdiagrams γ and remaining local one subtracted by usual UV counterterms $\delta Z_{1,\gamma}, \delta Z_{2,\Gamma}$.
- Additional UV counterterm $\delta\tilde{Z}_{1,\gamma} \propto \frac{(\bar{q}_1 - q_1)^2}{\varepsilon}$ for subdiagrams with mass dimension 2.
- $\delta\mathcal{R}_{2,\Gamma}$ is a **two-loop rational term** stemming from the interplay of $\tilde{\mathcal{N}}$ with UV poles, generally contains $1/\varepsilon$ poles.
- **Finite set of process-independent rational terms of UV origin.**
- **Available for QED and QCD corrections to the SM.** [Lang, Pozzorini, Zhang, Zoller,2021]
- Rational terms of IR origin currently under investigation.

Implementation of Renormalization, Rational Terms at NNLO

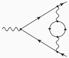
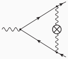
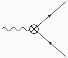
Status:

- Implementation of new tree (e.g. ) and one-loop (e.g. ) universal Feynman rules, **complete**
- Validation of new 1l tensor structures using pseudotree-test, **complete**
- **Ongoing**: Validation of implementation of two-loop rational terms, computation of first full amplitudes for simple processes $\rightarrow$ require tensor integrals

---

Pole Cancellation Check

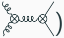
$\rightarrow$ ensure UV poles cancel

$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \Big[$  $+$  $\left(\delta Z_{1,\gamma} + \delta \tilde{Z}_{1,\gamma} + \delta \mathcal{R}_{1,\gamma}\right) +$  $\left(\delta Z_{2,\Gamma} + \delta \mathcal{R}_{2,\Gamma}\right) \Big]_{\text{4-dim numerators}}$

- nontrivial, in general $\delta\mathcal{R}_{2,\Gamma}$ contains $\frac{1}{\varepsilon}$ poles
- intermediate result in full calculation

Implementation of Renormalization, Rational Terms at NNLO

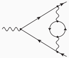
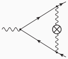
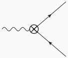
Status:

- Implementation of new tree (e.g. ) and one-loop (e.g. ) universal Feynman rules, complete
- Validation of new 1l tensor structures using pseudotree-test, complete
- Ongoing: **Validation of implementation of two-loop rational terms**, computation of first full amplitudes for simple processes $\rightarrow$ require tensor integrals

---

Pole Cancellation Check

$\rightarrow$ ensure UV poles cancel

$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \Big[$  $+$  $\left(\delta Z_{1,\gamma} + \delta\tilde{Z}_{1,\gamma} + \delta\mathcal{R}_{1,\gamma}\right) +$  $\left(\delta Z_{2,\Gamma} + \delta\mathcal{R}_{2,\Gamma}\right) \Big]_{\text{4-dim numerators}}$

- nontrivial, in general $\delta\mathcal{R}_{2,\Gamma}$ contains $\frac{1}{\varepsilon}$ poles
- intermediate result in full calculation

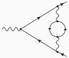
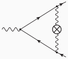
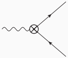
Status:

- Implementation of new tree (e.g. ) and one-loop (e.g. ) universal Feynman rules, complete
- Validation of new 1l tensor structures using pseudotree-test, complete
- Ongoing: Validation of implementation of two-loop rational terms, computation of **first full amplitudes for simple processes** $\rightarrow$ require tensor integrals

---

Pole Cancellation Check

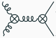
$\rightarrow$ ensure UV poles cancel

$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \Big[$  $+$  $\left(\delta Z_{1,\gamma} + \delta\tilde{Z}_{1,\gamma} + \delta\mathcal{R}_{1,\gamma}\right) +$  $\left(\delta Z_{2,\Gamma} + \delta\mathcal{R}_{2,\Gamma}\right) \Big]_{\text{4-dim numerators}}$

- nontrivial, in general $\delta\mathcal{R}_{2,\Gamma}$ contains $\frac{1}{\varepsilon}$ poles
- intermediate result in full calculation

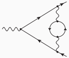
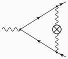
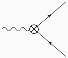
Implementation of Renormalization, Rational Terms at NNLO

Status:

- Implementation of new tree (e.g. ) and one-loop (e.g. ) universal Feynman rules, complete
- Validation of new 1l tensor structures using pseudotree-test, complete
- Ongoing: **Validation of implementation of two-loop rational terms**, computation of **first full amplitudes for simple processes** $\rightarrow$ **require tensor integrals**

Pole Cancellation Check

$\rightarrow$ ensure UV poles cancel

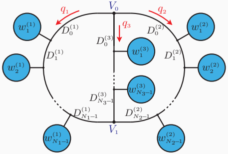
$\mathbf{R}\,\bar{\mathcal{M}}_{2,\Gamma} = \Big[$  $+$  $\left(\delta Z_{1,\gamma} + \delta\tilde{Z}_{1,\gamma} + \delta\mathcal{R}_{1,\gamma}\right)$ $+$  $\left(\delta Z_{2,\Gamma} + \delta\mathcal{R}_{2,\Gamma}\right)\Big]_{\text{4-dim numerators}}$

- nontrivial, in general $\delta\mathcal{R}_{2,\Gamma}$ contains $\frac{1}{\varepsilon}$ poles
- intermediate result in full calculation

Tensor Integral Reduction Tool

Tensor Integral Reduction Tool

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$ 

$$= \underbrace{C_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1, q_2)}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\substack{\text{4-dim numerator,}\\ \text{(D-dim denominator)}}} = C_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \, q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\text{tensor integral}}$$

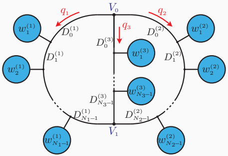
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
- Reduction and evaluation of tensor integrals

Tensor Integral Reduction Tool

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$



$$= \underbrace{\mathcal{C}_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1, q_2)}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\substack{\text{4-dim numerator,} \\ \text{(D-dim denominator)}}} = \mathcal{C}_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \, q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\text{tensor integral}}$$

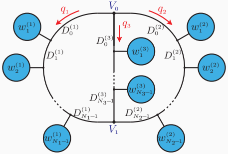
Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
- Reduction and evaluation of tensor integrals$\rightarrow$ wide range of methods and tools possible: analytical and numerical, in-house and external, and mixtures thereof

Tensor Integral Reduction Tool

Final result in D-dimensions, numerical tools: construct numerator in 4-dimensions.
For one diagram Γ:

$\mathcal{M}_{2,\Gamma} =$



$$= \underbrace{\mathcal{C}_{2,\Gamma}}_{\text{color}} \int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \; \underbrace{\frac{\mathcal{N}(q_1, q_2)}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\substack{\text{4-dim numerator,} \\ \text{(D-dim denominator)}}} = \mathcal{C}_{2,\Gamma} \sum_{r,s} \underbrace{\mathcal{N}_{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}}_{\text{tensor coefficient}} \underbrace{\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \, q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}(\bar{q}_1, \bar{q}_2)}}_{\text{tensor integral}}$$

Calculation decomposed into:

- Numerical construction of tensor coefficient in 4-dim
- Renormalization, restoration of (D-4)-dim numerator part by rational counterterms
  $\mathbf{R}\bar{\mathcal{M}}_{2,\Gamma} = \mathcal{M}_{2,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{1,1,\Gamma} + \mathcal{M}^{(\mathrm{CT})}_{0,2,\Gamma}$ [Lang, Pozzorini, Zhang, Zoller]
- Reduction and evaluation of tensor integrals→ wide range of methods and tools possible: analytical and numerical, in-house and external, and mixtures thereof

Currently working on small in-house tensor integral library for test purposes, 2 and 3 point topologies with off-shell external legs and massless propagators.

Summary

Summary

New algorithm for two loop tensor coefficients:

- **Fully general algorithm**
- **Excellent numerical stability**
- **Highly efficient, comparable to real virtual contribution**
  - Exploit factorization for ideal order of building blocks.
  - Efficient treatment of helicities and ranks in loop momenta.
- **Fully implemented for NNLO QED and QCD Corrections to SM**

Current and future projects

- Implementation of two-loop UV and rational counterterms
- Tensor integrals (in-house framework, external tools or mixture thereof)

Backup

Tensor Integral Reduction Tool: Covariant Decomposition

Example: Rank 2 tensor integral, 2 independent external momenta $\mathbf{p_1}, \mathbf{p_2}$

$$I^{\mu\nu} = \int d\bar{q}_1 d\bar{q}_2 \frac{\bar{q}_1^\mu \bar{q}_2^\nu}{D_1 \cdot D_2 \cdot D_3 \cdot D_4 \cdot D_5}$$

$D_1 = \bar{q}_1^2,\ D_2 = (\bar{q}_1 + p_1)^2,\ D_3 = (\bar{q}_1 + p_1 + p_2)^2,\ D_4 = \bar{q}_2^2,\ D_5 = (-\bar{q}_1 - \bar{q}_2)^2$

Covariant decomposition, final tensor structure can only contain external momenta, metric tensors:

$$I^{\mu\nu} = C_1\ p_1^\mu p_1^\nu + C_2\ p_2^\mu p_2^\nu + C_3\ g^{\mu\nu} + C_4\ p_1^\mu p_2^\nu + C_5\ p_2^\mu p_1^\nu$$

Use projectors to determine coefficients C_1, C_2, C_3, C_4, C_5:

$$\underbrace{\begin{pmatrix} p_{1\mu} p_{1\nu} I^{\mu\nu} \\ p_{2\mu} p_{2\nu} I^{\mu\nu} \\ g_{\mu\nu} \quad I^{\mu\nu} \\ p_{1\mu} p_{2\nu} I^{\mu\nu} \\ p_{2\mu} p_{1\nu} I^{\mu\nu} \end{pmatrix}}_{I\,=\,\text{scalar integrals}} = \underbrace{\begin{pmatrix} p_1^4 & (p_1.p_2)^2 & p_1^2 & p_1^2\ p_1.p_2 & p_1^2\ p_1.p_2 \\ (p_1.p_2)^2 & p_2^2 & p_2^4 & p_2^2\ p_1.p_2 & p_2^2\ p_1.p_2 \\ p_1^2 & p_2^2 & d & p_1.p_2 & p_1.p_2 \\ p_1^2\ p_1.p_2 & p_2^2\ p_1.p_2 & p_1.p_2 & p_1^2 p_2^2 & (p_1.p_2)^2 \\ p_2^2\ p_1.p_2 & p_1^2\ p_1.p_2 & p_1.p_2 & (p_1.p_2)^2 & p_1^2 p_2^2 \end{pmatrix}}_{M} \underbrace{\begin{pmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \end{pmatrix}}_{C} \Rightarrow C = M^{-1} \cdot I.$$

We have now related $I^{\mu\nu}$ to coefficients C depending only on **scalar integrals** I.

Our test library contains automated Mathematica implementation of this approach.

Challenge: inversion and simplification of M only feasible at low ranks in $\bar{q}_1, \bar{q}_2$.

Tensor Integral Reduction Tool: Interface to FIRE

Interface to FIRE [Smirnov, Chukharev] $\rightarrow$ **express scalar integrals in terms of D_i**

Example:

$$I_1 = p_{1\mu} p_{1\nu} I^{\mu\nu} = \int d\bar{q}_1 d\bar{q}_2 \frac{(p_1.\bar{q}_1)(p_1.\bar{q}_2)}{D_1 \cdot D_2 \cdot D_3 \cdot D_4 \cdot D_5}$$

$D_1 = \bar{q}_1^2, D_2 = (\bar{q}_1 + p_1)^2, D_3 = (\bar{q}_1 + p_1 + p_2)^2, D_4 = \bar{q}_2^2, D_5 = (-\bar{q}_1 - \bar{q}_2)^2$

Find $p_1.q_1 = \frac{1}{2}(D_2 - D_1 - p_1^2)$, $\qquad p_1.q_2 = \frac{1}{2}(D_6 - D_4 - p_1^2)$,
where we introduced an additional propagator $D_6 = (\bar{q}_2 + p_1)^2$ for $p_1.q_2$.

$$I_1 = p_{1\mu} p_{1\nu} I^{\mu\nu} = \int d\bar{q}_1 d\bar{q}_2 \frac{\frac{1}{4}(D_2 - D_1 - p_1^2)(D_6 - D_4 - p_1^2)}{D_1^1 \cdot D_2^1 \cdot D_3^1 \cdot D_4^1 \cdot D_5^1 \cdot D_6^0}$$

$\rightarrow$ Scalar integrals in I_1 are now uniquely identified by exponents of $\{D_1, D_2, D_3, D_4, D_5, D_6\}$

Example : $G[\{2,1,1,0,-1,0\}] = \int d\bar{q}_1 d\bar{q}_2 \frac{D_5}{{D_1}^2 \cdot D_2 \cdot D_3}$

These expressions are now ready for reduction.

Interface to FIRE automated in our test library.

Remaining steps: ε expansion of coefficients, implementation of master integrals from literature or numerical evaluation thereof.

On-The-Fly Helicity Summation at NLO

Final result: $\mathcal{W}_{01} = \sum_h \sum_{\text{col}} 2\,\text{Re}\left[\bar{\mathcal{M}}_1(h)\,\bar{\mathcal{M}}_0^*(h)\right]$

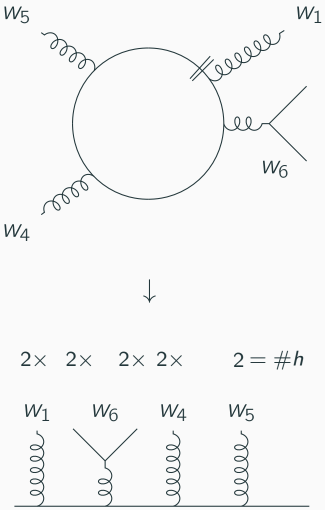
Instead of $\mathcal{N}(q,h) = \prod_a S_a(q,h)$, construct $\mathcal{U}(q) = \sum_h \left[2\sum_{col} C\mathcal{M}_0^*(h)\right]\mathcal{N}(q,h)$

Perform **on-the-fly helicity summation** [Buccioni, Pozzorini, Zoller], for each diagram:

- Use Born-color interfernce $\mathcal{U}_0 = 2\sum_{col} C\mathcal{M}_0^*(h)$ as initial condition, begin the recursion with maximal helicities.
- **Exploit factorization to sum helicities in each recursion step:**
  $\sum_h \mathcal{U}_0(h)\,\mathcal{N}(q,h) = \sum_{h_N}\left[\cdots\sum_{h_2}\left[\sum_{h_1}\mathcal{U}_0(h_1,h_2,\ldots)S_1(h_1)\right]S_2(h_2)\cdots\right]S_N(h_N)$
- (in renormalizable theories) each segment:
  - increases rank by 1 (or 0)
  - decreases total helicities by a factor of # helicities of subtree in the segment

Minimal helicities with maximal rank, complexity is kept low in final recursion steps.

On-The-Fly Helicity Summation: Example

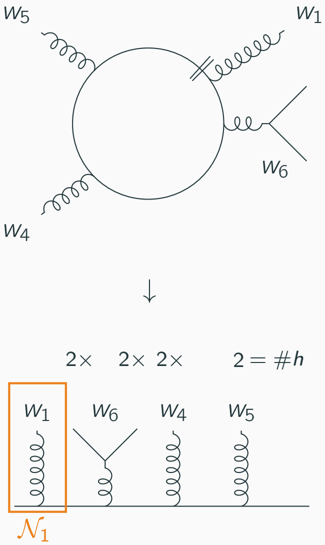


In each recursion step:

- increase rank by 1
- decrease total helicities by a factor of # helicities of wavefunction in the segment

helicities=32,
rank=0

On-The-Fly Helicity Summation: Example

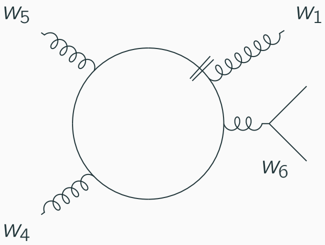


In each recursion step:

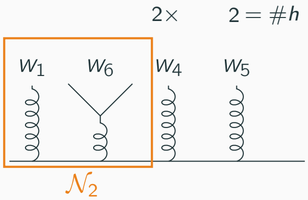
- increase rank by 1
- decrease total helicities by a factor of # helicities of wavefunction in the segment

helicities=16,
rank=1

On-The-Fly Helicity Summation: Example



$\downarrow$

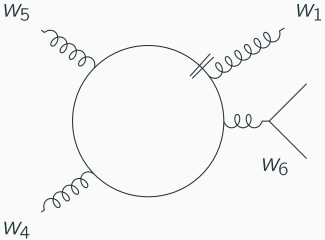


In each recursion step:

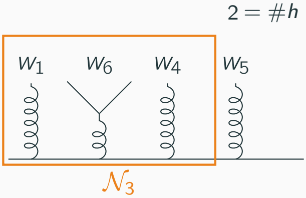
- increase rank by 1
- decrease total helicities by a factor of # helicities of wavefunction in the segment

helicities=4,
rank=2

On-The-Fly Helicity Summation: Example



↓

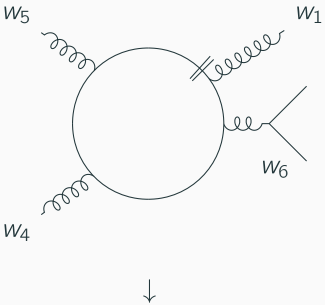


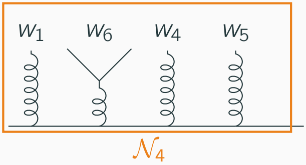
In each recursion step:

- increase rank by 1
- decrease total helicities by a factor of # helicities of wavefunction in the segment

helicities=2,
rank=3

On-The-Fly Helicity Summation: Example





In each recursion step:

- increase rank by 1
- decrease total helicities by a factor of # helicities of wavefunction in the segment

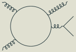
helicities=1,
rank=4

Symmetrization at One-Loop

OpenLoops uses **symmetrized** tensor coefficients:

$$\sum_{r=0}^{R} \mathcal{N}_{\mu_1 \cdots \mu_r} q^{\mu_1} \cdots q^{\mu_r} = \sum_{\substack{n_i=0,\\ n_0+n_1+n_2+n_3 \leq R}} \mathcal{N}_{n_0 n_1 n_2 n_3} \left(q^0\right)^{n_0} \left(q^1\right)^{n_1} \left(q^2\right)^{n_2} \left(q^3\right)^{n_3}$$

Example:



$$= \mathcal{M}_1 = \sum_{r=0}^{R} c_1 \mathcal{N}_{\mu_1 \ldots \mu_r} \int \mathrm{d}q \, \frac{q^{\mu_1} \ldots q^{\mu_r}}{D_0 D_1 D_2 D_3}$$

$\#$ components in $\mathcal{N}$ for $R = 4$

- without symmetrization: $\sum_{r=0}^{R} 4^r = 341$
- with symmetrization: $\binom{R+4}{R} = 70$

Bookkeeping in numerical code:

Map n_0, n_1, n_2, n_3 onto one-dimensional array $\ell(n_0, n_1, n_2, n_3)$:

$$\ell(n_0, n_1, n_2, n_3) = \binom{3+n_3-1}{n_3-1} + \binom{2+n_3+n_2-1}{n_3+n_2-1} + \binom{1+n_3+n_2+n_1-1}{n_3+n_2+n_1-1} + n_3 + n_2 + n_1 + n_0 + 1$$

$\rightarrow$ extension to two loops: use (ℓ_1, ℓ_2) for coefficients related to (q_1, q_2).

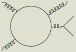
Symmetrization greatly reduces number of operations required in numerator construction.

Symmetrization at One-Loop

OpenLoops uses **symmetrized** tensor coefficients:

$$\sum_{r=0}^{R} \mathcal{N}_{\mu_1 \cdots \mu_r} q^{\mu_1} \cdots q^{\mu_r} = \sum_{\substack{n_i=0,\\ n_0+n_1+n_2+n_3 \leq R}} \mathcal{N}_{n_0 n_1 n_2 n_3} \left(q^0\right)^{n_0} \left(q^1\right)^{n_1} \left(q^2\right)^{n_2} \left(q^3\right)^{n_3} = \mathcal{N}(\ell) q(\ell)$$

Example:



$$= \mathcal{M}_1 = \sum_{r=0}^{R} c_1 \mathcal{N}_{\mu_1 \ldots \mu_r} \int \mathrm{d}q \frac{q^{\mu_1} \ldots q^{\mu_r}}{D_0 D_1 D_2 D_3}$$

$\#$ components in $\mathcal{N}$ for $R = 4$

- without symmetrization: $\sum_{r=0}^{R} 4^r = 341$
- with symmetrization: $\binom{R+4}{R} = 70$

Bookkeeping in numerical code:

Map n_0, n_1, n_2, n_3 onto one-dimensional array $\ell(n_0, n_1, n_2, n_3)$:

$$\ell(n_0, n_1, n_2, n_3) = \binom{3+n_3-1}{n_3-1} + \binom{2+n_3+n_2-1}{n_3+n_2-1} + \binom{1+n_3+n_2+n_1-1}{n_3+n_2+n_1-1} + n_3 + n_2 + n_1 + n_0 + 1$$

$\rightarrow$ extension to two loops: use (ℓ_1, ℓ_2) for coefficients related to (q_1, q_2).

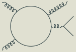
Symmetrization greatly reduces number of operations required in numerator construction.

Symmetrization at One-Loop

OpenLoops uses **symmetrized** tensor coefficients:

$$\sum_{r=0}^{R} \mathcal{N}_{\mu_1\cdots\mu_r} q^{\mu_1}\cdots q^{\mu_r} = \sum_{\substack{n_i=0,\\ n_0+n_1+n_2+n_3\leq R}} \mathcal{N}_{n_0 n_1 n_2 n_3} \left(q^0\right)^{n_0} \left(q^1\right)^{n_1} \left(q^2\right)^{n_2} \left(q^3\right)^{n_3} = \mathcal{N}(\ell) q(\ell)$$

Example:



$$= \mathcal{M}_1 = \sum_{r=0}^{R} C_1 \mathcal{N}_{\mu_1\ldots\mu_r} \int \mathrm{d}q \, \frac{q^{\mu_1}\ldots q^{\mu_r}}{D_0 D_1 D_2 D_3}$$

$\#$ components in $\mathcal{N}$ for $R=4$

- without symmetrization: $\sum_{r=0}^{R} 4^r = 341$
- with symmetrization: $\binom{R+4}{R} = 70$

Bookkeeping in numerical code:

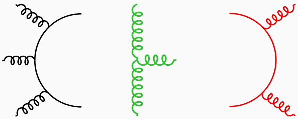
Map n_0, n_1, n_2, n_3 onto one-dimensional array $\ell(n_0, n_1, n_2, n_3)$:

rank	0	1				2									
$q(\ell)$	1	q^0	q^1	q^2	q^3	q^0q^0	q^0q^1	q^0q^2	q^0q^3	q^1q^1	q^1q^2	q^1q^3	q^2q^2	q^2q^3	q^3q^3
ℓ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

$\rightarrow$ extension to two loops: use (ℓ_1, ℓ_2) for coefficients related to (q_1, q_2).

Symmetrization greatly reduces number of operations required in numerator construction.

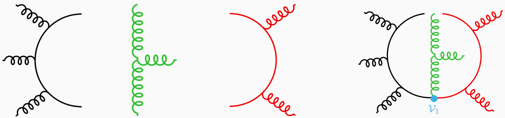
Two Loop Algorithm: Naive Approach



1. construct chains $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(2)}(q_2)$, $\mathcal{N}^{(3)}(q_3)$ using one-loop algorithm.

$$\left[\mathcal{N}^{(1)}(q_1)\right]_{\beta_0^{(1)}}^{\beta_{N_1}^{(1)}} \left[\mathcal{N}^{(2)}(q_2)\right]_{\beta_0^{(2)}}^{\beta_{N_2}^{(2)}} \left[\mathcal{N}^{(3)}(q_3)\right]_{\beta_0^{(3)}}^{\beta_{N_3}^{(3)}}$$

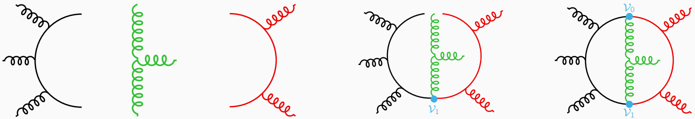
Two Loop Algorithm: Naive Approach



1. construct chains $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(2)}(q_2)$, $\mathcal{N}^{(3)}(q_3)$ using one-loop algorithm.
2. combine with vertex $\mathcal{V}_1$, closing indices $\beta^{(1)}_{N_1}, \beta^{(2)}_{N_2}, \beta^{(3)}_{N_3}$

$$\left[\mathcal{N}^{(1)}(q_1)\right]_{\beta_0^{(1)}}^{\beta_{N_1}^{(1)}} \left[\mathcal{N}^{(2)}(q_2)\right]_{\beta_0^{(2)}}^{\beta_{N_2}^{(2)}} \left[\mathcal{N}^{(3)}(q_3)\right]_{\beta_0^{(3)}}^{\beta_{N_3}^{(3)}} \left[\mathcal{V}_1(q1,\, q2)\right]_{\beta_{N_1}^{(1)}\beta_{N_2}^{(2)}\beta_{N_3}^{(3)}}$$

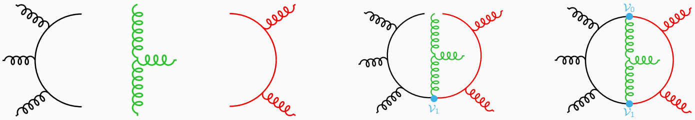
Two Loop Algorithm: Naive Approach



1. construct chains $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(2)}(q_2)$, $\mathcal{N}^{(3)}(q_3)$ using one-loop algorithm.
2. combine with vertex $\mathcal{V}_1$, closing indices $\beta^{(1)}_{N_1}, \beta^{(2)}_{N_2}, \beta^{(3)}_{N_3}$
3. combine with vertex $\mathcal{V}_0$, closing indices $\beta^{(1)}_{0}, \beta^{(2)}_{0}, \beta^{(3)}_{0}$

$$\left[\mathcal{N}^{(1)}(q_1)\right]_{\beta_0^{(1)}}^{\beta_{N_1}^{(1)}} \left[\mathcal{N}^{(2)}(q_2)\right]_{\beta_0^{(2)}}^{\beta_{N_2}^{(2)}} \left[\mathcal{N}^{(3)}(q_3)\right]_{\beta_0^{(3)}}^{\beta_{N_3}^{(3)}} \left[\mathcal{V}_1(q1,\, q2)\right]_{\beta_{N_1}^{(1)}\beta_{N_2}^{(2)}\beta_{N_3}^{(3)}} \left[\mathcal{V}_0(q1,\, q2)\right]^{\beta_0^{(1)}\beta_0^{(2)}\beta_0^{(3)}}$$

Two Loop Algorithm: Naive Approach



1. construct chains $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(2)}(q_2)$, $\mathcal{N}^{(3)}(q_3)$ using one-loop algorithm.
2. combine with vertex $\mathcal{V}_1$, closing indices $\beta^{(1)}_{N_1}, \beta^{(2)}_{N_2}, \beta^{(3)}_{N_3}$
3. combine with vertex $\mathcal{V}_0$, closing indices $\beta^{(1)}_{0}, \beta^{(2)}_{0}, \beta^{(3)}_{0}$
4. multiply Born-color interference, sum over helicities, map momenta

$$\sum_h \mathcal{U}_0(h) \left[\mathcal{N}^{(1)}(q_1, h)\right] \left[\mathcal{N}^{(2)}(q_2, h)\right] \left[\mathcal{N}^{(3)}(q_3, h)\right] \left[\mathcal{V}_1(q1, q2, h)\right] \left[\mathcal{V}_0(q1, q2, h)\right] \Big|_{q_3 \to -(q_1+q_2)}$$

Two Loop Algorithm: Observations and Challenges

$$\sum_h \mathcal{U}_0(h) \left[\mathcal{N}^{(1)}(q_1, h)\right] \left[\mathcal{N}^{(2)}(q_2, h)\right] \left[\mathcal{N}^{(3)}(q_3, h)\right] \left[\mathcal{V}_1(q1, q2, h)\right] \left[\mathcal{V}_0(q1, q2, h)\right] \Big|_{q_3 \to -(q_1+q_2)}$$

1. construct chains $\mathcal{N}^{(1)}(q_1)$, $\mathcal{N}^{(2)}(q_2)$, $\mathcal{N}^{(3)}(q_3)$ using one-loop algorithm
2. combine with vertex $\mathcal{V}_1$, closing indices $\beta^{(1)}_{N_1}, \beta^{(2)}_{N_2}, \beta^{(3)}_{N_3}$
3. combine with vertex $\mathcal{V}_0$, closing indices $\beta^{(1)}_0, \beta^{(2)}_0, \beta^{(3)}_0$
4. sum over helicities, map momenta, multiply Born-color interference

Observations:

- complexitiy of each step depends on ranks in q_1, q_2 and helicities
- step 2, 3 are performed for 6, 3 open spinor/Lorentz indices
- step 2, 3 are performed at maximal ranks
- all steps are performed for all helicities

Very inefficient: most expensive steps performed for maximal number of components and helicities.

Helicity Bookkeeping

For a set of particles $\mathcal{E} = \{1, 2, \ldots, N\}$ the helicity configurations are identified as:

$$\lambda_p = \begin{cases} 1,3 & \text{for fermions with helicity } s = -1/2, 1/2 \\ 1,2,3 & \text{for gauge bosons with } s = -1, 0, 1 \\ 0 & \text{for scalars with } s = 0 \text{ or unpolarized particles} \end{cases} \qquad \forall\ p \in \mathcal{E}$$

Each particle is assigned a base 4 helicity label

$$\bar{h}_p = \lambda_p \, 4^{p-1},$$

which can be used to define a similar numbering scheme for a set of particles:

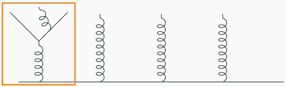
$\mathcal{E}_a = \{p_{a_1}, \ldots, p_{a_n}\}$ has the helicity label,

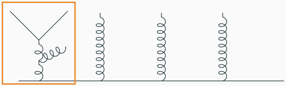
$$h_a = \sum_{p \in \mathcal{E}_a} \bar{h}_p.$$

Merging

Example:

- After one dressing step subsequent dressing steps are identical.
- Topology (scalar propagators) is identical for both diagrams.
- Diagrams can be merged.





For diagrams A,B with identical segments after n dressing steps (exploit factorization):

$$\mathcal{U}_{A,B} = \mathcal{U}_0 \, Tr(\mathcal{N}_{A,B}) = \text{numerator} \cdot \text{Born} \cdot \text{color}$$

$$\begin{aligned} \mathcal{U}_A + \mathcal{U}_B &= (\mathcal{U}_{n,A} \cdot S_{n+1} \cdots S_N) + (\mathcal{U}_{n,B} \cdot S_{n+1} \cdots S_N) \\ &= (\mathcal{U}_{n,A} + \mathcal{U}_{n,B}) \cdot S_{n+1} \cdots S_N \end{aligned}$$

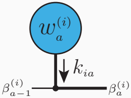
Only perform dressing steps n+1 to N once.

Highly efficient way of dressing a large number of diagrams for complicated processes.

Explicit dressing steps

Triple vertex loop segment:

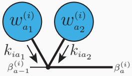
$$\left[S_a^{(i)}(q_i, h_a^{(i)})\right]_{\beta_{a-1}^{(i)}}^{\beta_a^{(i)}} =$$



$$= \left\{ \left[Y_{ia}^{\sigma}\right]_{\beta_{a-1}^{(i)}}^{\beta_a^{(i)}} + \left[Z_{ia,\nu}^{\sigma}\right]_{\beta_{a-1}^{(i)}}^{\beta_a^{(i)}} q_i^{\nu} \right\} w_{a\sigma}^{(i)}(k_{ia}, h_a^{(i)})$$

Quartic vertex segments:

$$\left[S_a^{(i)}(q_i, h_a^{(i)})\right]_{\beta_{a-1}^{(i)}}^{\beta_a^{(i)}} =$$



$$= \left[Y_{ia}^{\sigma_1\sigma_2}\right]_{\beta_{a-1}^{(i)}}^{\beta_a^{(i)}} w_{a_1\sigma_1}^{(i)}(k_{ia_1}, h_{a_1}^{(i)})\, w_{a_2\sigma_2}^{(i)}(k_{ia_2}, h_{a_2}^{(i)})$$

with $h_a^{(i)} = h_{a_1}^{(i)} + h_{a_2}^{(i)}$ and $k_{ia} = k_{ia_1} + k_{ia_2}$.

Dressing step for a segment with a triple vertex:

$$\begin{aligned}\left[\mathcal{N}_{n;\,\mu_1\ldots\mu_r}^{(1)}(\hat{h}_n^{(1)})\right]_{\beta_0^{(1)}}^{\beta_n^{(1)}} = & \Bigg\{ \left[\mathcal{N}_{n-1;\,\mu_1\ldots\mu_r}^{(1)}(\hat{h}_{n-1}^{(1)})\right]_{\beta_0^{(1)}}^{\beta_{n-1}^{(1)}} \left[Y_{1n}^{\sigma}\right]_{\beta_{n-1}^{(1)}}^{\beta_n^{(1)}} \\ & + \left[\mathcal{N}_{n-1;\,\mu_2\ldots\mu_r}^{(1)}(\hat{h}_{n-1}^{(1)})\right]_{\beta_0^{(1)}}^{\beta_{n-1}^{(1)}} \left[Z_{1n,\mu_1}^{\sigma}\right]_{\beta_{n-1}^{(1)}}^{\beta_n^{(1)}} \Bigg\} w_{n\sigma}^{(1)}(k_n, h_n^{(1)}).\end{aligned}$$

Processes considered in performance tests

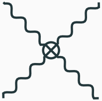
corrections	process type	massless fermions	massive fermions	process
QED	$2 \to 2$	e	–	$e^+e^- \to e^+e^-$
	$2 \to 3$	e	–	$e^+e^- \to e^+e^-\gamma$
QCD	$2 \to 2$	u	–	$gg \to u\bar{u}$
		u, d	–	$d\bar{d} \to u\bar{u}$
		u	–	$gg \to gg$
		u	t	$u\bar{u} \to t\bar{t}g$
		u	t	$gg \to t\bar{t}$
		u	t	$gg \to t\bar{t}g$
	$2 \to 3$	u, d	–	$d\bar{d} \to u\bar{u}g$
		u	–	$gg \to ggg$
		u, d	–	$u\bar{d} \to W^+gg$
		u, d	–	$u\bar{u} \to W^+W^-g$
		u	t	$u\bar{u} \to t\bar{t}H$
		u	t	$gg \to t\bar{t}H$

Memory usage of the two-loop algorithm

	virtual–virtual memory [MB]		real–virtual [MB]	
hard process	segment-by-segment	diagram-by-diagram	coefficients	full
$e^+e^- \to e^+e^-$	18	8	6	23
$e^+e^- \to e^+e^-\gamma$	154	25	22	54
$gg \to u\bar{u}$	75	31	10	26
$gg \to t\bar{t}$	94	35	15	34
$gg \to t\bar{t}g$	2000	441	152	213
$u\bar{d} \to W^+gg$	563	143	54	90
$u\bar{u} \to W^+W^-g$	264	67	36	67
$u\bar{u} \to t\bar{t}H$	82	28	14	40
$gg \to t\bar{t}H$	604	145	50	90
$u\bar{u} \to t\bar{t}g$	323	83	41	74
$gg \to gg$	271	94	41	55
$d\bar{d} \to u\bar{u}$	18	10	9	20
$d\bar{d} \to u\bar{u}g$	288	85	39	68
$gg \to ggg$	6299	1597	623	683

Implementation of Renormalization, Rational Terms

Example (from `arXiv:2001.11388v3`) :



$$\sim ie^2 \underbrace{\left(g^{\mu\nu}g^{\rho\sigma} + g^{\mu\rho}g^{\nu\sigma} + g^{\mu\sigma}g^{\nu\rho}\right)}_{\text{tensor structure}} \sum_{k=1}^{2} \left(\frac{\alpha}{4\pi}\right)^k \underbrace{\delta R^{(s)}_{k,4\gamma}}_{\substack{\text{rational} \\ \text{counterterms}}}$$

where k=1,2 is the loop order.

For NNLO need to implement:

- universal Feynman rules for new tensor structures
- new rational counterterms

Tensor Integrals

At NNLO require:

- One-loop tensor integrals
  - One-loop diagrams with counterterm insertions: up to $\mathcal{O}(\epsilon)$, new topolgies due to squared propagator,
    e.g.  $= \int \mathrm{d}\bar{q}_1 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r}}{\bar{D}_0 \bar{D}_0 \bar{D}_1 \bar{D}_2} = I^{\mu_1 \cdots \mu_r}.$
  - Solution for $\delta \tilde{Z}_1 \propto \tilde{q}^2$ integrals, stemming from resotration of $(D-4)$-dimensional numerator parts.
  - Integrals for reducible double-virtual, virtual, real-virtual and loop-squared diagrams available in public OpenLoops.
- Two-loop tensor integrals
  - irreducible double-virtual diagrams:
    $$\int \mathrm{d}\bar{q}_1 \int \mathrm{d}\bar{q}_2 \frac{q_1^{\mu_1} \cdots q_1^{\mu_r} \; q_2^{\nu_1} \cdots q_2^{\nu_s}}{\mathcal{D}^{(1)}(\bar{q}_1)\, \mathcal{D}^{(2)}(\bar{q}_2)\, \mathcal{D}^{(3)}(\bar{q}_3)} \Bigg|_{q_3 \to -(q_1+q_2)} = I^{\mu_1 \cdots \mu_r \nu_1 \cdots \nu_s}$$