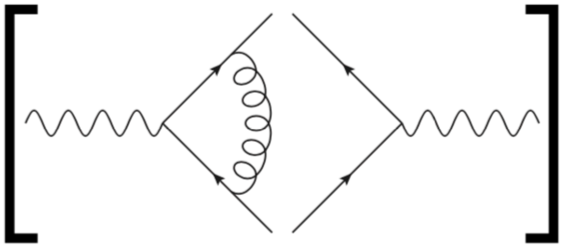
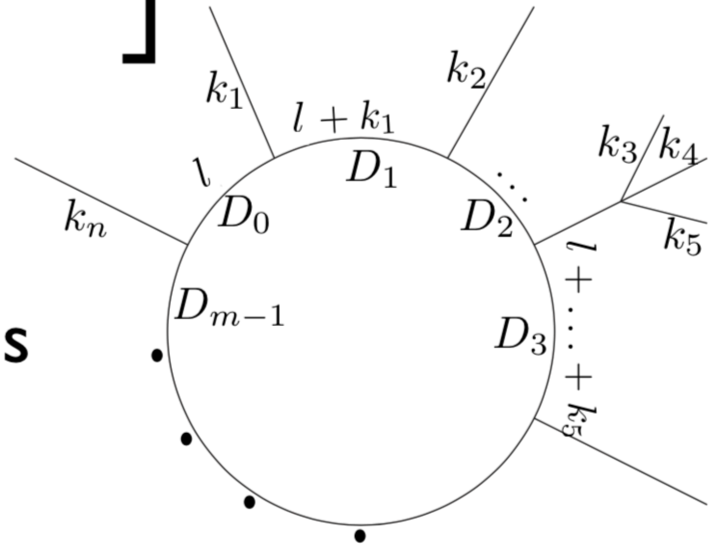
Loop Computation

Loop computation

- Consider a m-point one-loop diagram with n external momenta

$$d\sigma_V = 2\Re$$





$$p_1 = k_1$$

$$p_2 = k_1 + k_2$$

$$p_3 = \sum_1^5 k_i$$

- The integral to compute is

$$\int d^d l \frac{N(l)}{D_0 D_1 \ldots D_{m-1}} \qquad D_i = (l + p_i)^2 - m_i^2$$

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

$$\text{Tadpole}_{i_0} = \int d^dl \frac{1}{D_{i_0}}$$

$$\text{Bubble}_{i_0 i_1} = \int d^dl \frac{1}{D_{i_0} D_{i_1}}$$

$$\text{Triangle}_{i_0 i_1 i_2} = \int d^dl \frac{1}{D_{i_0} D_{i_1} D_{i_2}}$$

$$\text{Box}_{i_0 i_1 i_2 i_3} = \int d^dl \frac{1}{D_{i_0} D_{i_1} D_{i_2} D_{i_3}}$$

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0 i_1 i_2 i_3} \text{Box}_{i_0 i_1 i_2 i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0 i_1 i_2} \text{Triangle}_{i_0 i_1 i_2} \\ & + \sum_{i_0<i_1} b_{i_0 i_1} \text{Bubble}_{i_0 i_1} \\ & + \sum_{i_0} a_{i_0} \text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

- Available in computer libraries (FF [v. Oldenborgh], QCDLoop [Ellis, Zanderighi], OneLOop [v. Hameren])

Divergences

- The a, b, c, d and R coefficients depend only on external parameters and momenta

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0 i_1 i_2 i_3} \text{Box}_{i_0 i_1 i_2 i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0 i_1 i_2} \text{Triangle}_{i_0 i_1 i_2} \\ & + \sum_{i_0<i_1} b_{i_0 i_1} \text{Bubble}_{i_0 i_1} \\ & + \sum_{i_0} a_{i_0} \text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

$$D_i = (l+p_i)^2 - m_i^2$$

$$\text{Tadpole}_{i_0} = \int d^d l \frac{1}{D_{i_0}}$$

$$\text{Bubble}_{i_0 i_1} = \int d^d l \frac{1}{D_{i_0} D_{i_1}}$$

$$\text{Triangle}_{i_0 i_1 i_2} = \int d^d l \frac{1}{D_{i_0} D_{i_1} D_{i_2}}$$

$$\text{Box}_{i_0 i_1 i_2 i_3} = \int d^d l \frac{1}{D_{i_0} D_{i_1} D_{i_2} D_{i_3}}$$

➡The coefficients d, c, b and a are finite and do not contain poles in 1/ϵ

➡The 1/ϵ dependence is in the scalar integrals (and the UV renormalization)

➡Divergencies related to the Real

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

Standard Approach

- Passarino-Veltman reduction:

$$\int d^d l \, \frac{N(l)}{D_0 D_1 D_2 \cdots D_{m-1}} \rightarrow \sum_i \mathrm{coeff}_i \int d^d l \, \frac{1}{D_0 D_1 \cdots}$$

  - Reduce a general integral to “scalar integrals” by “completing the square”

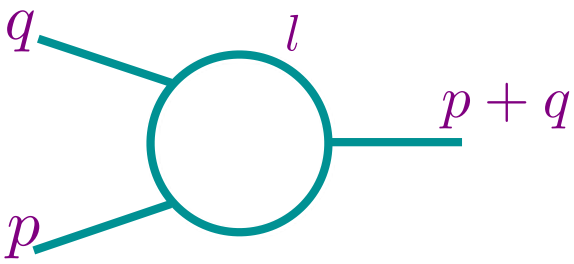
Standard Approach

- Passarino-Veltman reduction:

$$\int d^d l \, \frac{N(l)}{D_0 D_1 D_2 \cdots D_{m-1}} \to \sum_i \mathrm{coeff}_i \int d^d l \, \frac{1}{D_0 D_1 \cdots}$$

  - Reduce a general integral to "scalar integrals" by "completing the square"

- Let's do an example:
Suppose we want to calculate this triangle integral



$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$\int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l{\cdot}p}{l^2(l{+}p)^2(l{+}q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p \cdot p & 2p \cdot q \\ 2p \cdot q & 2q \cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Passarino-Veltman

Main Idea

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)}$$

- The only independent four vectors are p^μ and q^μ . Therefore, the integral must be proportional to those. We can set-up a system of linear equations.

$$\int \frac{d^n l}{(2\pi)^n} \frac{l^\mu}{(l^2 - m_1^2)((l+p)^2 - m_2^2)((l+q)^2 - m_3^2)} = \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

Resolution (dropping the mass)

- contracting with $2p^\mu$ and $2q^\mu$

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} 2p \cdot p & 2p \cdot q \\ 2p \cdot q & 2q \cdot q \end{pmatrix}} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l \cdot p] \\ [2l \cdot q] \end{pmatrix}$$

Gram Determinant: G

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} \begin{pmatrix} p^\mu & q^\mu \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Gram Determinant: G

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \ q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Gram Determinant: G

Resolution (dropping the mass)

- express the integral as simpler integral

$$\int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2} = \int \frac{d^n l}{(2\pi)^n}\frac{(l+p)^2 - l^2 - p^2}{l^2(l+p)^2(l+q)^2}$$

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Gram Determinant: G

Resolution (dropping the mass)

- express the integral as simpler integral

$$\int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2} = \int \frac{d^n l}{(2\pi)^n}\frac{(l+p)^2 - l^2 - p^2}{l^2(l+p)^2(l+q)^2}$$

$$= \int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+q)^2} - \int \frac{d^n l}{(2\pi)^n}\frac{1}{(l+p)^2(l+q)^2} - p^2\int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+p)^2(l+q)^2}$$

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \ q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

Gram Determinant: G

Resolution (dropping the mass)

- express the integral as simpler integral

$$\int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2} = \int \frac{d^n l}{(2\pi)^n}\frac{(l+p)^2 - l^2 - p^2}{l^2(l+p)^2(l+q)^2}$$

$$= \int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+q)^2} - \int \frac{d^n l}{(2\pi)^n}\frac{1}{(l+p)^2(l+q)^2} - p^2 \int \frac{d^n l}{(2\pi)^n}\frac{1}{l^2(l+p)^2(l+q)^2}$$

Scalar Integral: Know analytically

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^n l}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix}(p^\mu \ q^\mu)\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix}}\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}}$$

Gram Determinant: G

🟪 Already computed

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l \cdot p] = \int \frac{d^n l}{(2\pi)^n} \frac{2l \cdot p}{l^2 (l+p)^2 (l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \ q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} 2p \cdot p & 2p \cdot q \\ 2p \cdot q & 2q \cdot q \end{pmatrix}} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} [2l \cdot p] \\ [2l \cdot q] \end{pmatrix}}$$

Gram Determinant: G

🟪 Already computed

Passarino-Veltman

Resolution (dropping the mass)

- contracting with 2*p and 2*q

$$[2l\cdot p] = \int \frac{d^nl}{(2\pi)^n}\frac{2l\cdot p}{l^2(l+p)^2(l+q)^2}$$

$$\begin{pmatrix} 2p_\mu \\ 2q_\mu \end{pmatrix} (p^\mu \;\; q^\mu) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} 2p\cdot p & 2p\cdot q \\ 2p\cdot q & 2q\cdot q \end{pmatrix}} \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = \boxed{\begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}}$$

Gram Determinant: G

Final Step

- Inverting the Gram Determinant

$$\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} = G^{-1} \begin{pmatrix} [2l\cdot p] \\ [2l\cdot q] \end{pmatrix}$$

- We have an expression in term of scalar integral

$$\int \frac{d^nl}{(2\pi)^n}\frac{l^\mu}{(l^2-m_1^2)((l+p)^2-m_2^2)((l+q)^2-m_3^2)} = (\, p^\mu \quad q^\mu \,) \begin{pmatrix} C_1 \\ C_2 \end{pmatrix}$$

🟪 Already computed

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

Integrand reduction

Key Point

- Any one-loop integral can be decomposed in scalar integrals
- The task is to find these coefficients efficiently (analytically or numerically)

Two methods

- Passarino-Veltman
- OPP

OPP Reduction

- The decomposition to scalar integrals presented before works at the level of the integrals

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0i_1i_2i_3}\text{Box}_{i_0i_1i_2i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0i_1i_2}\text{Triangle}_{i_0i_1i_2} \\ & + \sum_{i_0<i_1} b_{i_0i_1}\text{Bubble}_{i_0i_1} \\ & + \sum_{i_0} a_{i_0}\text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

OPP Reduction

- The decomposition to scalar integrals presented before works at the level of the integrals

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0i_1i_2i_3}\text{Box}_{i_0i_1i_2i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0i_1i_2}\text{Triangle}_{i_0i_1i_2} \\ & + \sum_{i_0<i_1} b_{i_0i_1}\text{Bubble}_{i_0i_1} \\ & + \sum_{i_0} a_{i_0}\text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

- If we would know a similar relation at the **integrand** level, we would be able to manipulate the integrands and extract the coefficients without doing the integrals

OPP Reduction

- The decomposition to scalar integrals presented before works at the level of the integrals

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0i_1i_2i_3}\text{Box}_{i_0i_1i_2i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0i_1i_2}\text{Triangle}_{i_0i_1i_2} \\ & + \sum_{i_0<i_1} b_{i_0i_1}\text{Bubble}_{i_0i_1} \\ & + \sum_{i_0} a_{i_0}\text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

- If we would know a similar relation at the **integrand** level, we would be able to manipulate the integrands and extract the coefficients without doing the integrals

$$\begin{aligned}N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\ & + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\ & + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\ & + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\ & + \tilde{P}(l) \prod_{i}^{m-1} D_i\end{aligned}$$

OPP Reduction

- The decomposition to scalar integrals presented before works at the level of the integrals

$$\begin{aligned}\mathcal{M}^{\text{1-loop}} = & \sum_{i_0<i_1<i_2<i_3} d_{i_0 i_1 i_2 i_3}\,\text{Box}_{i_0 i_1 i_2 i_3} \\ & + \sum_{i_0<i_1<i_2} c_{i_0 i_1 i_2}\,\text{Triangle}_{i_0 i_1 i_2} \\ & + \sum_{i_0<i_1} b_{i_0 i_1}\,\text{Bubble}_{i_0 i_1} \\ & + \sum_{i_0} a_{i_0}\,\text{Tadpole}_{i_0} \\ & + R + \mathcal{O}(\epsilon)\end{aligned}$$

- If we would know a similar relation at the **integrand** level, we would be able to manipulate the integrands and extract the coefficients without doing the integrals

$$\begin{aligned}N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \boxed{\tilde{d}_{i_0 i_1 i_2 i_3}(l)}\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\ & + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \boxed{\tilde{c}_{i_0 i_1 i_2}(l)}\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\ & + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \boxed{\tilde{b}_{i_0 i_1}(l)}\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\ & + \sum_{i_0}^{m-1} \left[a_{i_0} + \boxed{\tilde{a}_{i_0}(l)}\right] \prod_{i\neq i_0}^{m-1} D_i \\ & + \tilde{P}(l) \prod_{i}^{m-1} D_i\end{aligned}$$

Spurious term

spurious terms

- The functional form of the spurious terms is known (it depends on the rank of the integral and the number of propagators in the loop) [del Aguila, Pittau 2004]
  - for example, a box coefficient from a rank 1 numerator is

$$\tilde{d}_{i_0 i_1 i_2 i_3}(l) = \tilde{d}_{i_0 i_1 i_2 i_3}\, \epsilon^{\mu\nu\rho\sigma}\, l^\mu p_1^\nu p_2^\rho p_3^\sigma$$

(remember that p_i is the sum of the momentum that has entered the loop so far, so we always have $p_0 = 0$)

  - The integral is zero

$$\int d^d l \frac{\tilde{d}_{i_0 i_1 i_2 i_3}(l)}{D_0 D_1 D_2 D_3} = \tilde{d}_{i_0 i_1 i_2 i_3} \int d^d l \frac{\epsilon^{\mu\nu\rho\sigma}\, l^\mu p_1^\nu p_2^\rho p_3^\sigma}{D_0 D_1 D_2 D_3} = 0$$

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
+ & \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
+ & \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
+ & \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
+ & \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

How it works...

$$N(l) = \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i$$

$$+ \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i$$

$$+ \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i$$

$$+ \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i$$

$$+\tilde{P}(l) \prod_{i}^{m-1} D_i$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

For example, choosing *l* such that

$$D_0(l^\pm) = D_1(l^\pm) =$$
$$= D_2(l^\pm) = D_3(l^\pm) = 0$$

sets all the terms in this equation to zero except the first line

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

For example, choosing *l* such that

$$
\begin{aligned}
D_0(l^{\pm}) = D_1(l^{\pm}) = \\
= D_2(l^{\pm}) = D_3(l^{\pm}) = 0
\end{aligned}
$$

sets all the terms in this equation to zero except the first line

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

For example, choosing *l* such that

$$
\begin{aligned}
D_0(l^\pm) = D_1(l^\pm) = \\
= D_2(l^\pm) = D_3(l^\pm) = 0
\end{aligned}
$$

sets all the terms in this equation to zero except the first line

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

To solve the OPP reduction, choosing special values for the loop momenta helps a lot

For example, choosing *l* such that

$$
\begin{aligned}
D_0(l^{\pm}) = D_1(l^{\pm}) = \\
= D_2(l^{\pm}) = D_3(l^{\pm}) = 0
\end{aligned}
$$

sets all the terms in this equation to zero except the first line

There are two (complex) solutions to this equation due to the quadratic nature of the propagators

How it works...

$$\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}$$

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

Now we choose l such that

$$D_0(l^i) = D_1(l^i) = D_2(l^i) = 0$$

sets all the terms in this equation to zero except the first and second line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now we choose l such that

$$D_0(l^i) = D_1(l^i) = D_2(l^i) = 0$$

sets all the terms in this equation to zero except the first and second line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l) \right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l) \right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l) \right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l) \right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now we choose l such that

$$D_0(l^i) = D_1(l^i) = D_2(l^i) = 0$$

sets all the terms in this equation to zero except the first and second line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l) \right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l) \right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l) \right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l) \right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l) \right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l) \right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l) \right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l) \right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

Now, choosing *l* such that

$$D_0(l^i) = D_1(l^i) = 0$$

sets all the terms in this equation to zero except the first, second and third line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now, choosing *l* such that

$$D_0(l^i) = D_1(l^i) = 0$$

sets all the terms in this equation to zero except the first, second and third line

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l) \right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l) \right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l) \right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l) \right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

Now, choosing *l* such that

$$D_1(l^i) = 0$$

sets the last line to zero

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i \\
& = 0
\end{aligned}
$$

Now, choosing *l* such that

$$D_1(l^i) = 0$$

sets the last line to zero

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0 i_1 i_2 i_3} + \tilde{d}_{i_0 i_1 i_2 i_3}(l)\right] \prod_{i \neq i_0, i_1, i_2, i_3}^{m-1} D \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0 i_1 i_2} + \tilde{c}_{i_0 i_1 i_2}(l)\right] \prod_{i \neq i_0, i_1, i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0 i_1} + \tilde{b}_{i_0 i_1}(l)\right] \prod_{i \neq i_0, i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i \neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

Now, choosing arbitrary *l*

🟪 Coefficient computed in a previous step

How it works...

$$
\begin{aligned}
N(l) = & \sum_{i_0<i_1<i_2<i_3}^{m-1} \left[d_{i_0i_1i_2i_3} + \tilde{d}_{i_0i_1i_2i_3}(l)\right] \prod_{i\neq i_0,i_1,i_2,i_3}^{m-1} D_i \\
& + \sum_{i_0<i_1<i_2}^{m-1} \left[c_{i_0i_1i_2} + \tilde{c}_{i_0i_1i_2}(l)\right] \prod_{i\neq i_0,i_1,i_2}^{m-1} D_i \\
& + \sum_{i_0<i_1}^{m-1} \left[b_{i_0i_1} + \tilde{b}_{i_0i_1}(l)\right] \prod_{i\neq i_0,i_1}^{m-1} D_i \\
& + \sum_{i_0}^{m-1} \left[a_{i_0} + \tilde{a}_{i_0}(l)\right] \prod_{i\neq i_0}^{m-1} D_i \\
& + \tilde{P}(l) \prod_{i}^{m-1} D_i
\end{aligned}
$$

We have our Numerator!

🟪 Coefficient computed in a previous step

How it works...

- For each phase-space point we have to solve the system of equations
- Due to the fact that the system reduces when picking special values for the loop momentum, the system greatly reduces
- For a given phase-space point, we have to compute the numerator function several times (~50 or so for a box loop)
  - Trick can be used here (OpenLoop method)

d dimensions

- In the previous consideration I was very sloppy in considering if we are working in 4 or d dimensions
- In general, external momenta and polarization vectors are in 4 dimensions; only the loop momentum is in d dimensions

- To be more correct, we compute the integral

$$\int d^d l \, \frac{N(l,\tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

$$\boxed{\underset{\substack{\nearrow \\ \text{d dim}}}{\bar{l}} = \underset{\substack{\uparrow \\ \text{4 dim}}}{l} + \underset{\substack{\nwarrow \\ \text{epsilon dim}}}{\tilde{l}}}$$

$$\bar{D}_i = (\bar{l} + p_i)^2 - m_i^2 = (l + p_i)^2 - m_i^2 + \tilde{l}^2 = D_i + \tilde{l}^2$$

$$l \cdot \tilde{l} = 0 \qquad \bar{l} \cdot p_i = l \cdot p_i \qquad \bar{l} \cdot \bar{l} = l \cdot l + \tilde{l} \cdot \tilde{l}$$

Implications

- The decomposition in terms of scalar integrals has to be done in d dimensions
- This is why the rational part R is needed

$$
\begin{aligned}
& \sum_{0\leq i_0<i_1<i_2<i_3}^{m-1} d(i_0 i_1 i_2 i_3) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}\bar{D}_{i_2}\bar{D}_{i_3}} \\
+ & \sum_{0\leq i_0<i_1<i_2}^{m-1} c(i_0 i_1 i_2) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}\bar{D}_{i_2}} \\
+ & \sum_{0\leq i_0<i_1}^{m-1} b(i_0 i_1) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}\bar{D}_{i_1}} \\
+ & \sum_{i_0=0}^{m-1} a(i_0) \int d^d\bar{\ell}\, \frac{1}{\bar{D}_{i_0}} \\
+ & R\,.
\end{aligned}
$$

Rational terms

$$\int d^d l \, \frac{N(l, \tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

$$R = R_1 + R_2$$

Rational terms

$$\int d^d l \, \frac{N(l,\tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

- They are split into two contributions, generally called

$$R = R_1 + R_2$$

- Both have their origin in the UV part of the model,

Rational terms

$$\int d^d l \, \frac{N(l,\tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

- They are split into two contributions, generally called

$$R = \boxed{R_1} + R_2$$

- Both have their origin in the UV part of the model,
  - R1: originates from the propagator (calculate on the flight)

Rational terms

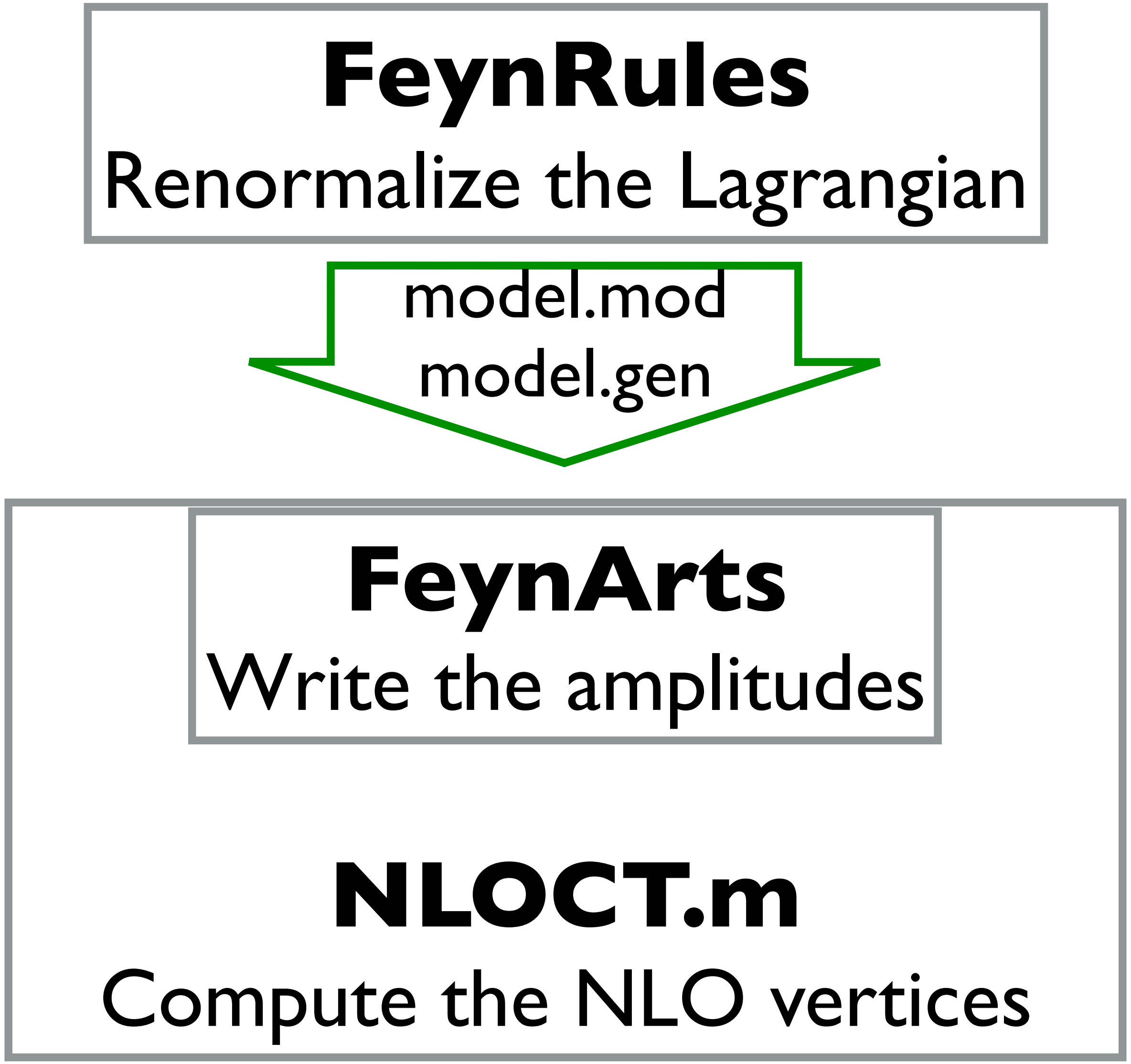
$$\int d^d l \, \frac{N(l,\tilde{l})}{\bar{D}_0 \bar{D}_1 \bar{D}_2 \cdots \bar{D}_{m-1}}$$

- They are split into two contributions, generally called

$$R = R_1 + \enclose{circle}{R_2}$$

- Both have their origin in the UV part of the model,
  - R1: originates from the propagator (calculate on the flight)
  - R2: originates from the numerator (need in the model)

How does it work?



FeynRules

Renormalize the Lagrangian

model.mod
model.gen

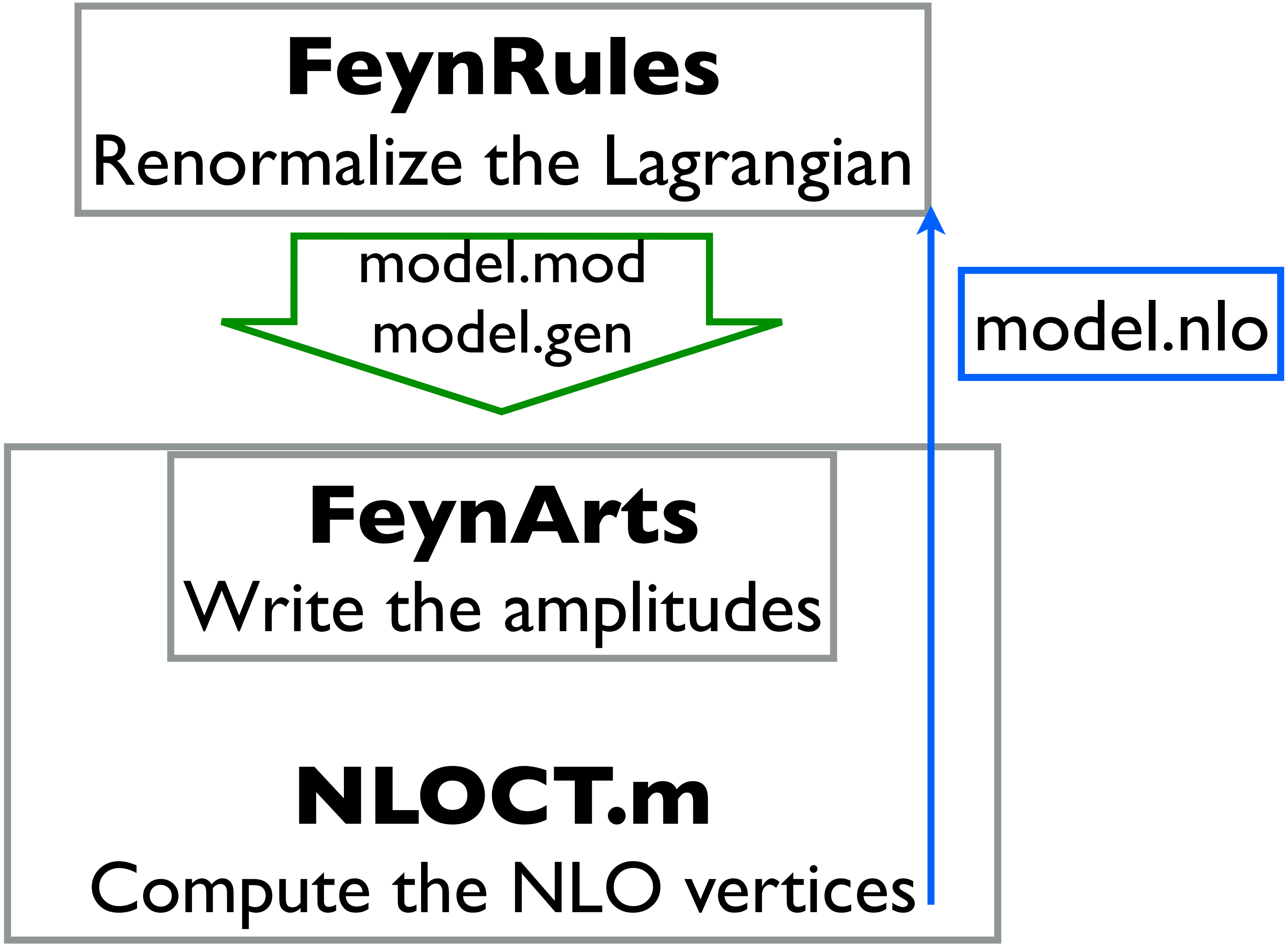
FeynArts

Write the amplitudes

NLOCT.m

Compute the NLO vertices

How does it work?



FeynRules

Renormalize the Lagrangian

model.mod
model.gen

FeynArts

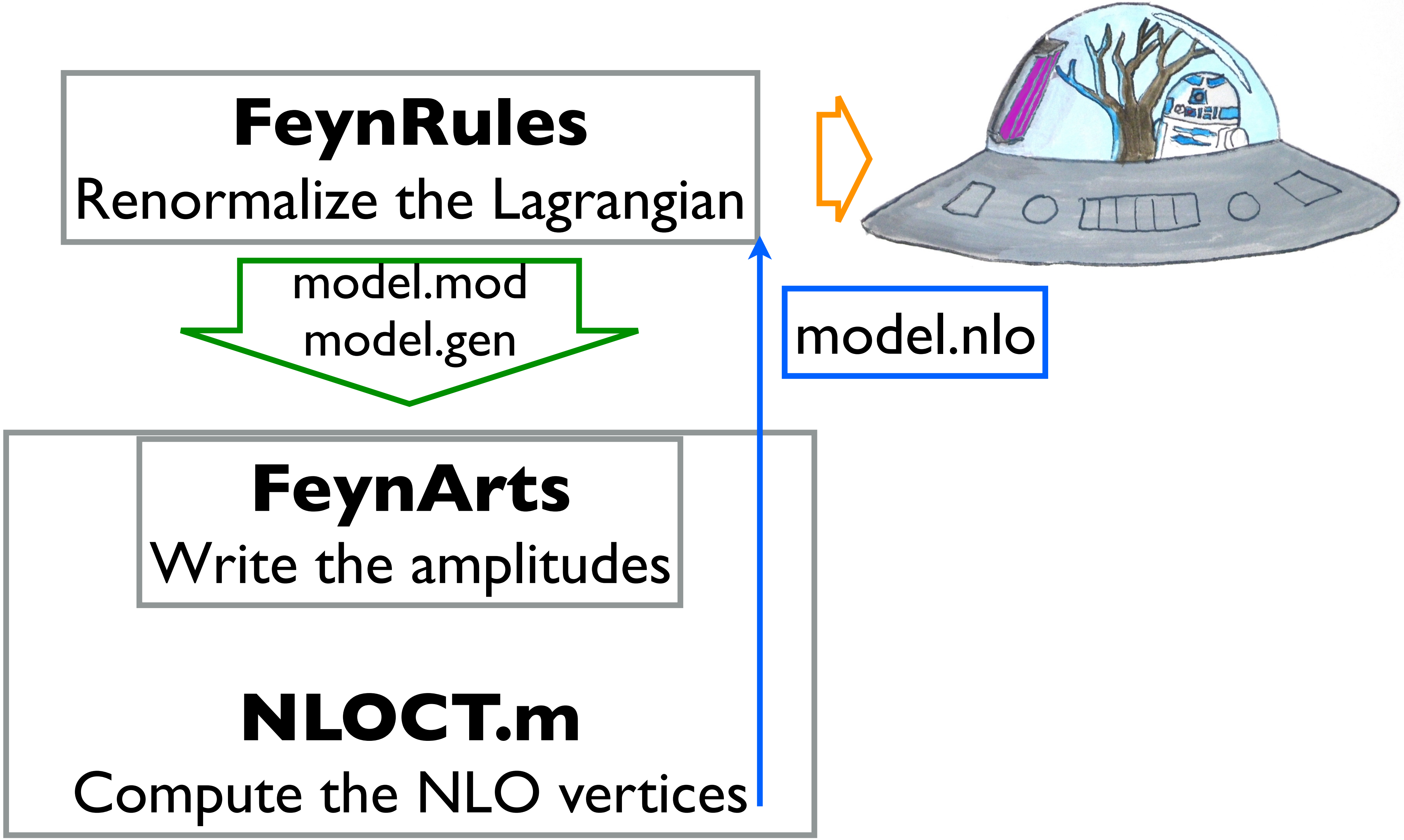
Write the amplitudes

NLOCT.m

Compute the NLO vertices

model.nlo

How does it work?



FeynRules

Renormalize the Lagrangian

model.mod

model.gen

FeynArts

Write the amplitudes

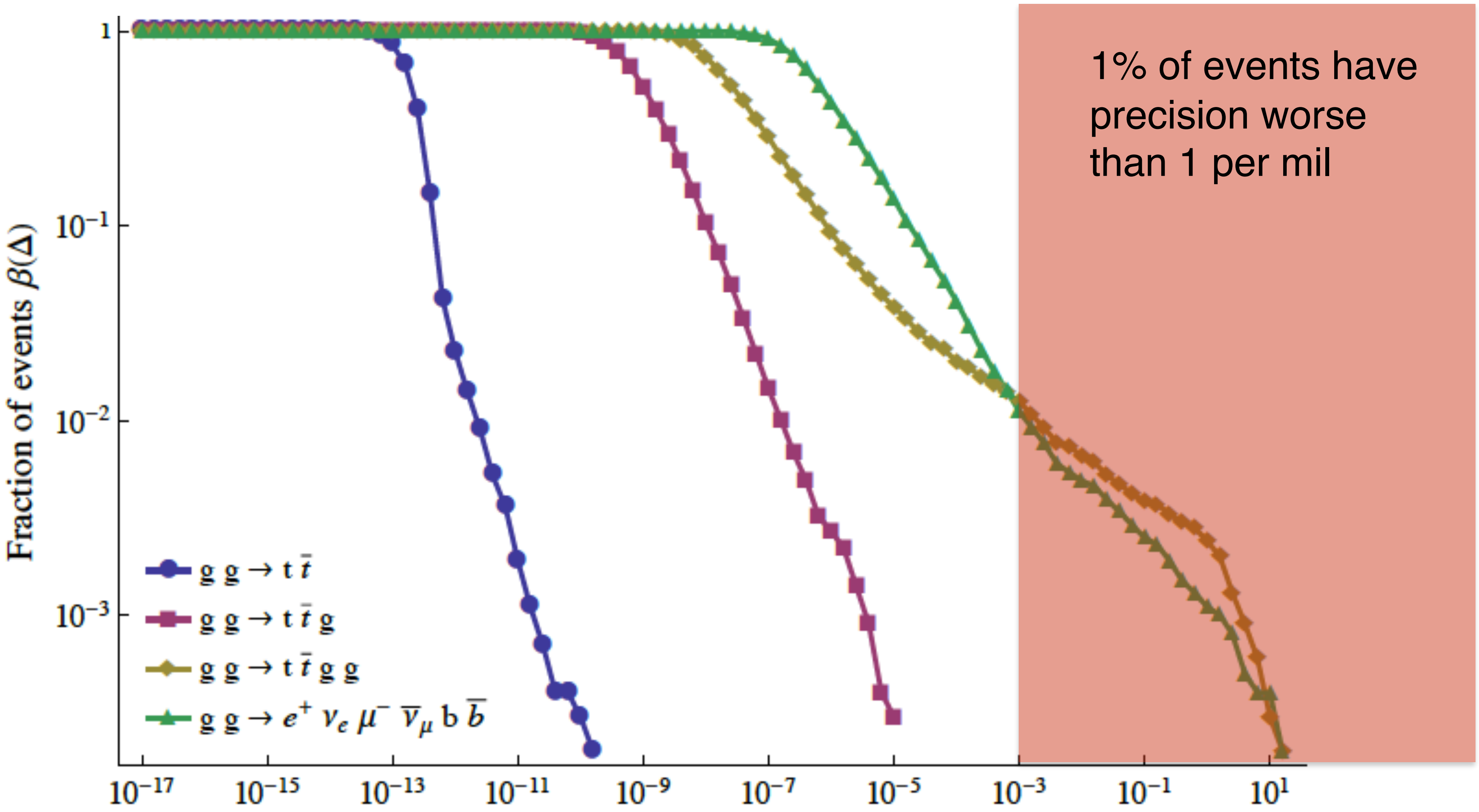
NLOCT.m

Compute the NLO vertices

model.nlo

C. Degrande

Numerical Stability



1% of events have precision worse than 1 per mil

- For 2 to 4 processes, ~7% of the Phase-space point have a precision worse than 1e-3

➡ Previous solution pass to quadruple precision (extremelly slow)

Stability

Quadruple precision

- Very slow (100 times slower)
- 1% unstable point means 50% of the time is used in those points
- Stability curve are crucial for comparing code efficiency

Stability

Quadruple precision

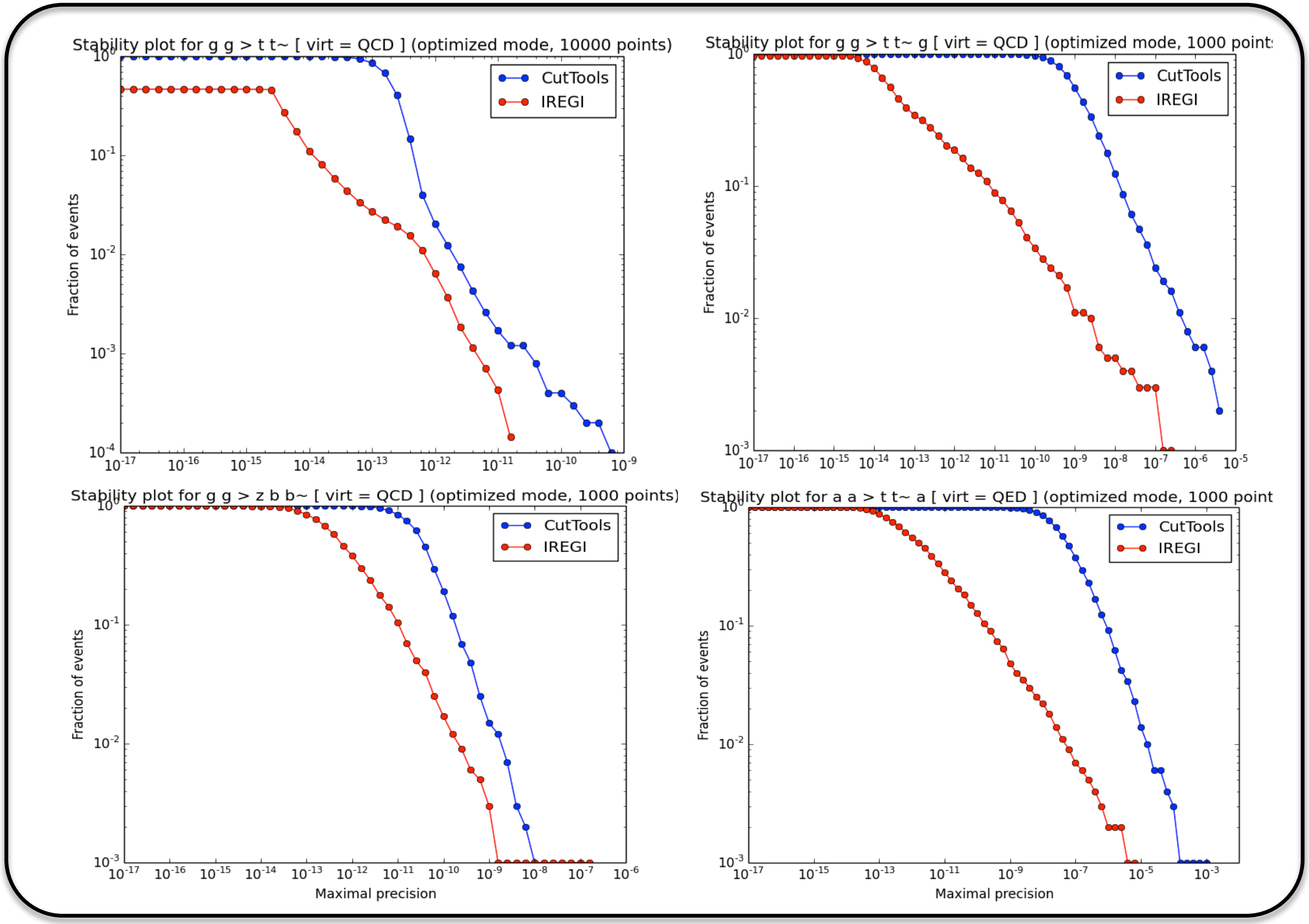
- Very slow (100 times slower)
- 1% unstable point means 50% of the time is used in those points
- Stability curve are crucial for comparing code efficiency

Avoid Quadruple precision

- Use another method (TIR instead of OPP) to evaluate the loop reduces the need of quadruple precision

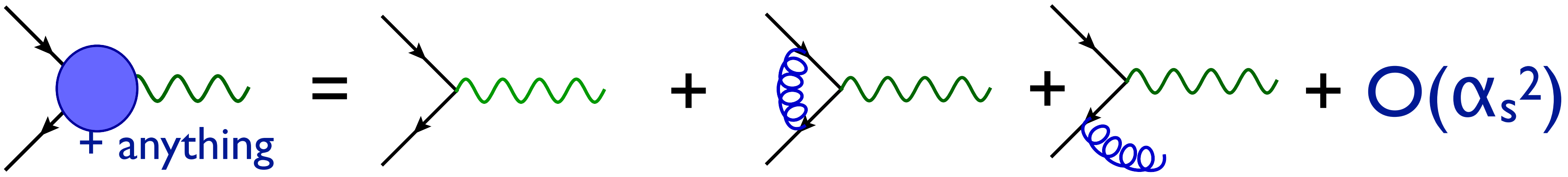
IREGI

- New Solution use IREGI: a TIR program
  - ➡ Slower than previous method but faster than quadruple precision
  - ➡Usually less uncertainty (and not for the same PS point)



[H.-shao]

Difficulties



- **3 questions:**
    - Virtual amplitudes: how to compute the loops automatically in a reasonable amount of time
    - How to deal with divergencies for phase-space integration
    - How to match these processes to a parton shower without double counting

Dealing with divergencies

More details in S. Schuman lectures

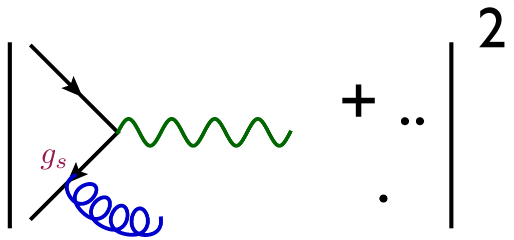
Example

$\int_0^1 dx\, f(x)$ $f(x) = \frac{g(x)}{x}$ $g(x)$ Finite everywhere

Example

$$\int_0^1 dx\, f(x) \qquad f(x) = \frac{g(x)}{x} \qquad g(x) \text{ Finite everywhere}$$

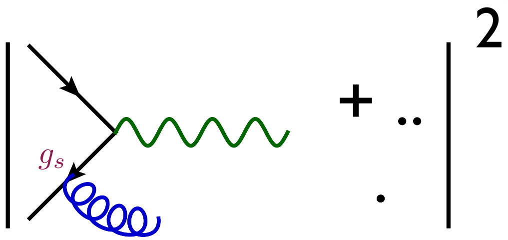
- Type of Divergencies of the real



Example

$$\int_0^1 dx\, f(x) \qquad f(x) = \frac{g(x)}{x} \qquad g(x)$$ Finite everywhere

- Type of Divergencies of the real



- Let's introduce a regulator

$$\lim_{\epsilon \longrightarrow 0} \int_0^1 dx\, \frac{g(x)}{x^{1+\epsilon}} = \lim_{\epsilon \longrightarrow 0} \int_0^1 dx\, x^{-\epsilon} f(x)$$

for any non-integer non-zero value for ϵ this integral is finite

- We would like to factor out the explicit poles in ϵ so that they can be canceled explicitly against the virtual corrections

Phase-Space Slicing

- We introduce a small parameter δ≪1:

$$\lim_{\varepsilon\to 0}\int_0^1 dx x^\varepsilon f(x) = \lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}}$$

$$\begin{aligned}
\lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}} &= \lim_{\varepsilon\to 0}\left(\int_0^\delta dx \frac{g(x)}{x^{1-\varepsilon}} + \int_\delta^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&\simeq \lim_{\varepsilon\to 0}\left(\int_0^\delta dx \frac{g(0)}{x^{1-\varepsilon}} + \int_\delta^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&= \lim_{\varepsilon\to 0}\frac{\delta^\varepsilon}{\varepsilon} g(0) + \int_\delta^1 dx \frac{g(x)}{x} \\
&= \lim_{\varepsilon\to 0}\left(\frac{1}{\varepsilon} + \log\delta\right) g(0) + \int_\delta^1 dx \frac{g(x)}{x}
\end{aligned}$$

Phase-Space Slicing

- We introduce a small parameter δ≪1:

$$\lim_{\varepsilon\to 0}\int_0^1 dx x^{\varepsilon} f(x) = \lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}}$$

$$\begin{aligned}
\lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}} &= \lim_{\varepsilon\to 0}\left(\int_0^{\delta} dx \frac{g(x)}{x^{1-\varepsilon}} + \int_{\delta}^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&\simeq \lim_{\varepsilon\to 0}\left(\int_0^{\delta} dx \frac{g(0)}{x^{1-\varepsilon}} + \int_{\delta}^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&= \lim_{\varepsilon\to 0}\frac{\delta^{\varepsilon}}{\varepsilon} g(0) + \int_{\delta}^1 dx \frac{g(x)}{x} \\
&= \lim_{\varepsilon\to 0}\left(\underset{\textbf{Pole}}{\boxed{\frac{1}{\varepsilon}}} + \log\delta\right) g(0) + \int_{\delta}^1 dx \frac{g(x)}{x}
\end{aligned}$$

Phase-Space Slicing

- We introduce a small parameter δ≪1:

$$\lim_{\varepsilon\to 0}\int_0^1 dx x^{\varepsilon} f(x) = \lim_{\varepsilon\to 0}\int_0^1 dx\frac{g(x)}{x^{1-\varepsilon}}$$

$$\begin{aligned}
\lim_{\varepsilon\to 0}\int_0^1 dx\frac{g(x)}{x^{1-\varepsilon}} &= \lim_{\varepsilon\to 0}\left(\int_0^{\delta} dx\frac{g(x)}{x^{1-\varepsilon}} + \int_{\delta}^1 dx\frac{g(x)}{x^{1-\varepsilon}}\right)\\
&\simeq \lim_{\varepsilon\to 0}\left(\int_0^{\delta} dx\frac{g(0)}{x^{1-\varepsilon}} + \int_{\delta}^1 dx\frac{g(x)}{x^{1-\varepsilon}}\right)\\
&= \lim_{\varepsilon\to 0}\frac{\delta^{\varepsilon}}{\varepsilon}g(0) + \int_{\delta}^1 dx\frac{g(x)}{x}\\
&= \lim_{\varepsilon\to 0}\left(\boxed{\frac{1}{\varepsilon}} + \log\delta\right)g(0) + \boxed{\int_{\delta}^1 dx\frac{g(x)}{x}}
\end{aligned}$$

Finite peace

Pole

Phase-Space Slicing

- We introduce a small parameter δ≪1:

$$\lim_{\varepsilon\to 0}\int_0^1 dx x^\varepsilon f(x) = \lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}}$$

$$\begin{aligned}
\lim_{\varepsilon\to 0}\int_0^1 dx \frac{g(x)}{x^{1-\varepsilon}} &= \lim_{\varepsilon\to 0}\left(\int_0^\delta dx \frac{g(x)}{x^{1-\varepsilon}} + \int_\delta^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&\simeq \lim_{\varepsilon\to 0}\left(\int_0^\delta dx \frac{g(0)}{x^{1-\varepsilon}} + \int_\delta^1 dx \frac{g(x)}{x^{1-\varepsilon}}\right) \\
&= \lim_{\varepsilon\to 0}\frac{\delta^\varepsilon}{\varepsilon} g(0) + \int_\delta^1 dx \frac{g(x)}{x} \\
&= \lim_{\varepsilon\to 0} \boxed{\Big(\frac{1}{\varepsilon}} \boxed{+\log\delta\Big) g(0)} + \boxed{\int_\delta^1 dx \frac{g(x)}{x}}
\end{aligned}$$

Finite peace

Pole **Large cancelation**

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad f(x) = \frac{g(x)}{x}$$

- Add and subtract the same term

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) = \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]$$

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x) = \frac{g(x)}{x}$$

- Add and subtract the same term

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) = \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]$$

$$= \lim_{\epsilon\to 0}\int_0^1 dx\left[g(0)\frac{x^{-\epsilon}}{x} + \frac{g(x)-g(0)}{x^{1+\epsilon}}\right]$$

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x) = \frac{g(x)}{x}$$

- Add and subtract the same term

$$\begin{aligned}\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) &= \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]\\ &= \lim_{\epsilon\to 0}\int_0^1 dx \left[g(0)\frac{x^{-\epsilon}}{x} + \frac{g(x)-g(0)}{x^{1+\epsilon}}\right]\\ &= \lim_{\epsilon\to 0}\frac{-1}{\epsilon}g(0) + \int_0^1 dx\,\frac{g(x)-g(0)}{x}\end{aligned}$$

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x) = \frac{g(x)}{x}$$

- Add and subtract the same term

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) = \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]$$

$$= \lim_{\epsilon\to 0}\int_0^1 dx \left[g(0)\frac{x^{-\epsilon}}{x} + \frac{g(x)-g(0)}{x^{1+\epsilon}}\right]$$

$$= \lim_{\epsilon\to 0}\boxed{\frac{-1}{\epsilon}}\, g(0) + \int_0^1 dx\, \frac{g(x)-g(0)}{x}$$

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x) = \frac{g(x)}{x}$$

- Add and subtract the same term

$$\begin{aligned}\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) &= \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]\\ &= \lim_{\epsilon\to 0}\int_0^1 dx\left[g(0)\frac{x^{-\epsilon}}{x} + \frac{g(x)-g(0)}{x^{1+\epsilon}}\right]\\ &= \lim_{\epsilon\to 0}\boxed{\frac{-1}{\epsilon}}g(0) + \boxed{\int_0^1 dx\,\frac{g(x)-g(0)}{x}}\end{aligned}$$

Subtraction method

$$\lim_{\epsilon\longrightarrow 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x)=\frac{g(x)}{x}$$

- Add and subtract the same term

$$\begin{aligned}\lim_{\epsilon\longrightarrow 0}\int_0^1 dx\, x^{-\epsilon} f(x) &= \lim_{\epsilon\longrightarrow 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x}+f(x)-\frac{g(0)}{x}\right]\\ &= \lim_{\epsilon\longrightarrow 0}\int_0^1 dx\left[g(0)\frac{x^{-\epsilon}}{x}+\frac{g(x)-g(0)}{x^{1+\epsilon}}\right]\\ &= \lim_{\epsilon\longrightarrow 0}\boxed{\frac{-1}{\epsilon}}\,g(0)+\boxed{\int_0^1 dx\,\frac{g(x)-g(0)}{x}}\end{aligned}$$

- We have factored out the $1/\epsilon$ divergence and are left with a finite integral

Subtraction method

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) \qquad\qquad f(x)=\frac{g(x)}{x}$$

- Add and subtract the same term

$$\lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon} f(x) = \lim_{\epsilon\to 0}\int_0^1 dx\, x^{-\epsilon}\left[\frac{g(0)}{x} + f(x) - \frac{g(0)}{x}\right]$$

$$= \lim_{\epsilon\to 0}\int_0^1 dx \left[g(0)\frac{x^{-\epsilon}}{x} + \frac{g(x)-g(0)}{x^{1+\epsilon}}\right]$$

$$= \lim_{\epsilon\to 0} \boxed{\frac{-1}{\epsilon}}\, g(0) + \boxed{\int_0^1 dx\, \frac{g(x)-g(0)}{x}}$$

- We have factored out the $1/\epsilon$ divergence and are left with a finite integral

- According to the KLN theorem the divergence cancels against the virtual corrections

To Remember

- In both cases the pole is extracted and we end up with a finite remainder:

$$g(0)\log\delta + \int_\delta^1 dx\,\frac{g(x)}{x}$$

$$\int_0^1 dx\,\frac{g(x)-g(0)}{x}$$

- Subtraction acts like a plus distribution
- Slicing works only for small δ, and one has to prove the δ-independence of cross section and distribution; subtraction is exact
- In both methods there are cancelation between large numbers. If for a given observable $\lim_{x\to 0} O(x) \neq O(0)$ or we choose a too small bin size, instabilities will arise (we cannot ask for an infinite resolution)
- Subtraction is more flexible: good for automation

NLO with Counter-term

$$\sigma_{NLO} = \int d^4\Phi_n \mathcal{B} + \int d^4\Phi_n \mathcal{V} + \int d^4\Phi_{n+1} \mathcal{R}$$

- With the subtraction terms the expression becomes

$$\sigma_{NLO} = \int d^4\Phi_n \mathcal{B}$$

$$+ \int d^4\Phi_n \left(\mathcal{V} + \int d^d\Phi_1 \mathcal{C} \right)_{\varepsilon \to 0}$$

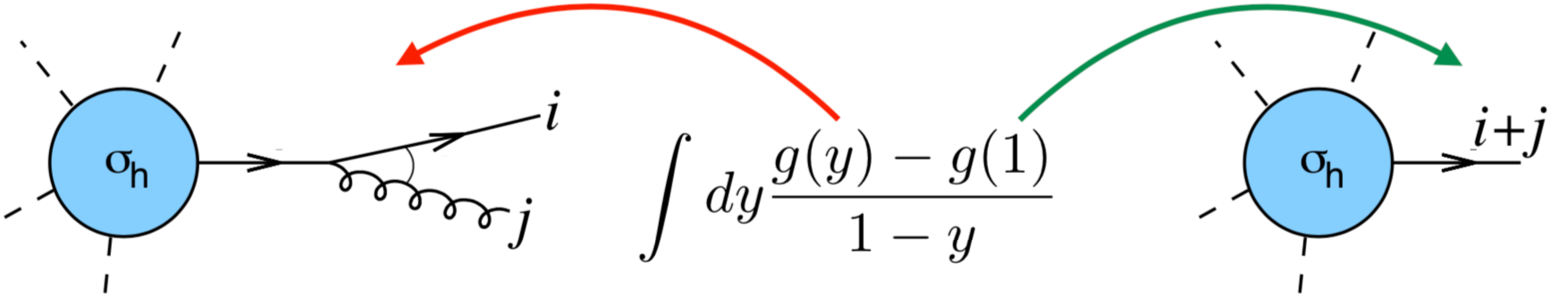
Poles cancel from d-dim integration

$$+ \int d^4\Phi_{n+1} \left(\mathcal{R} - \mathcal{C} \right)$$

Integrand is finite in 4 dimension

- Terms in brackets are finite and can be integrated numerically in d=4 and independently one from another

Kinematics of counter events



$$\int dy \frac{g(y) - g(1)}{1 - y}$$

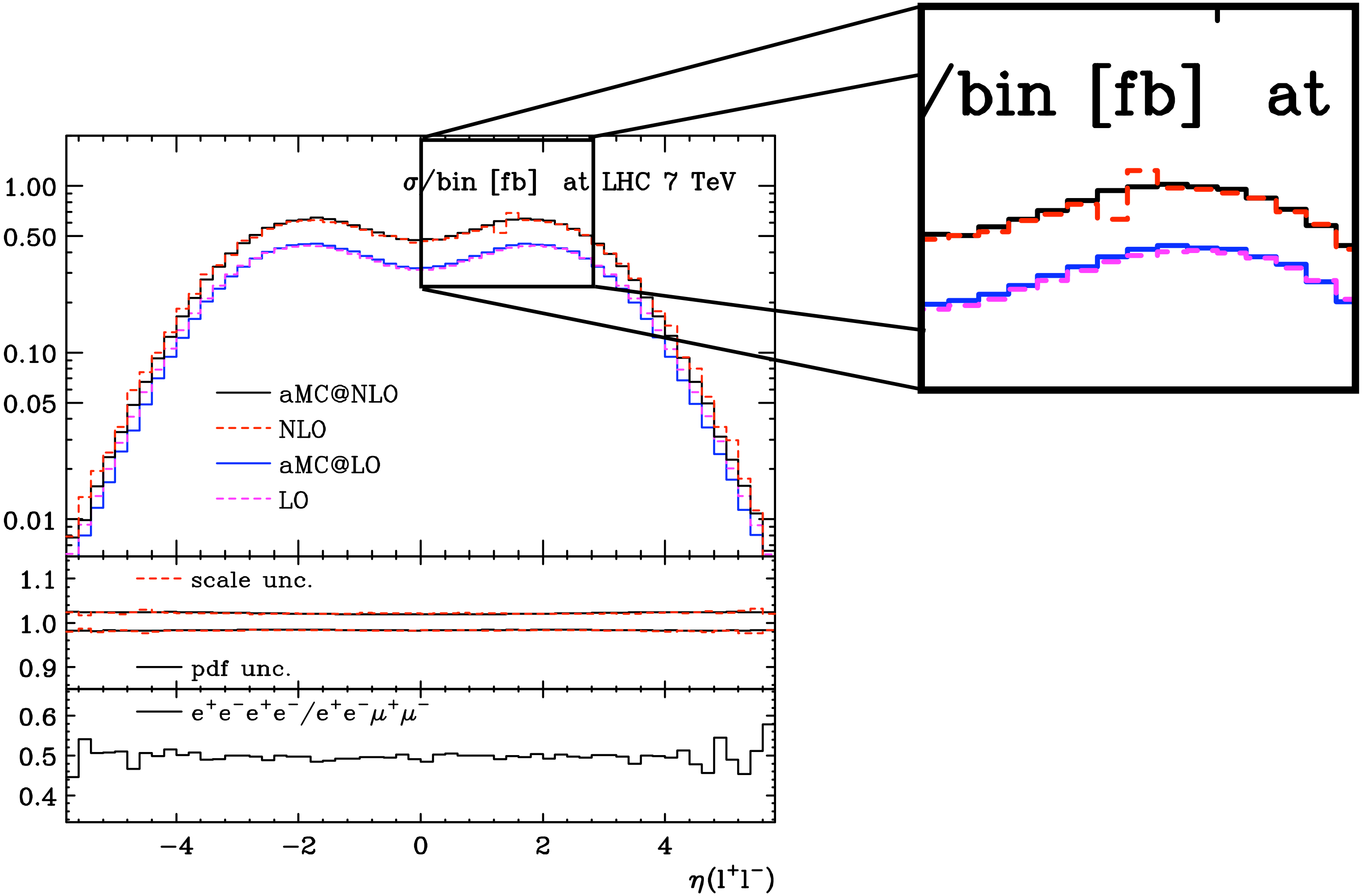
Real emission

Subtraction term

- If i and j are on-shell in the event, for the counterevent the combined particle $i+j$ must be on shell
- $i+j$ can be put on shell only be reshuffling the momenta of the other particles
- It can happen that event and counterevent end up in different histogram bins
    - Use IR-safe observables and don't ask for infinite resolution!
    - Still, these precautions do not eliminate the problem…

4 charged lepton

- The NLO results shows a typical peak-dip structure that hampers fixed order calculations



Event Generation?

- Another consequence of the kinematic mismatch is that we cannot generate events at NLO
- n+1-body contribution and n-body contribution are not bounded from above → unweighting not possible
- Further ambiguity on which kinematics to use for the unweighted events

Event Generation?

- Another consequence of the kinematic mismatch is that we cannot generate events at NLO
- $n+1$-body contribution and n-body contribution are not bounded from above → unweighting not possible
- Further ambiguity on which kinematics to use for the unweighted events

Histogram on the flight

- In practice, two set of momenta are generated during the MC integration
  - A n-body set, for Born, virtuals and counterterms
  - A $n+1$-body set, for the real emission
- The various terms are computed. Cuts are applied on the corresponding momenta and histograms are filled with the weight and kinematics of each term

To Remember

- Virtual and real matrix element are not finite, but their sum is. Subtraction methods can be used to extract divergences for real-emission matrix elements and cancel explicitly the poles from the virtuals

To Remember

- Virtual and real matrix element are not finite, but their sum is. Subtraction methods can be used to extract divergences for real-emission matrix elements and cancel explicitly the poles from the virtuals
- Event and counterevents have different kinematics. Unweighting is not possible, we need to fill plots on-the-fly with weighted events

To Remember

- Virtual and real matrix element are not finite, but their sum is. Subtraction methods can be used to extract divergences for real-emission matrix elements and cancel explicitly the poles from the virtuals
- Event and counterevents have different kinematics. Unweighting is not possible, we need to fill plots on-the-fly with weighted events
- For plots, only IR-safe observable with finite resolution must be used!

aMC@NLO

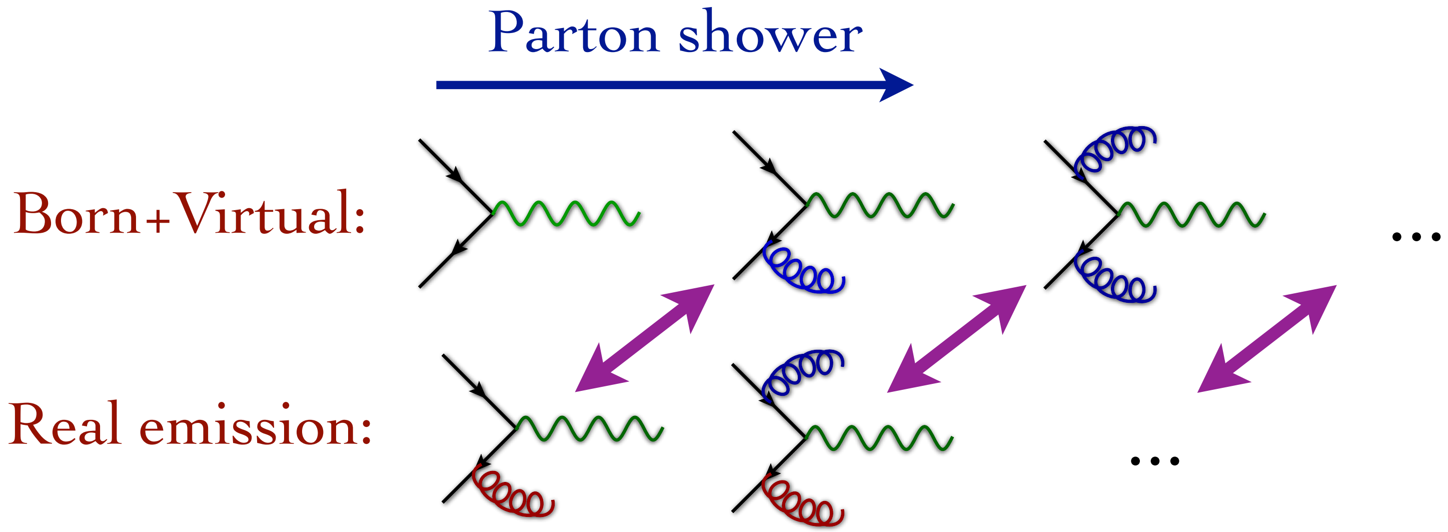
Matching NLO

- GOAL: We want to allow to have PS on NLO sample

Matching NLO

- **GOAL**: We want to allow to have PS on NLO sample

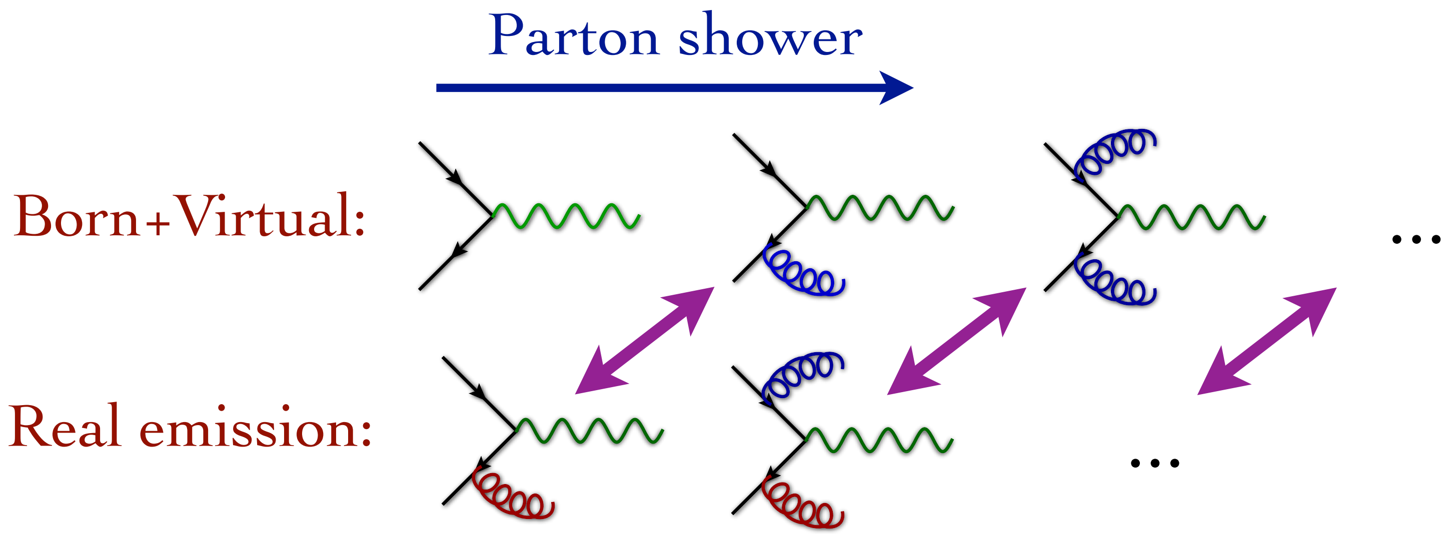
- At NLO one faces double-counting issues:



Matching NLO

- GOAL: We want to allow to have PS on NLO sample

- At NLO one faces double-counting issues:



- And also part of the virtual contribution is double counted through the definition of the Sudakov factor Δ

Double counting

- Since **Δ** = 1 - *P*, **Δ** contains contributions from the virtual corrections implicitly
- Because at NLO the virtual corrections are already included via explicit matrix elements, **Δ** is double counting with the virtual corrections
- In fact, because the shower is unitary, what we are double counting in the real emission corrections is exactly equal to what we are double counting in the virtual corrections (but with opposite sign)!

Attach Parton-Shower

$$\frac{d\sigma_{``NLO"}}{dO} = \left[\mathcal{B} + \mathcal{V}\right] d\Phi_n \qquad\qquad + d\Phi_{n+1}\mathcal{R}$$

Attach Parton-Shower

$$\frac{d\sigma_{``NLO}}{dO} = \left[\mathcal{B} + \mathcal{V}\right] d\Phi_n \qquad\qquad + d\Phi_{n+1}\mathcal{R}$$

- **What's wrong?**
  - Let's expand this at first order in the strong coupling

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B} + \mathcal{V}]\, d\Phi_n\, I_{MC}^{n}(O) + d\Phi_{n+1}\mathcal{R}\, I_{MC}^{n+1}(O)$$

- **What's wrong?**
  - Let's expand this at first order in the strong coupling

- **Let Focus on the parton shower part**
  - First order means zero or one emission

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n\, I_{MC}^{n}(O) + d\Phi_{n+1}\mathcal{R}\, I_{MC}^{n+1}(O)$$

- **What’s wrong?**
  - Let’s expand this at first order in the strong coupling

- **Let Focus on the parton shower part**
  - First order means zero or one emission

$$I_{MC} = \Delta_a\left(Q,Q_0\right) + \Delta_a\left(Q,Q_0\right) d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n\, I^{n}_{MC}(O) + d\Phi_{n+1}\mathcal{R}\, I^{n+1}_{MC}(O)$$

- **What's wrong?**
  - Let's expand this at first order in the strong coupling

- **Let Focus on the parton shower part**
  - First order means zero or one emission

$$I_{MC} = \Delta_a(Q, Q_0) + \Delta_a(Q, Q_0)\, d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

$$\Delta_a(Q, Q_0) = \exp\left[-\int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}\right] \simeq 1 - \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = \left[\mathcal{B}+\mathcal{V}\right] d\Phi_n \, I_{MC}^{n}(O) + d\Phi_{n+1}\mathcal{R}\, I_{MC}^{n+1}(O)$$

- **What's wrong?**
  - Let's expand this at first order in the strong coupling

- **Let Focus on the parton shower part**
  - First order means zero or one emission

$$I_{MC} = \Delta_a\left(Q,Q_0\right) + \Delta_a\left(Q,Q_0\right) d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

$$\Delta_a\left(Q,Q_0\right) = \exp\left[-\int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}\right] \simeq 1 - \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

$$I_{MC} \simeq 1 - \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc} + d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n\, I^n_{MC}(O) + d\Phi_{n+1}\mathcal{R}\, I^{n+1}_{MC}(O)$$

- **What's wrong?**
  - Let's expand this at first order in the strong coupling

$$I_{MC} \simeq 1 - \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc} + d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n + d\Phi_{n+1}\mathcal{R}$$

Expected result

$$- \mathcal{B}\, d\Phi_n \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc} + \mathcal{B}\, d\Phi_n d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

NLO breaking term (cancelling for inclusive observables)

Attach Parton-Shower

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n\, I^n_{MC}(O) + d\Phi_{n+1}\mathcal{R}\, I^{n+1}_{MC}(O)$$

- **What’s wrong?**
  - Let’s expand this at first order in the strong coupling

$$I_{MC} \simeq 1 - \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc} + d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

$$\frac{d\sigma_{\text{“}NLO+PS\text{”}}}{dO} = [\mathcal{B}+\mathcal{V}]\, d\Phi_n + d\Phi_{n+1}\mathcal{R}$$

Expected result

$$- \mathcal{B}\, d\Phi_n \int d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc} + \mathcal{B}\, d\Phi_n d\Phi_1 \frac{\alpha_s(t)}{2\pi} P_{a\to bc}$$

NLO breaking term (cancelling for inclusive observables)

MC@NLO procedure

[Frixione & Webber (2002)]

- To remove the double counting, we can add and subtract the same term to the *m* and *m*+1 body configurations

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m(B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC)\right] I_{\mathrm{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$

MC@NLO procedure

[Frixione & Webber (2002)]

- To remove the double counting, we can add and subtract the same term to the *m* and *m*+1 body configurations

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m (B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC)\right] I_{\mathrm{MC}}^{(m)}(O) + \left[d\Phi_{m+1}(R - MC)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$

- Where the *MC* are defined to be the contribution of the parton shower to get from the *m* body Born final state to the *m*+1 body real emission final state

MC@NLO procedure

[Frixione & Webber (2002)]

- To remove the double counting, we can add and subtract the same term to the *m* and *m*+1 body configurations

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m \left(B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC \right) \right] I_{\mathrm{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1} \left(R - MC \right) \right] I_{\mathrm{MC}}^{(m+1)}(O)$$

- Where the *MC* are defined to be the contribution of the parton shower to get from the *m* body Born final state to the *m*+1 body real emission final state

$$MC = \left| \frac{\partial \left(t^{MC}, z^{MC}, \phi \right)}{\partial \Phi_1} \right| \frac{1}{t^{MC}} \frac{\alpha_s}{2\pi} \frac{1}{2\pi} P\left(z^{MC} \right) \mathcal{B}$$

MC@NLO procedure

[Frixione & Webber (2002)]

- To remove the double counting, we can add and subtract the same term to the *m* and *m*+1 body configurations

$$\frac{d\sigma_{\rm NLOwPS}}{dO} = \left[d\Phi_m\left(B + \int_{\rm loop} V + \int d\Phi_1 MC\right)\right] I_{\rm MC}^{(m)}(O) + \left[d\Phi_{m+1}(R - MC)\right] I_{\rm MC}^{(m+1)}(O)$$

- Where the *MC* are defined to be the contribution of the parton shower to get from the *m* body Born final state to the *m*+1 body real emission final state

$$MC = \left|\frac{\partial\left(t^{MC}, z^{MC}, \phi\right)}{\partial\Phi_1}\right| \boxed{\frac{1}{t^{MC}}\frac{\alpha_s}{2\pi}\frac{1}{2\pi}P\left(z^{MC}\right)} \mathcal{B}$$

Double counting avoided

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m\left(B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC\right)\right] I_{\mathrm{MC}}^{(m)}(O)$$
$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$

- Expanded at NLO

Double counting avoided

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m (B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC)\right] I_{\mathrm{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$

- Expanded at NLO

$$I_{\mathrm{MC}}^{(m)}(O)dO = 1 - \int d\Phi_1 \frac{MC}{B} + d\Phi_1 \frac{MC}{B} + \ldots$$

Double counting avoided

$$\frac{d\sigma_{\rm NLOwPS}}{dO} = \left[d\Phi_m\left(B + \int_{\rm loop} V + \int d\Phi_1 MC\right)\right] I_{\rm MC}^{(m)}(O)$$
$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\rm MC}^{(m+1)}(O)$$

- Expanded at NLO

$$I_{\rm MC}^{(m)}(O)dO = 1 - \int d\Phi_1 \frac{MC}{B} + d\Phi_1 \frac{MC}{B} + \ldots$$

$$\frac{d\sigma_{\text{“}MC@NLO\text{”}}}{dO} = \left[\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right] d\Phi_n + d\Phi_{n+1}\left[\mathcal{R} - MC\right]$$
$$+ \quad \left[- \int d\Phi_1 MC + d\Phi_1 MC\right] d\Phi_n$$

Double counting avoided

$$\frac{d\sigma_{\text{NLOwPS}}}{dO} = \left[d\Phi_m\left(B + \int_{\text{loop}} V + \int d\Phi_1 MC\right)\right] I_{\text{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\text{MC}}^{(m+1)}(O)$$

- Expanded at NLO

$$I_{\text{MC}}^{(m)}(O)dO = 1 - \int d\Phi_1 \frac{MC}{B} + d\Phi_1 \frac{MC}{B} + \dots$$

$$\frac{d\sigma_{\text{``}MC@NLO\text{''}}}{dO} = \left[\mathcal{B} + \mathcal{V} + \cancel{\int d\Phi_1 MC}\right] d\Phi_n + d\Phi_{n+1}\left[\mathcal{R} - MC\right]$$

$$+ \left[\cancel{-\int d\Phi_1 MC} + d\Phi_1 MC\right] d\Phi_n$$

Double counting avoided

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m\left(B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC\right)\right] I_{\mathrm{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$

- Expanded at NLO

$$I_{\mathrm{MC}}^{(m)}(O)dO = 1 - \int d\Phi_1 \frac{MC}{B} + d\Phi_1 \frac{MC}{B} + \ldots$$

$$\frac{d\sigma_{``MC@NLO"}}{dO} = \left[\mathcal{B} + \mathcal{V} + \cancel{\int d\Phi_1 MC}\right] d\Phi_n + d\Phi_{n+1}\left[\mathcal{R} - \cancel{MC}\right]$$

$$+ \left[\cancel{-\int d\Phi_1 MC} + \cancel{d\Phi_1 MC}\right] d\Phi_n$$

MC@NLO properties

- Good features of including the MC counter terms

  1. **Double counting avoided:** The rate expanded at NLO coincides with the total NLO cross section

MC@NLO properties

- Good features of including the MC counter terms

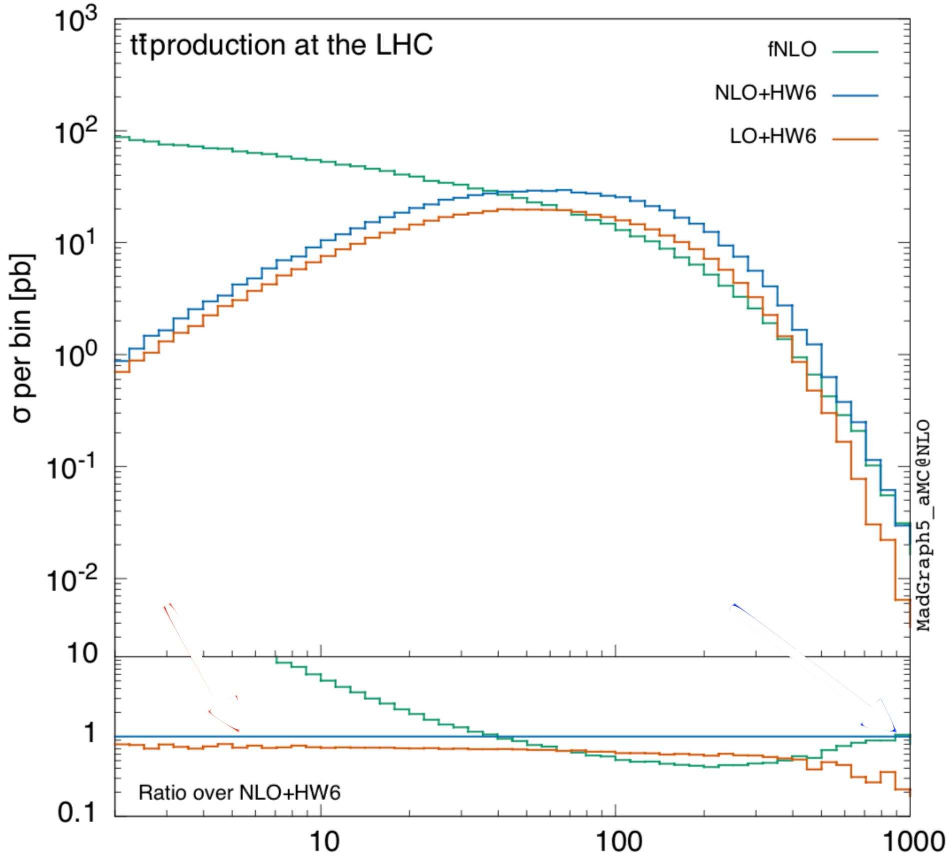
  1. **Double counting avoided**: The rate expanded at NLO coincides with the total NLO cross section

  2. **Smooth matching**: MC@NLO coincides (in shape) with the parton shower in the soft/collinear region, while it agrees with the NLO in the hard region

Matching

$$\frac{d\sigma_{\mathrm{NLOwPS}}}{dO} = \left[d\Phi_m\left(B + \int_{\mathrm{loop}} V + \int d\Phi_1 MC\right)\right] I_{\mathrm{MC}}^{(m)}(O)$$

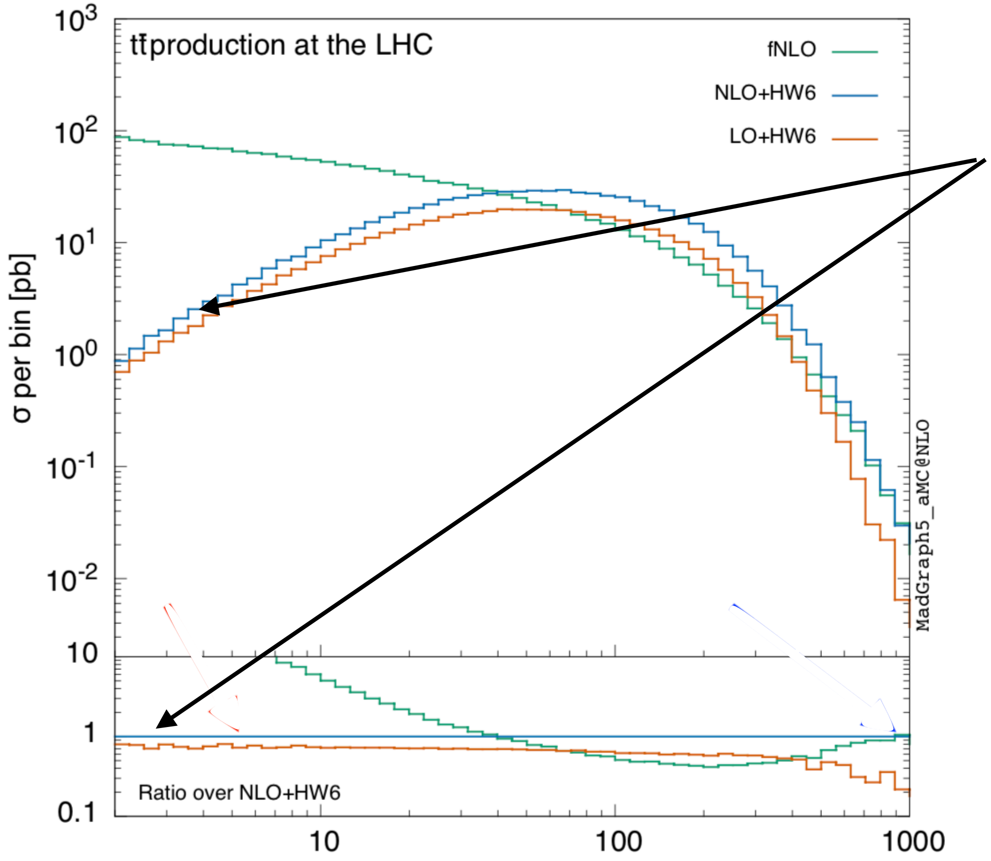
$$+ \left[d\Phi_{m+1}\left(R - MC\right)\right] I_{\mathrm{MC}}^{(m+1)}(O)$$



Matching

$$\frac{d\sigma_{\rm NLOwPS}}{dO} = \left[d\Phi_m \left(B + \int_{\rm loop} V + \int d\Phi_1 MC\right)\right] I_{\rm MC}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\rm MC}^{(m+1)}(O)$$



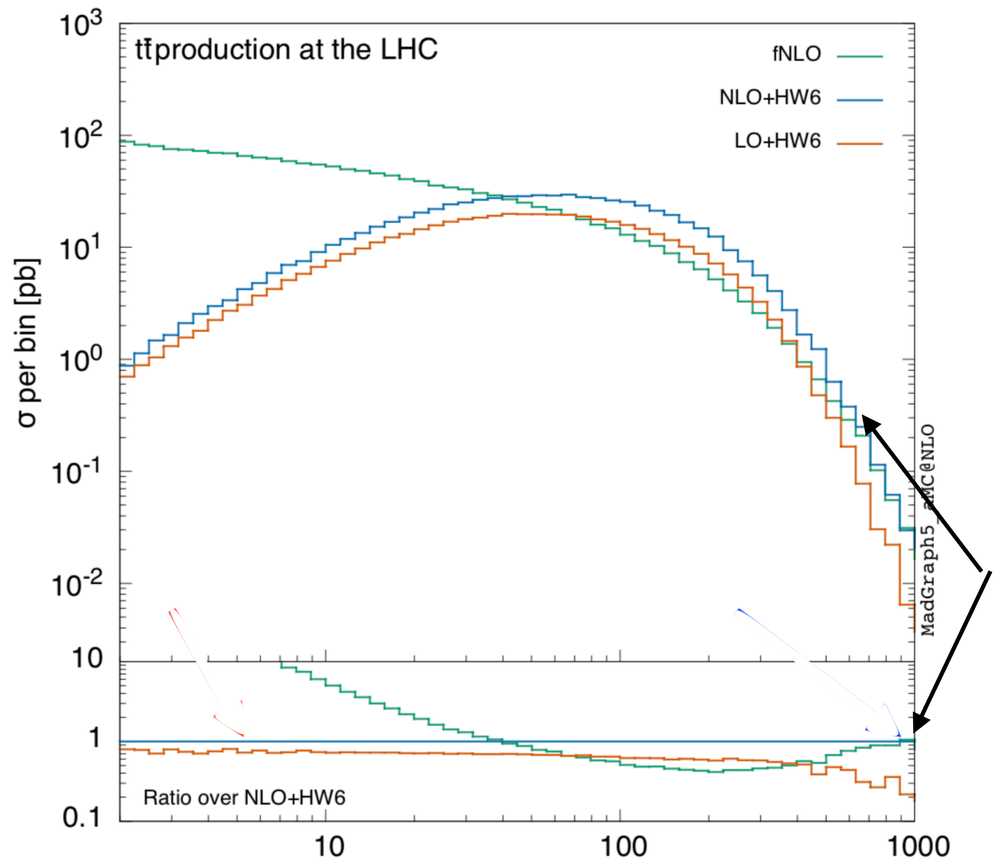
Collinear

$$R \simeq MC \quad \Rightarrow d\sigma_{\rm MC@NLO} \sim I_{\rm MC}^{(m)}(O) dO$$

Matching

$$\frac{d\sigma_{\text{NLOwPS}}}{dO} = \left[d\Phi_m\left(B + \int_{\text{loop}} V + \int d\Phi_1 MC\right)\right] I_{\text{MC}}^{(m)}(O)$$

$$+ \left[d\Phi_{m+1}(R - MC)\right] I_{\text{MC}}^{(m+1)}(O)$$



Collinear

$$R \simeq MC \quad \Rightarrow d\sigma_{\text{MC@NLO}} \sim I_{\text{MC}}^{(m)}(O)dO$$

Hard Region

$$MC \simeq 0$$

$$I_{\text{MC}}^{(m)}(O) \simeq 0 \quad I_{\text{MC}}^{(m+1)}(O) \simeq 1$$

$$\Rightarrow d\sigma_{\text{MC@NLO}} \sim d\Phi_{m+1} R$$

MC@NLO properties

- Good features of including the MC counter terms

1. **Double counting avoided:** The rate expanded at NLO coincides with the total NLO cross section
2. **Smooth matching:** MC@NLO coincides (in shape) with the parton shower in the soft/collinear region, while it agrees with the NLO in the hard region
3. **Un-weighting:** weights associated to different multiplicities are separately finite. The *MC* term has the same infrared behavior as the real emission (there is a subtlety for the soft divergence)

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^{n}_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n \, I_{MC}^{n}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I_{MC}^{n+1}(O)$$

- Because MC counter-term

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)
  - Has the same collinear singularities as the real/virtual

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)
  - Has the same collinear singularities as the real/virtual
- Both term are finite over the phase-space

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)
  - Has the same collinear singularities as the real/virtual
- Both term are finite over the phase-space
- But R-MC can be negative

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n\, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)
  - Has the same collinear singularities as the real/virtual
- Both term are finite over the phase-space
- But R-MC can be negative
- So we can unweight events

Unweighting

$$\frac{d\sigma_{MC@NLO}}{dO} = \left(\mathcal{B} + \mathcal{V} + \int d\Phi_1 MC\right) d\Phi_n \, I^n_{MC}(O) + (\mathcal{R} - MC)\, d\Phi_{n+1} I^{n+1}_{MC}(O)$$

- Because MC counter-term
  - Has the same kinematic of the real (no re-shuffling)
  - Has the same collinear singularities as the real/virtual
- Both term are finite over the phase-space
- But R-MC can be negative
- So we can unweight events
  - But we have negative events

MC@NLO properties

- Good features of including the MC counter terms

1. **Double counting avoided**: The rate expanded at NLO coincides with the total NLO cross section
2. **Smooth matching**: MC@NLO coincides (in shape) with the parton shower in the soft/collinear region, while it agrees with the NLO in the hard region
3. **Un-weighting:** weights associated to different multiplicities are separately finite. The *MC* term has the same infrared behaviour as the real emission (there is a subtlety for the soft divergence)

MC@NLO properties

- Good features of including the MC counter terms

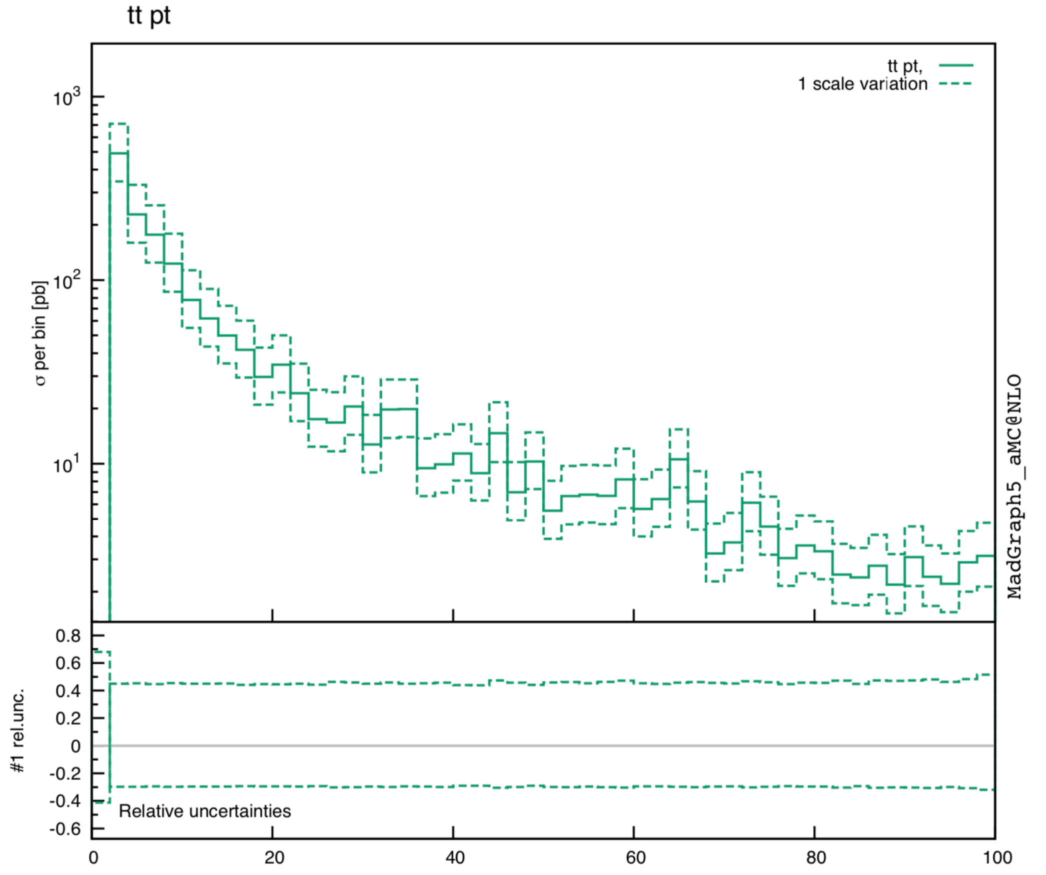
1. **Double counting avoided:**
2. **Smooth matching**
3. : **Un-weighting:**

- Weak points / limitations

1. **Soft limit can be problematic**
2. **Negative events**
3. **Need dedicated implementation of the counter-term**

To Remember (1/2)

- Not all observables are NLO accurate in a NLO computation
- Loop computation
  - ➡ We know a basis of loop (not existing for 2-loop)
  - ➡ Matrix to inverse
    - ☐ Instability



To Remember (2/2)

- fNLO computation done with counter-events
  - ➡ No event generation
  - ➡ bin miss-match
- NLO+PS generation: event generation
  - ➡ Events Physical only after the Parton-Shower.
  - ➡ The Events should be generated for a given shower (in MC@NLO)
  - ➡ Negative events