Three-loop Feynman integrals for leading-colour diboson production at hadron colliders

Mattia Pozzoli

Based on JHEP 02 (2025) 088 and JHEP 05 (2026) 226 with D. Canko

Theory and Phenomenology of Fundamental Interactions
UNIVERSITY AND INFN · BOLOGNA

Istituto Nazionale di Fisica Nucleare
Sezione di Bologna

3rd INPP Demokritos-APCTP Workshop, Athens, 2/7/2026

Motivation

- **Percent-level experimental precision** for many collider observables $\Longrightarrow$ precision of **theoretical predictions** is a **limiting factor**
- Diboson production: **N3LO QCD** needed
- Interesting studies on the **mathematical structure** of the **Feynman integrals**

From cross-sections to Feynman integrals

Cross sections for $pp \to f$:

$$\mathrm{d}\sigma_{pp\to f} = \sum_{i,j=q,\bar{q},g} \int\int \mathrm{d}x_1 \mathrm{d}x_2\, \mathscr{F}_i(x_1,\mu^2)\mathscr{F}_j(x_2,\mu^2)\, \mathrm{d}\hat{\sigma}_{ij\to f}(\vec{s},\mu^2)$$

Partonic cross section:

$$\mathrm{d}\hat{\sigma}_{ij\to f} \backsim \int \mathrm{d}\Phi\, |\mathscr{A}|^2$$

Amplitude:

Rational coefficients

$$\mathscr{A} \backsim \sum_i F_i(\vec{s};\varepsilon)\, G_i(\vec{s};\varepsilon)$$

Feynman Integrals

Setup and kinematics

$$p_1 + p_2 + p_3 + p_4 = 0$$

$$p_1^2 = p_2^2 = 0,$$

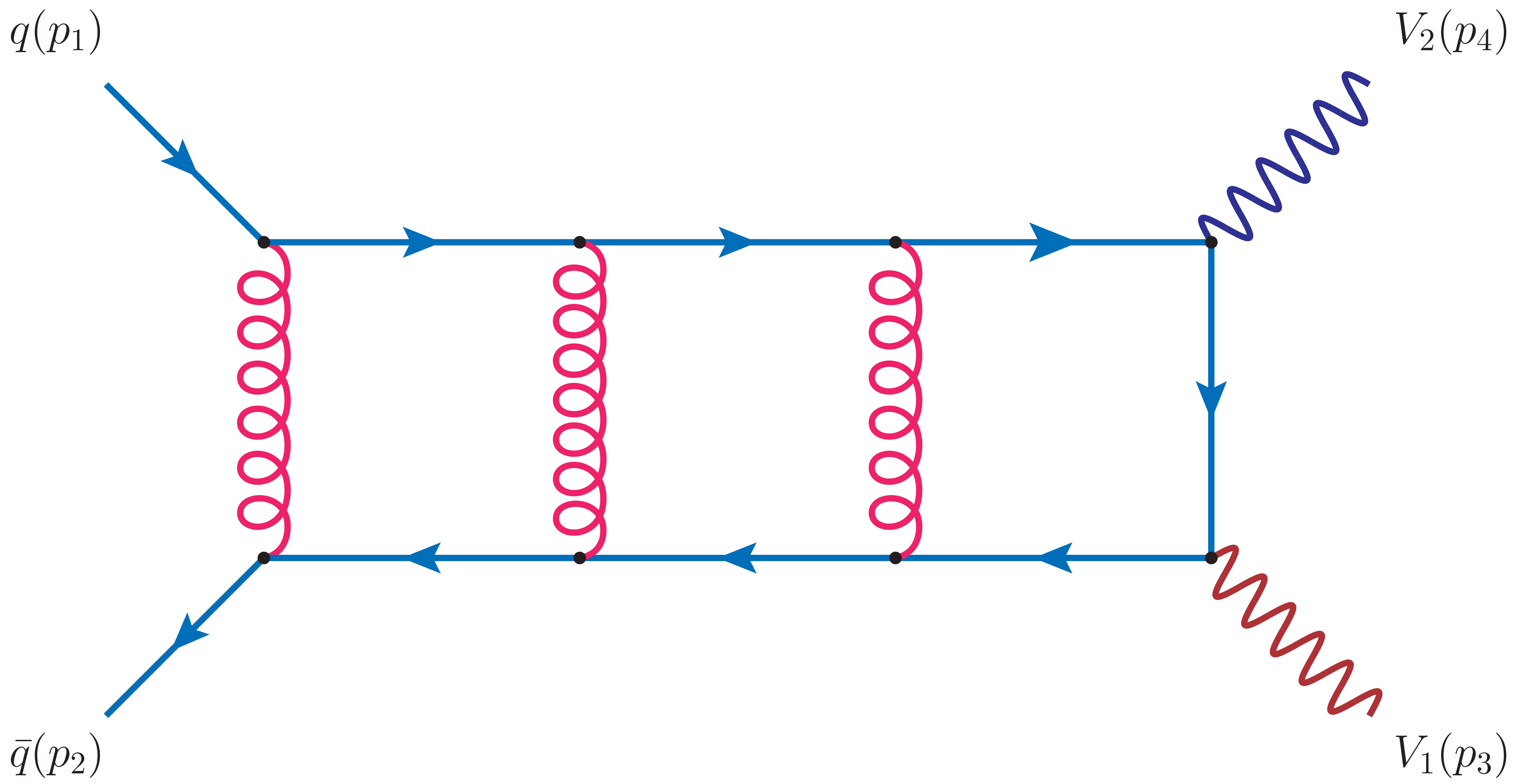
$$p_3^2 = m_3^2,\ p_4^2 = m_4^2$$

4 invariants:

$$\vec{s} := \left\{s_{12}, s_{23}, m_3^2, m_4^2\right\}, \quad s_{ij} = (p_i + p_j)^2$$

Dimensional regularisation:

$$d = 4 - 2\varepsilon$$



Outline

- Families of Feynman integrals
- Calculation of the Feynman integrals
- Solving the differential equations

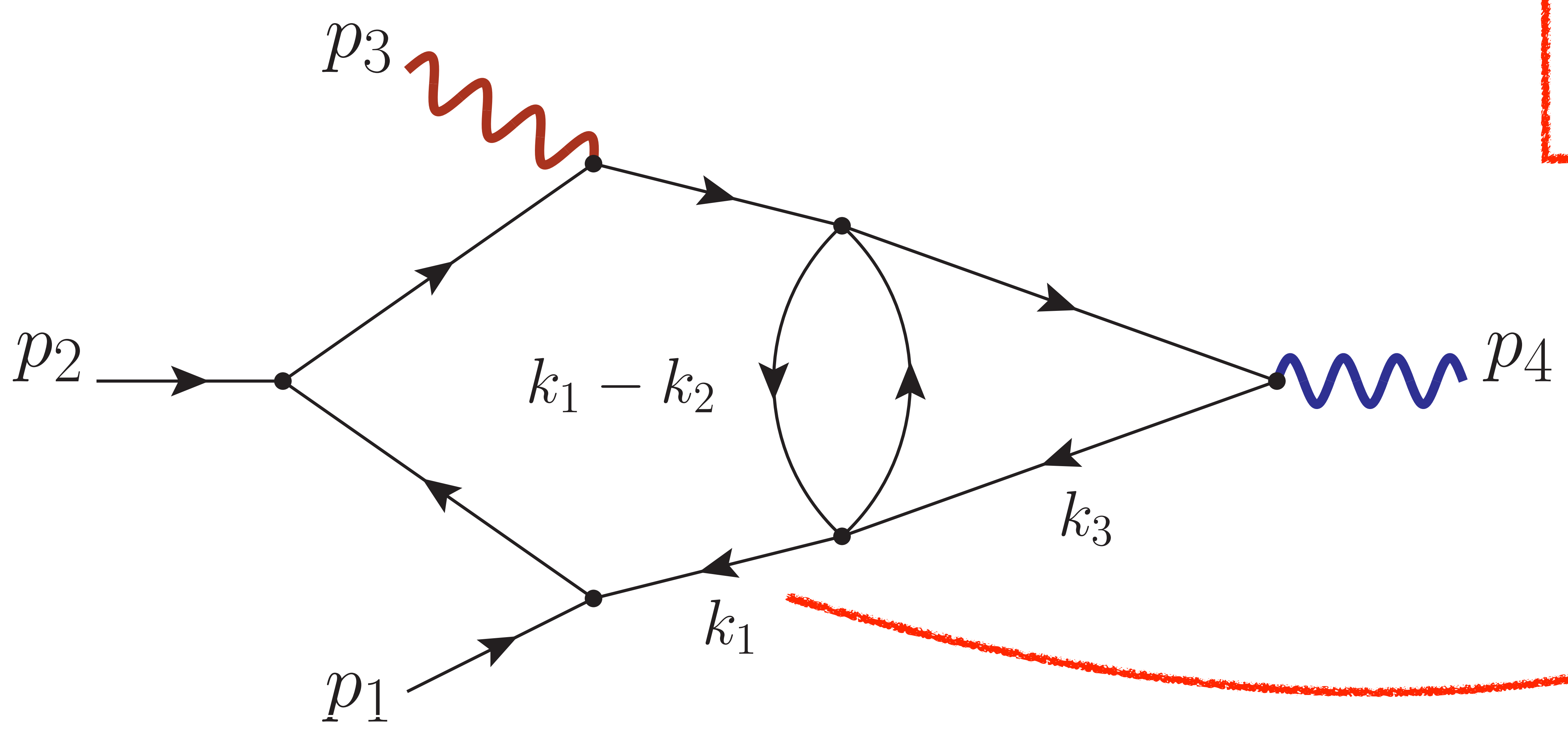
Families of Feynman integrals

Integral Families

$$G_{a_1,\ldots,a_{15}} = \int \mathrm{d}^d k_1 \mathrm{d}^d k_2 \mathrm{d}^d k_3 \frac{1}{\boxed{D_1^{a_1} \cdots D_{15}^{a_{15}}}}, \qquad (a_1, \ldots, a_{15}) \in \mathbb{Z}^{15}$$

Inverse propagators of Feynman diagrams

$$D_j = l_j^2 - m_j^2$$



e.g. $D_1 = k_1^2$

Integral Families

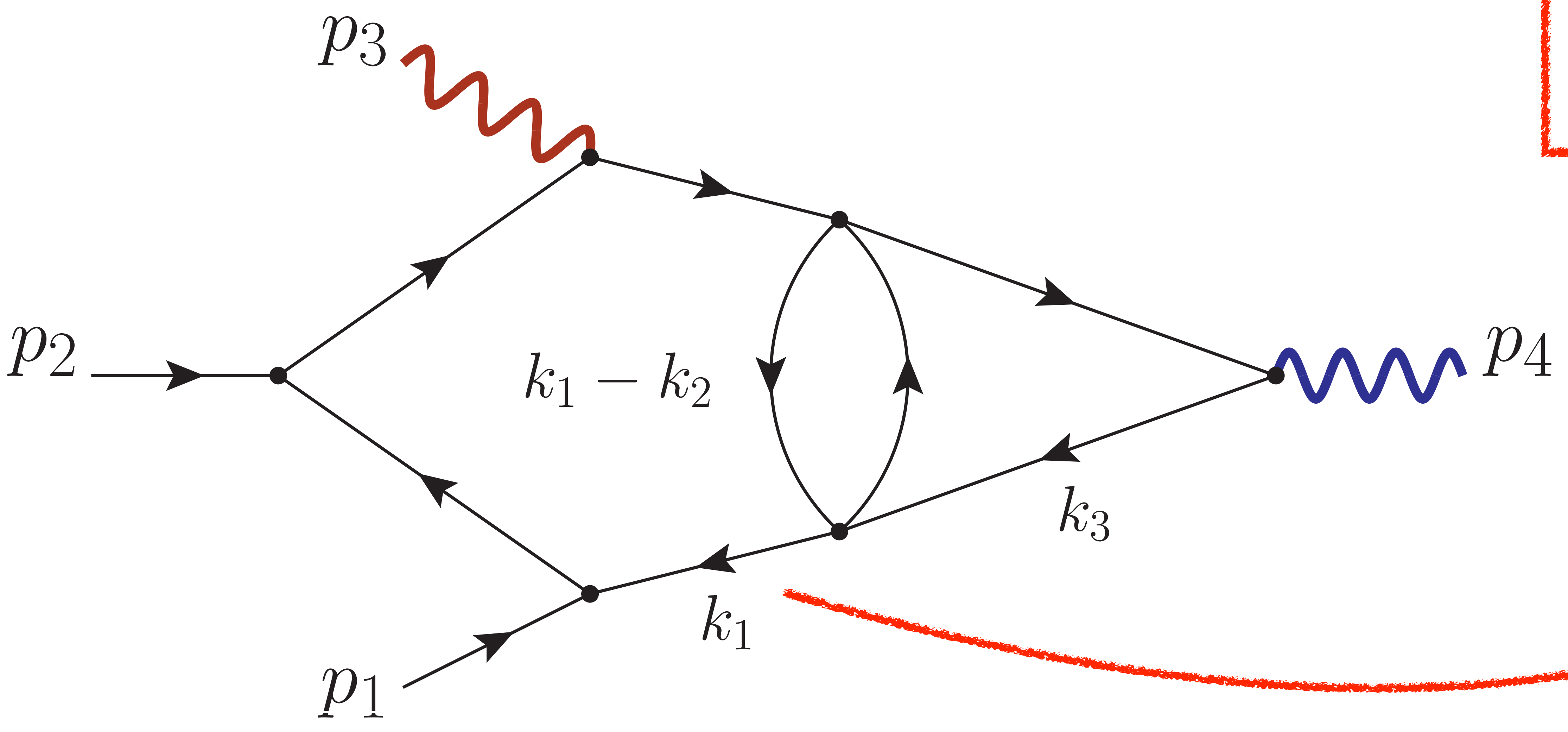
Sector: same set of $a_i > 0$

Irreducible scalar products (ISPs): $a_i \leq 0$ for all integrals

$$G_{a_1,\ldots,a_{15}} = \int \mathrm{d}^d k_1 \mathrm{d}^d k_2 \mathrm{d}^d k_3 \frac{1}{\boxed{D_1^{a_1} \cdots D_{15}^{a_{15}}}}, \qquad (a_1, \ldots, a_{15}) \in \mathbb{Z}^{15}$$

Inverse propagators of Feynman diagrams

$$D_j = l_j^2 - m_j^2$$



e.g. $D_1 = k_1^2$

IBPs and reduction to master integrals

Feynman integrals satisfy **linear relations**: integration by part identities (IBPs) [Tkachov '81], [Chetyrkin, Tkachov '81]

$$0 = \int \mathrm{d}^d k_1 \mathrm{d}^d k_2 \mathrm{d}^d k_3 \frac{\partial}{\partial k_l^\mu} \frac{v^\mu}{D_1^{a_1} \cdots D_{15}^{a_{15}}}, \quad v^\mu \in \{k_j^\mu, p_j^\mu\}$$

Master integrals (MIs) $\vec{I}(\vec{x})$

Reduction to **master integrals**

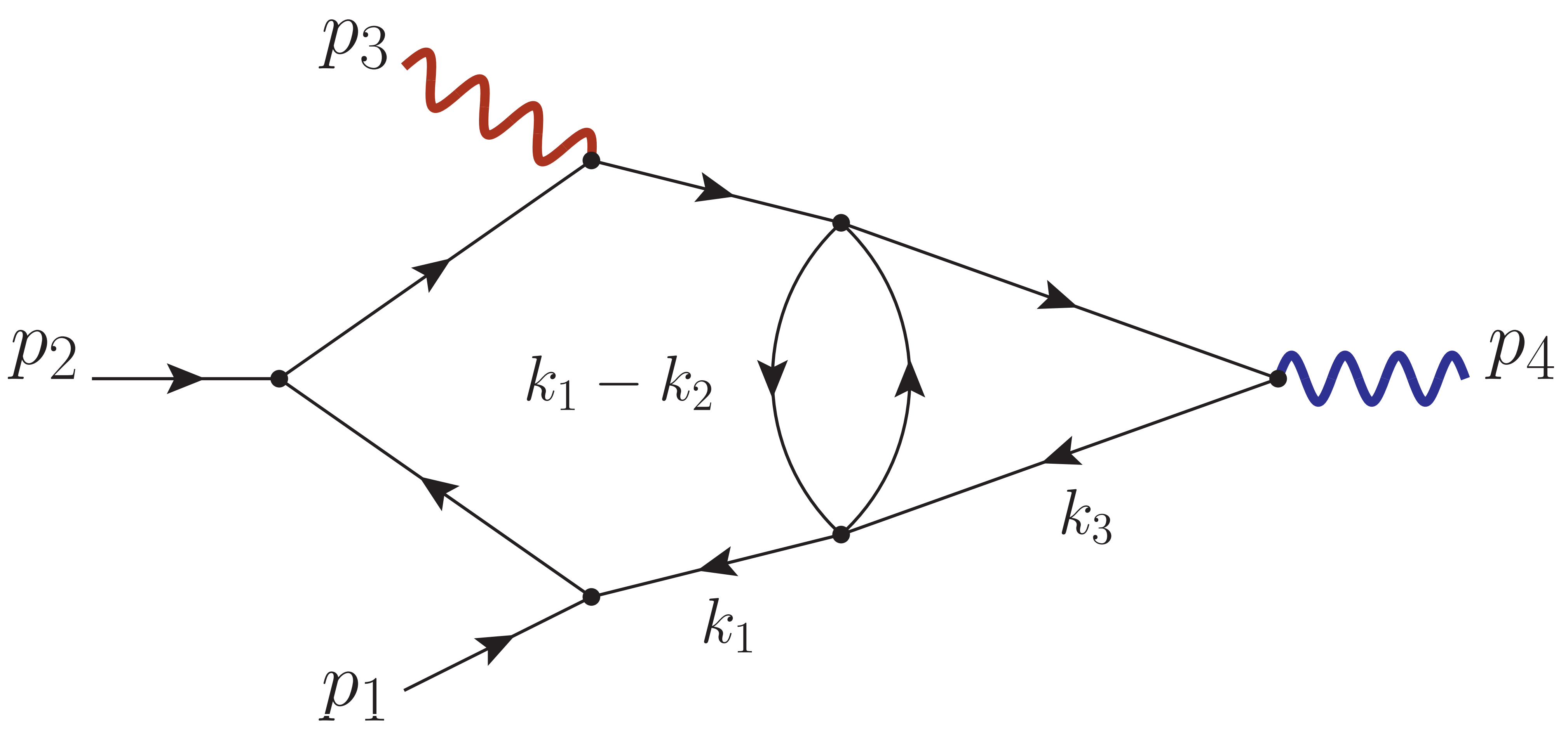
$$\sum_{\vec{a}_k} c_{\vec{a}_k}(\vec{x};\varepsilon) G_{\vec{a}_k}(\vec{x};\varepsilon) = 0 \Longrightarrow G_{\vec{a}}(\vec{x};\varepsilon) = \sum_j c_{\vec{a},j}(\vec{x};\varepsilon) \boxed{I_j(\vec{x};\varepsilon)}$$

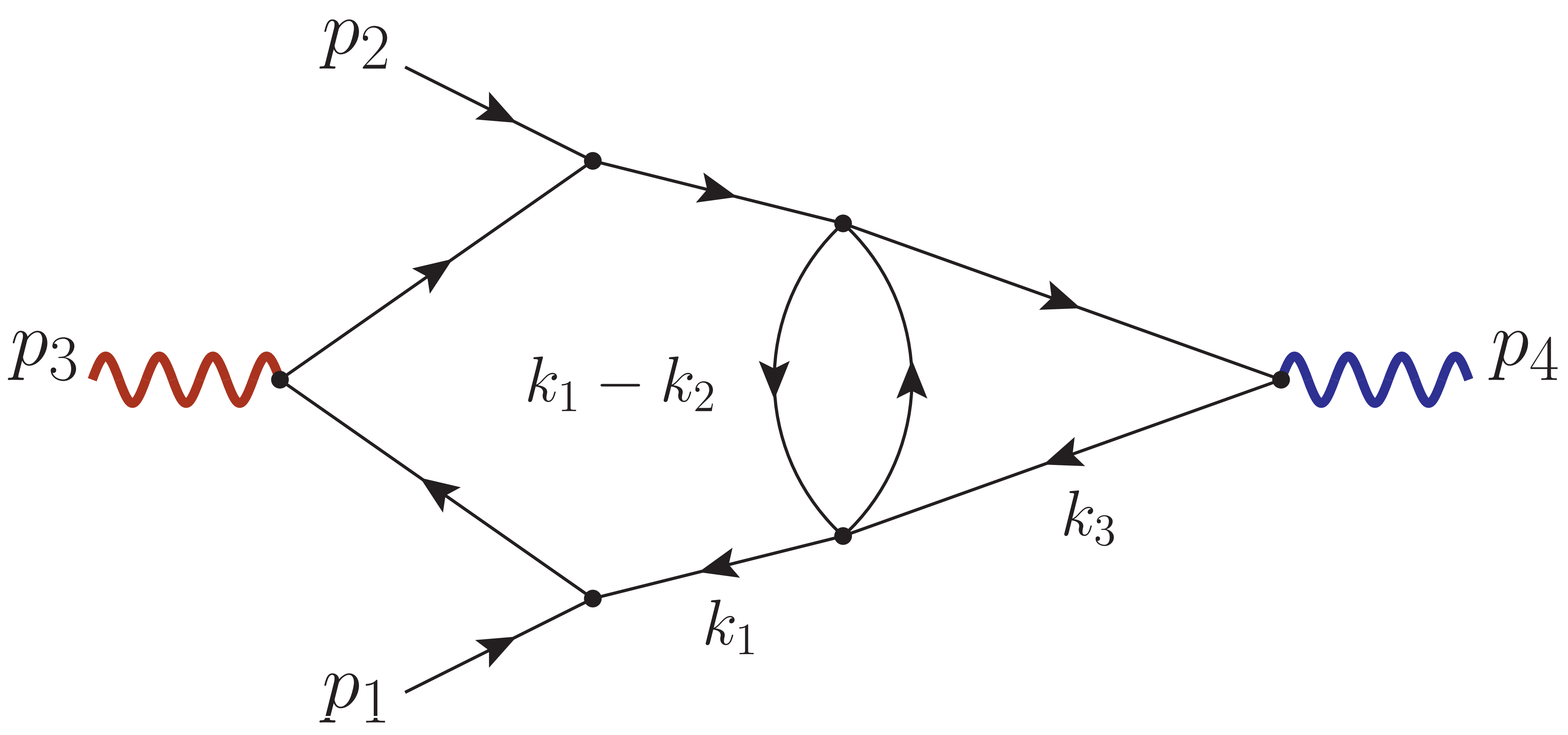
Laporta algorithm: we generate IBPs for some choices of exponents [Laporta 2000]

Finite fields techniques [von Manteuffel, Schabinger 2014; Peraro 2016] to tackle algebraic complexity

`FFIntRed` and `FiniteFlow` [Peraro 2019] to generate and solve IBPs

The reducible ladder-box families

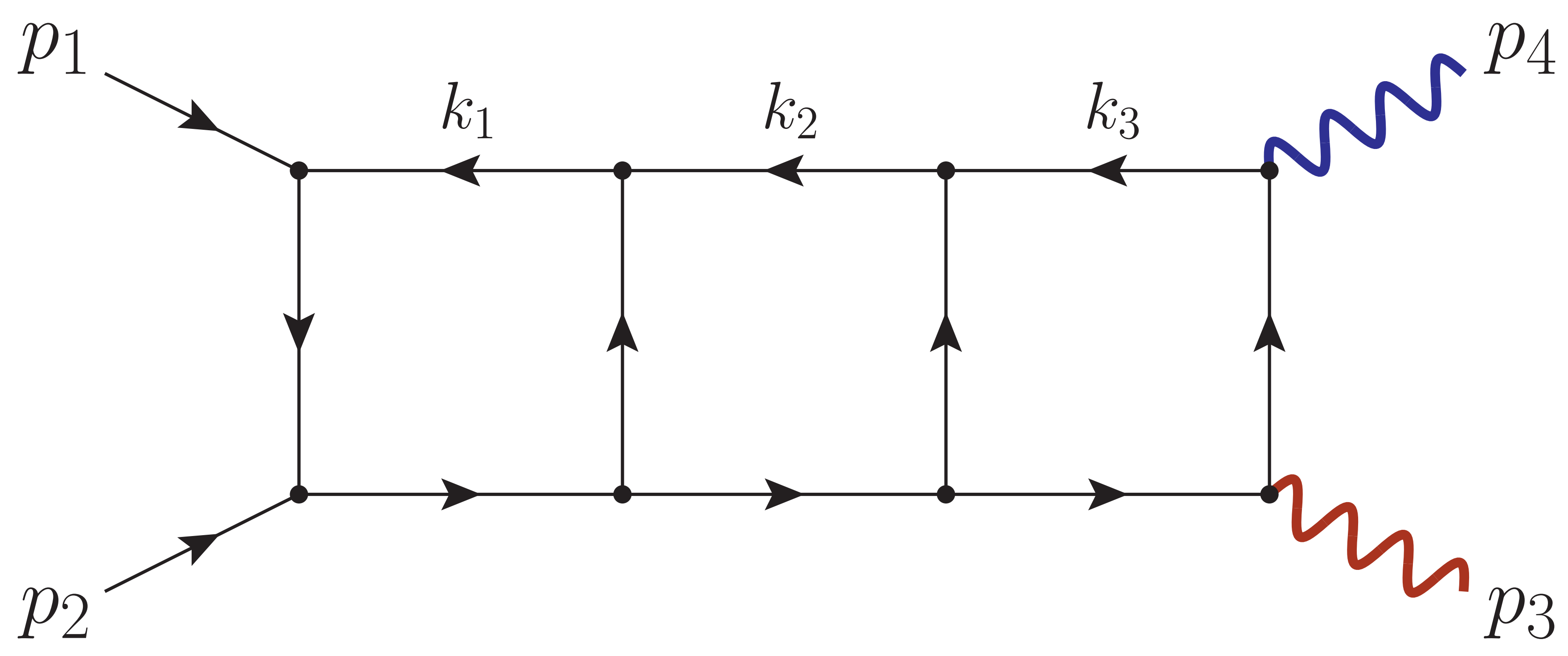


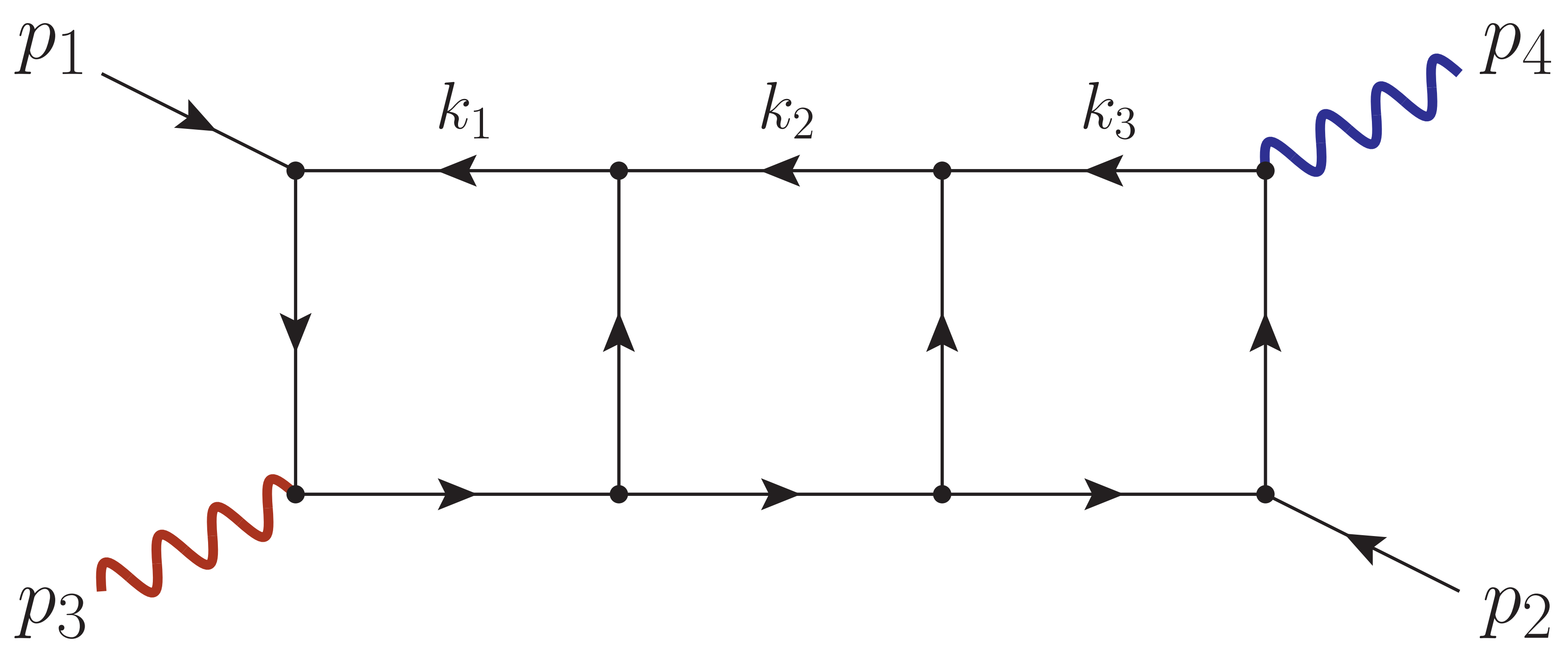


$\boxed{\mathrm{RL}_1\text{: 27 MIs (24 for equal masses)}}$

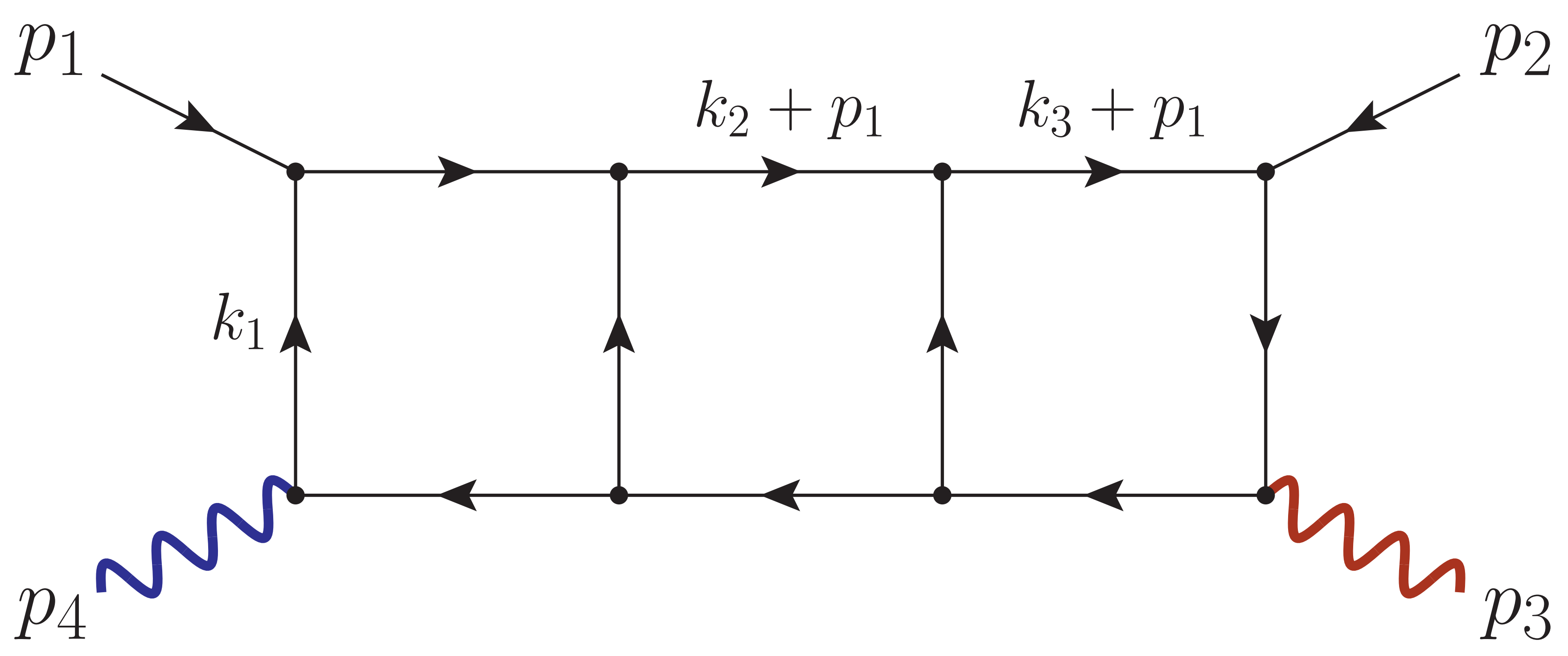
$\boxed{\mathrm{RL}_2\text{: 25 MIs (22)}}$

The ladder-box families

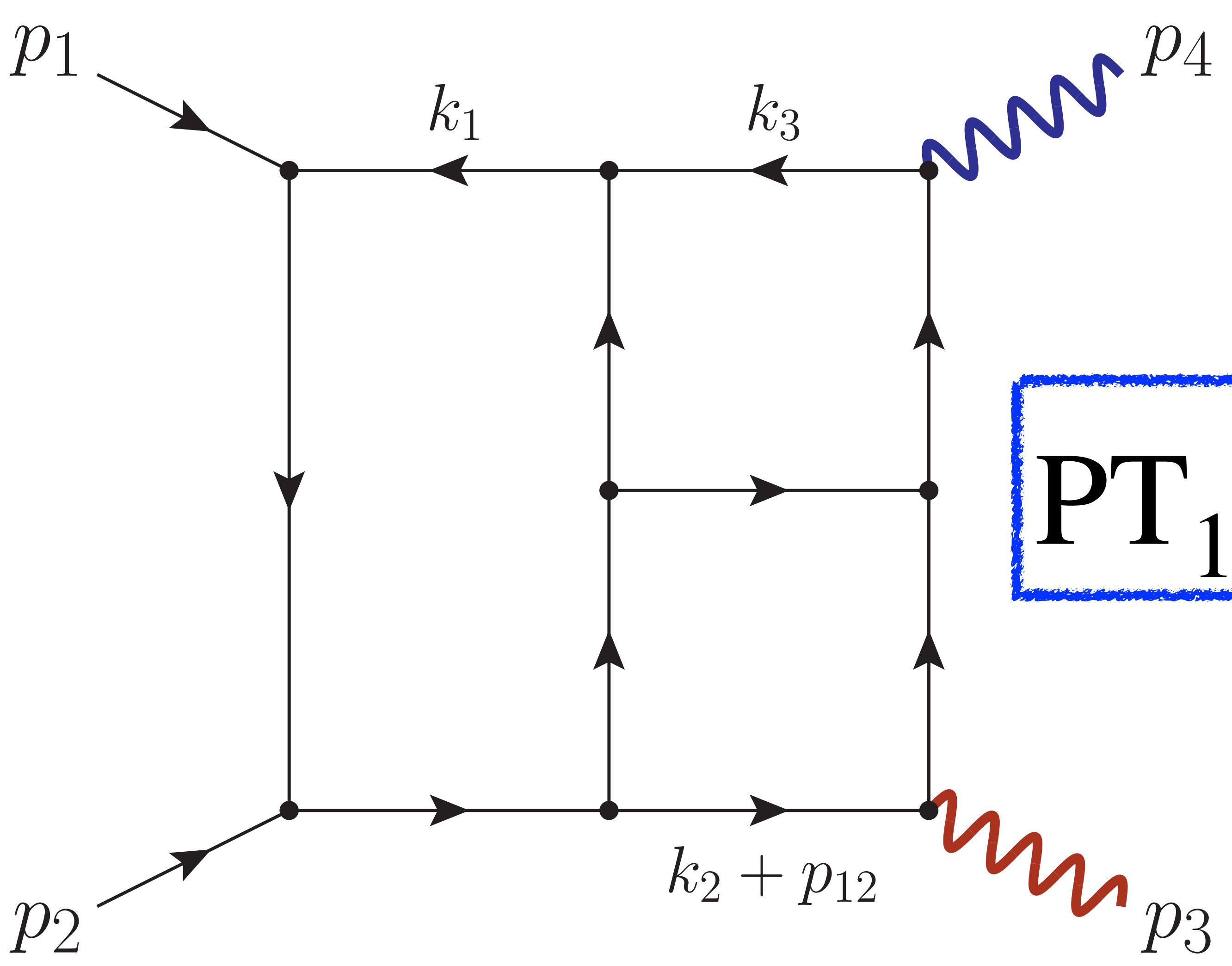




$\boxed{\mathrm{PL}_1\text{: 150 MIs (94)}}$ $\boxed{\mathrm{PL}_2\text{: 143 MIs (84)}}$ $\boxed{\mathrm{PL}_3\text{: 142 MIs (81)}}$

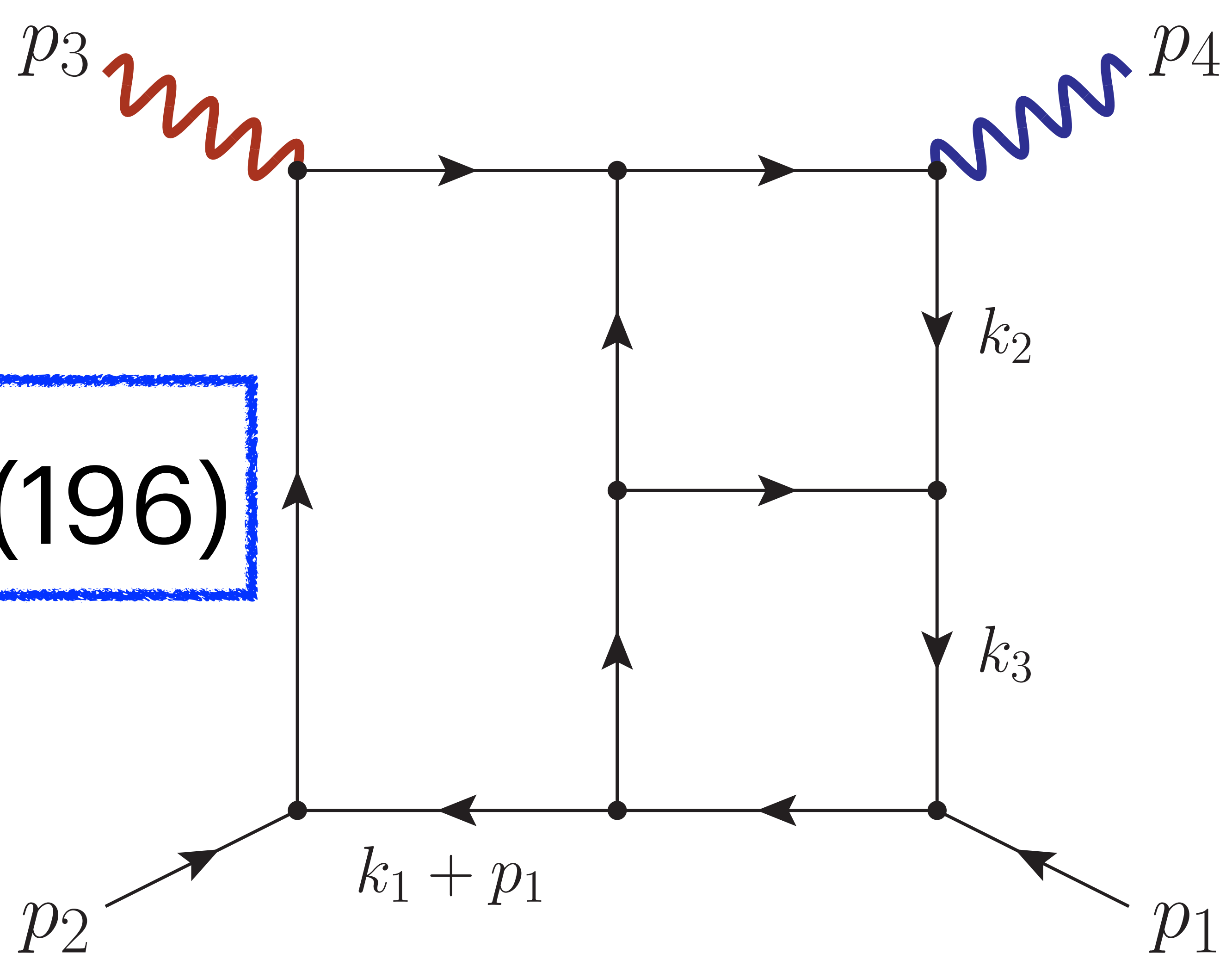


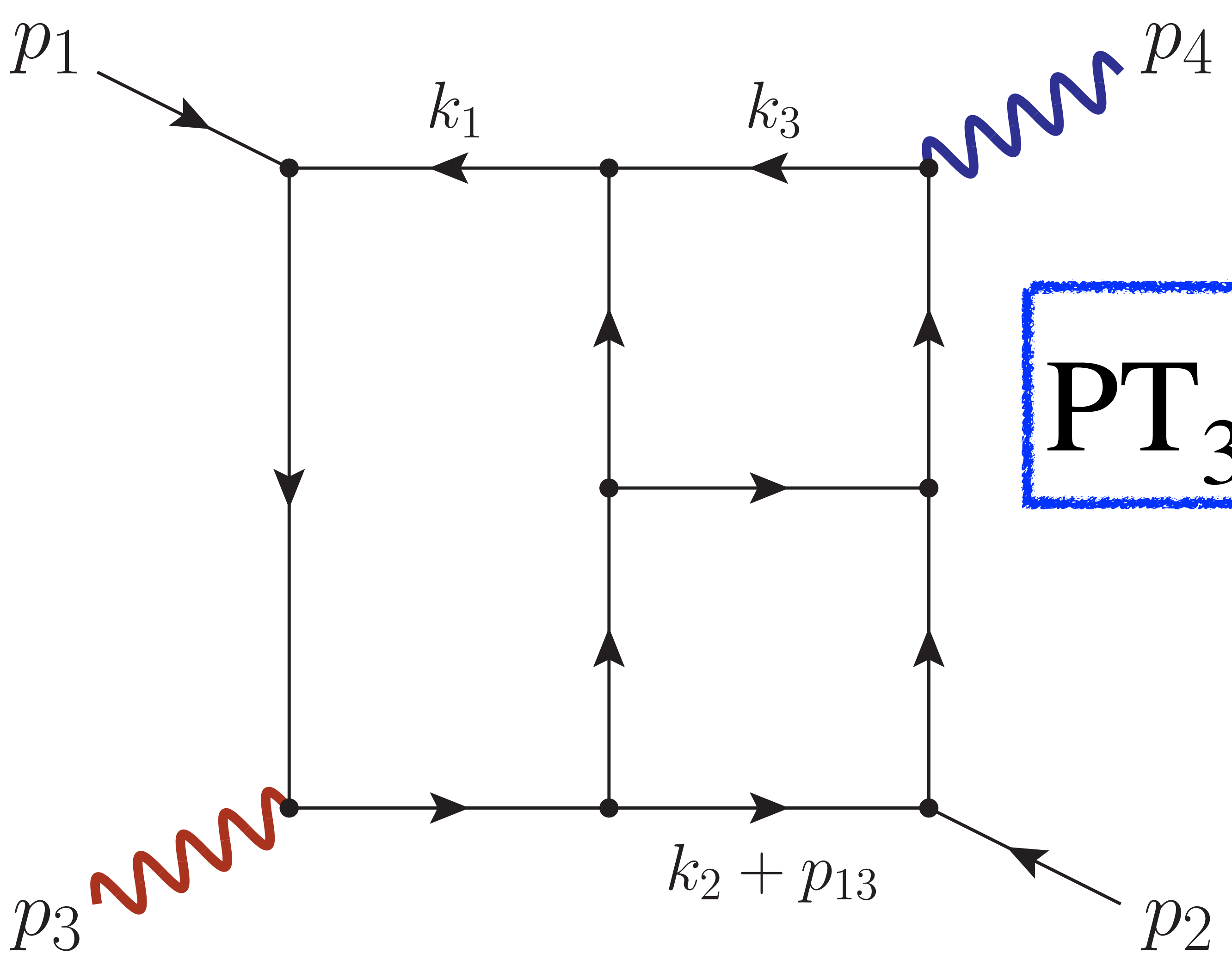
The tennis-court families



PT_1: 344 MIs (188)

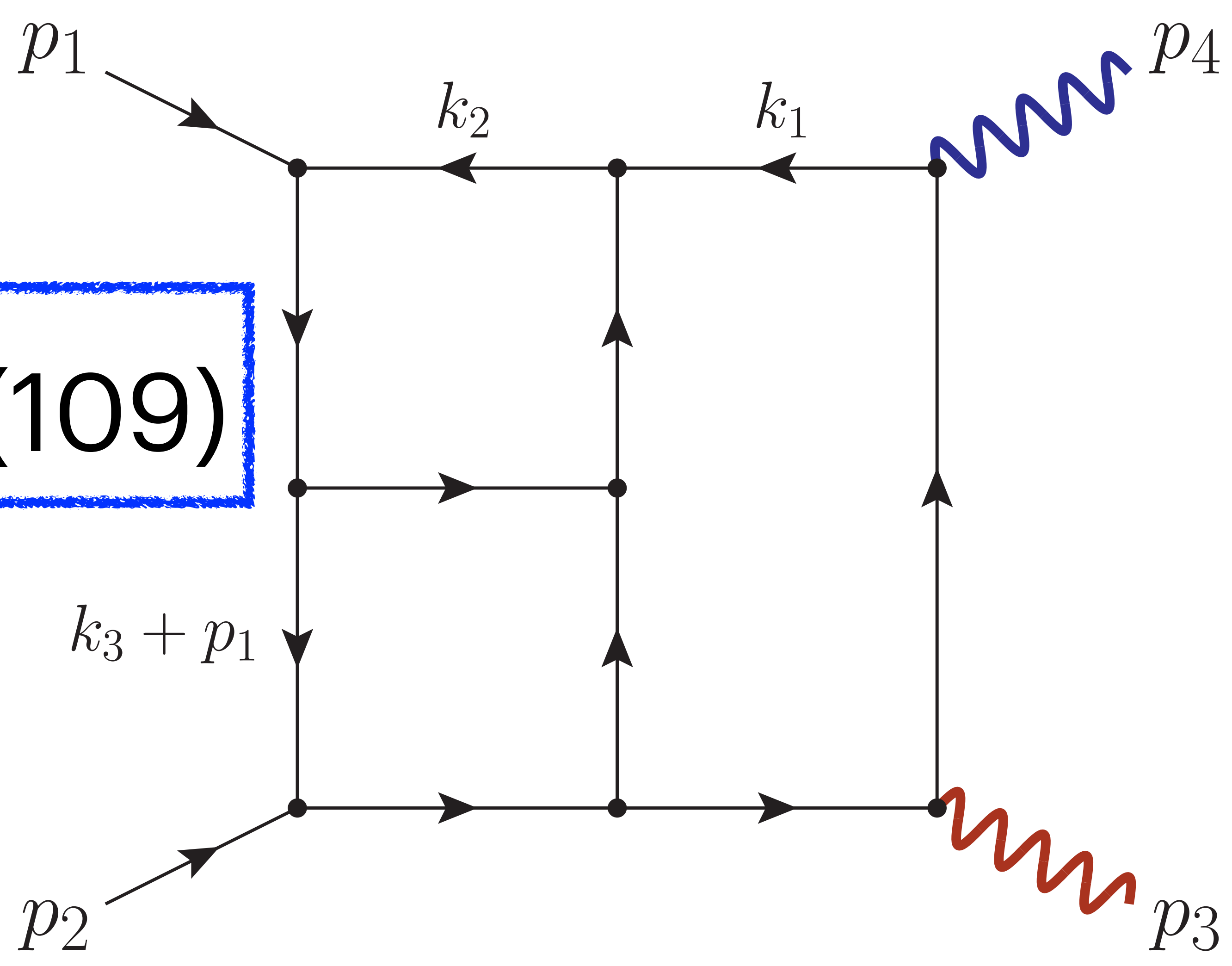
PT_2: 252 MIs (196)





PT_3: 240 MIs (181)

PT_4: 189 MIs (109)



Calculation of the Feynman integrals

Method of differential equations

[Kotikov '91; Bern, Dixon, Kosower '94; Gehrmann, Remiddi 2000]

Using IBPs we can construct **linear differential equations** (DEs) for the MIs

$$\forall\ \xi \in \vec{x}: \qquad \partial_\xi I_i(\vec{x};\varepsilon) = \sum_{\vec{a}} c_{i,\vec{a}}(\vec{x};\varepsilon) G_{\vec{a}}(\vec{x};\varepsilon)$$

IBP reduction

$$\Longrightarrow \partial_\xi \vec{I}(\vec{x};\varepsilon) = B_\xi(\vec{x};\varepsilon)\cdot \vec{I}(\vec{x};\varepsilon)$$

For a general basis the **matrices** B_ξ are **rational functions** that

- mix ε and the kinematics
- have spurious poles

What is a good choice of MIs?

A good choice of basis can greatly simplify the DEs

[Henn 2014]: DEs in **canonical form** (no general algorithm)

$$\mathrm{d}\vec{I}(\vec{x};\varepsilon) = \varepsilon \, \boxed{\mathrm{d}\tilde{A}(\vec{x})} \, \vec{I}(\vec{x};\varepsilon)$$

one-forms with **at most simple poles**

- ε-dependence factorises: solution at each order depends only on previous order
- Formal solution in terms of **Chen iterated integrals**

See D. Canko's talk for a case beyond the *dlog*-form

In the best understood cases the one-forms are logarithmic

$$\mathrm{d}\tilde{A}(\vec{x}) = \sum_i a^{(i)} \, \mathrm{d}\log \boxed{W_i(\vec{x})}$$

All our families are *dlog*!

Letters

$$\mathrm{d}\log(z+c) = \frac{\mathrm{d}z}{z+c}$$

Canonical form and iterated integrals

$$\vec{I}(\vec{x};\varepsilon) = \sum_{w=0}^{M} \varepsilon^w \ \vec{I}^{(w)}(\vec{x}) + \mathcal{O}(\varepsilon^{M+1})$$

The factorisation of ε allows us to solve the DEs iteratively:

$$\mathrm{d}\vec{I}(\vec{x};\varepsilon) = \varepsilon \ \mathrm{d}\tilde{A}(\vec{x}) \cdot \vec{I}(\vec{x};\varepsilon) \Longrightarrow \mathrm{d}\vec{I}^{(w)}(\vec{x}) = \mathrm{d}\tilde{A}(\vec{x}) \cdot \vec{I}^{(w-1)}(\vec{x})$$

Solution in terms of iterated integrals [Chen 1977]

$$\mathrm{I}_\gamma(\omega_1,\ldots,\omega_n) = \int_\gamma \omega_1\ldots\omega_n = \int_{0\leq t_1\leq\ldots\leq t_n} f_1(t_1)\ \mathrm{d}t_1\ldots f_n(t_n)\ \mathrm{d}t_n$$

Canonical form and iterated integrals

$$\boxed{\vec{I}(\vec{x};\varepsilon) = \sum_{w=0}^{M} \varepsilon^w \, \vec{I}^{(w)}(\vec{x}) + \mathcal{O}(\varepsilon^{M+1})}$$

$\boxed{\text{For canonical basis: } M = 6}$

The factorisation of ε allows us to solve the DEs iteratively:

$$\mathrm{d}\vec{I}(\vec{x};\varepsilon) = \varepsilon \ \mathrm{d}\tilde{A}(\vec{x}) \cdot \vec{I}(\vec{x};\varepsilon) \Longrightarrow \mathrm{d}\vec{I}^{(w)}(\vec{x}) = \mathrm{d}\tilde{A}(\vec{x}) \cdot \vec{I}^{(w-1)}(\vec{x})$$

Solution in terms of iterated integrals [Chen 1977]

$$\mathrm{I}_\gamma(\omega_1, \ldots, \omega_n) = \int_\gamma \omega_1 \ldots \omega_n = \int_{0 \le t_1 \le \ldots \le t_n} f_1(t_1) \ \mathrm{d}t_1 \ldots f_n(t_n) \ \mathrm{d}t_n$$

Canonical form and iterated integrals

$$\boxed{\vec{I}(\vec{x};\varepsilon) = \sum_{w=0}^{M} \varepsilon^{w}\ \vec{I}^{(w)}(\vec{x}) + \mathcal{O}(\varepsilon^{M+1})}$$

For canonical basis: $M = 6$

The factorisation of ε allows us to solve the DEs iteratively:

$$\mathrm{d}\vec{I}(\vec{x};\varepsilon) = \varepsilon\ \mathrm{d}\tilde{A}(\vec{x})\cdot\vec{I}(\vec{x};\varepsilon) \Longrightarrow \mathrm{d}\vec{I}^{(w)}(\vec{x}) = \mathrm{d}\tilde{A}(\vec{x})\cdot\vec{I}^{(w-1)}(\vec{x})$$

Solution in terms of iterated integrals [Chen 1977]

$$\mathrm{I}_{\gamma}(\omega_1,\ldots,\omega_n) = \int_{\gamma} \boxed{\omega_1\ldots\omega_n} = \int_{0\le t_1\le\ldots\le t_n} \boxed{f_1(t_1)}\ \mathrm{d}t_1\ldots f_n(t_n)\ \mathrm{d}t_n$$

One-forms over space X

Pull-backs of the one-forms by the path γ

For canonical DEs in *dlog*-form: $\omega_i = \mathrm{d}\log(W_i(x))$

Integrand analysis

[Arkani-Hamed, Bourjaily, Cachazo, Trnka 2012], [Henn, Mistlberger, Smirnov, Wasser, 2020]

Conjecture: integrals with a loop-integrand with **at most simple poles** and a **constant leading singularity** satisfy canonical DEs

$$I^{\varepsilon=0} \propto \boxed{R(\vec{x})} \int \boxed{\mathrm{d}\log(\ldots) \wedge \mathrm{d}\log(\ldots)}$$

$$\boxed{\mathrm{d}\log(z+c) = \frac{\mathrm{d}z}{z+c}}$$

Leading singularity (LS)

***dlog*-form**

$$\implies \frac{1}{R(\vec{x})} I$$ is a good MI

Integrand analysis

[Arkani-Hamed, Bourjaily, Cachazo, Trnka 2012], [Henn, Mistlberger, Smirnov, Wasser, 2020]

Conjecture: integrals with a loop-integrand with **at most simple poles** and a **constant leading singularity** satisfy canonical DEs

$$I^{\varepsilon=0} \propto \boxed{R(\vec{x})} \int \boxed{\mathrm{d}\log(\ldots) \wedge \mathrm{d}\log(\ldots)}$$

$$\boxed{\mathrm{d}\log(z+c) = \frac{\mathrm{d}z}{z+c}}$$

Leading singularity (LS)

***dlog*-form**

$\Longrightarrow \frac{1}{R(\vec{x})} I$ is a good MI

Commonly: **rational functions** and **simple square roots**

An alternative approach

[Argeri et al. 2014]
[Gehrmann et al. 2014]

Over 800 MIs to compute $\Longrightarrow$ **integrand analysis** too **time-consuming**

Pre-canonical basis such that

$$B_\xi(\vec{x};\varepsilon) = \varepsilon\ B_\xi^{(1)}(\vec{x}) + \boxed{B_\xi^{(0)}(\vec{x})}$$

$$\boxed{\begin{pmatrix} * & * & * \\ 0 & * & * \\ 0 & 0 & * \end{pmatrix}}$$

$B_\xi^{(0)}$ can be removed through **algebraic transformations**

$$\vec{I}(\vec{x};\varepsilon) \longrightarrow T(\vec{x};\varepsilon)\cdot\vec{I}(\vec{x};\varepsilon)$$

Such that

$$B_\xi(\vec{x};\varepsilon) \longrightarrow \Big(T(\vec{x};\varepsilon)\cdot B_\xi(\vec{x};\varepsilon) + \partial_\xi T(\vec{x};\varepsilon)\Big)\cdot T^{-1}(\vec{x};\varepsilon)$$

Removing the diagonal entries

Removing the diagonal entries

Toy example:

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right) f(x;\varepsilon)$$

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right) f(x;\varepsilon)$$

Find **logarithmic primitive**

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right) f(x;\varepsilon)$$

Find **logarithmic primitive**

$$b^{(0)}(x) = -\,\partial_x \log(c(x)) = -\,\frac{\partial_x c(x)}{c(x)}$$

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right) f(x;\varepsilon)$$

Find **logarithmic primitive**

$$b^{(0)}(x) = -\,\partial_x \log(c(x)) = -\,\frac{\partial_x c(x)}{c(x)}$$

Normalise the integral as $g(x,\varepsilon) = c(x)\ f(x;\varepsilon)$

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right)\ f(x;\varepsilon)$$

Find **logarithmic primitive**

$$b^{(0)}(x) = -\,\partial_x \log(c(x)) = -\,\frac{\partial_x c(x)}{c(x)}$$

Normalise the integral as $g(x,\varepsilon) = c(x)\ f(x;\varepsilon)$

$$\Longrightarrow \partial_x g(x;\varepsilon) = \left(\partial_x c(x)\right)\ f(x;\varepsilon) + c(x)\left(\varepsilon\ b^{(1)}(x) - \frac{\partial_x c(x)}{c(x)}\right)\ f(x;\varepsilon)$$

$$= \varepsilon\ b^{(1)}(x)\ g(x;\varepsilon)$$

Removing the diagonal entries

Toy example:

$$\partial_x f(x;\varepsilon) = \left(\varepsilon\ b^{(1)}(x) + b^{(0)}(x)\right)\ f(x;\varepsilon)$$

Find **logarithmic primitive**

$$b^{(0)}(x) = -\,\partial_x \log(c(x)) = -\,\frac{\partial_x c(x)}{c(x)}$$

Normalise the integral as $g(x,\varepsilon) = c(x)\ f(x;\varepsilon)$

$$\implies \partial_x g(x;\varepsilon) = \cancel{\left(\partial_x c(x)\right)\ f(x;\varepsilon)} + c(x)\left(\varepsilon\ b^{(1)}(x) - \cancel{\frac{\partial_x c(x)}{c(x)}}\right)\ f(x;\varepsilon)$$

$$= \varepsilon\ b^{(1)}(x)\ g(x;\varepsilon)$$

Removing the off-diagonal entries

Removing the off-diagonal entries

Toy example:

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

$$b(x) = -\,\partial_x c(x)$$

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

$$b(x) = -\partial_x c(x)$$

Then $G(x;\varepsilon) = \begin{pmatrix} f_1 + c\, f_2 \\ f_2 \end{pmatrix}$ satisfies

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

$$b(x) = -\,\partial_x c(x)$$

Then $G(x;\varepsilon) = \begin{pmatrix} f_1 + c\, f_2 \\ f_2 \end{pmatrix}$ satisfies

$$\partial_x G(x;\varepsilon) = \partial_x F(x;\varepsilon) + c \begin{pmatrix} \partial_x f_2 \\ 0 \end{pmatrix} - b(x) \begin{pmatrix} f_2 \\ 0 \end{pmatrix}$$

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

$$b(x) = -\,\partial_x c(x)$$

Then $G(x;\varepsilon) = \begin{pmatrix} f_1 + c\, f_2 \\ f_2 \end{pmatrix}$ satisfies

$$\partial_x G(x;\varepsilon) = \partial_x F(x;\varepsilon) + \boxed{c \begin{pmatrix} \partial_x f_2 \\ 0 \end{pmatrix}} - b(x) \begin{pmatrix} f_2 \\ 0 \end{pmatrix}$$

$\boxed{\propto \varepsilon}$

Removing the off-diagonal entries

Toy example:

$$F(x;\varepsilon) = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix}, \quad \text{with} \quad B_x^{(0)} = \begin{pmatrix} 0 & b(x) \\ 0 & 0 \end{pmatrix}$$

Find **algebraic primitive**

$$b(x) = -\,\partial_x c(x)$$

Removes $B_x^{(0)}$

Then $G(x;\varepsilon) = \begin{pmatrix} f_1 + c\, f_2 \\ f_2 \end{pmatrix}$ satisfies

$$\partial_x G(x;\varepsilon) = \partial_x F(x;\varepsilon) + \boxed{c \begin{pmatrix} \partial_x f_2 \\ 0 \end{pmatrix}} - \boxed{b(x) \begin{pmatrix} f_2 \\ 0 \end{pmatrix}}$$

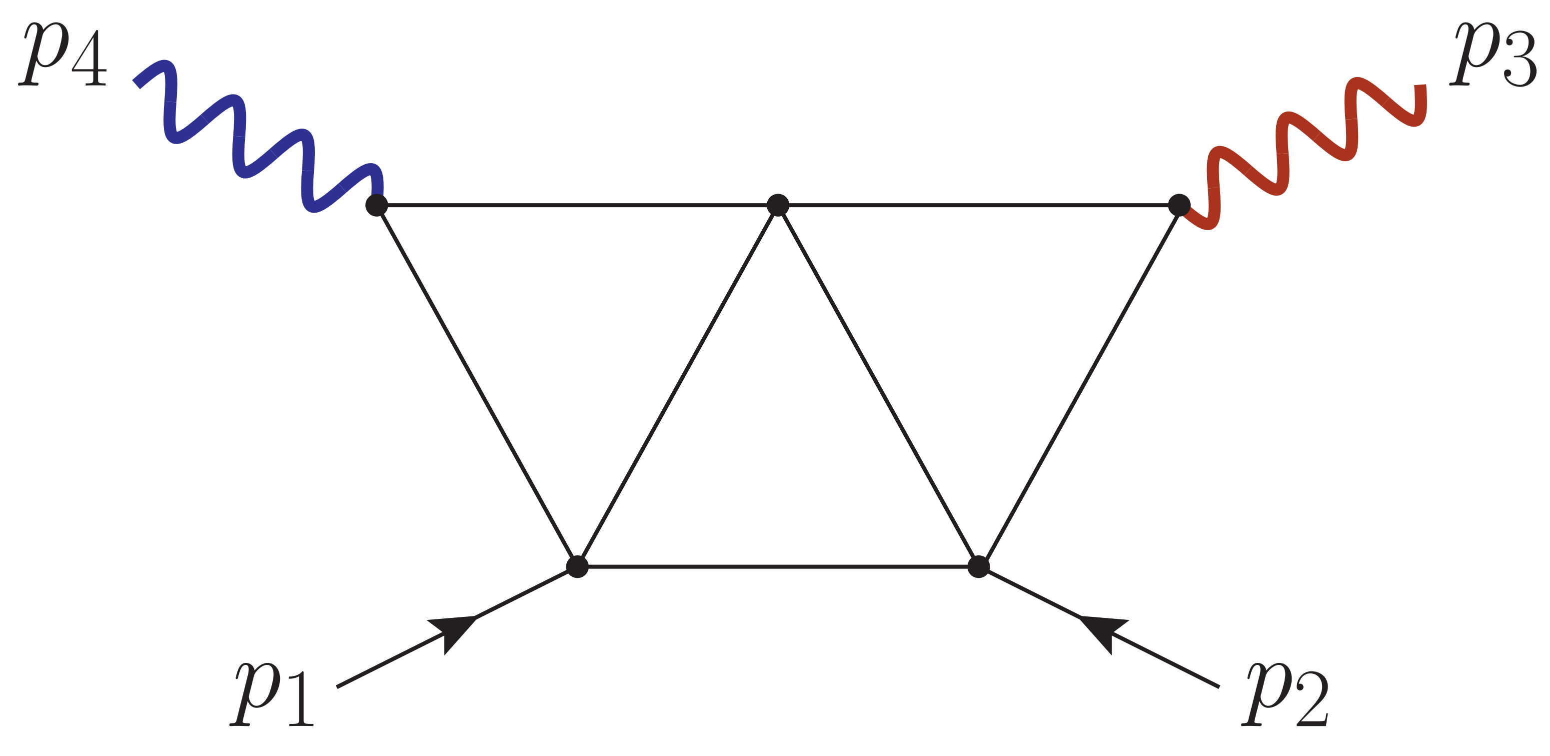
$\boxed{\propto \varepsilon}$

Finding the pre-canonical basis is non-trivial

No general algorithm to find the pre-canonical integrals

- **Public packages** (e.g. `DlogBasis` [Henn, Mistlberger, Smirnov, Wasser 2020])
- **Integrand analysis** in Baikov representation
- Use **patterns** known from **similar calculations**
- Need only ε-dependence $\Rightarrow$use **numerical slices** to test multiple candidates efficiently

Difficulty grows with the number of MIs associated to a given graph



15 MIs in this sector (10 in the equal mass case)

Solving the DEs

Iterated integrals of logarithmic forms

Rational alphabet: Multiple Polylogarithms (**MPLs**)

$$G(a_1, a_2, \ldots, a_n; x) = \int_0^x \frac{\mathrm{d}t}{t - a_1} G(a_2, \ldots, a_n; t)$$

$$G(; x) = 1$$

$$G(\underbrace{0,\ldots,0}_{n}; x) = \frac{1}{n!} \log(x)^n$$

Well-established mathematical technology to deal with them and **public tools** to evaluate them

Analytic solution in terms of GPLs

At **two loops** the canonical DEs involved only **one square root**

$$r_1 = \sqrt{s_{12}^2 + (m_4^2 - m_3^2)^2 - 2s_{12}(m_4^2 + m_3^2)} \left(\sqrt{s_{12}(s_{12} - 4m^2)} \text{ in the massless case} \right)$$

Change of variables

$$\frac{s_{12}}{m_3^2} = (1 + x)(1 + xy), \qquad \frac{s_{23}}{m_3^2} = -xz, \qquad \frac{m_4^2}{m_3^2} = x^2 y$$

$$\Longrightarrow r_1 = m_3^2 x(1 - y)$$

[Canko, MP 2024]: families $\mathrm{RL}_1, \mathrm{RL}_2, \mathrm{PL}_1$ and PT_4 involve only the same square root $\Longrightarrow$ solution in terms of **MPLs**

New square roots appear at three loops!

The families $\mathrm{PT}_1, \mathrm{PT}_2$ and PL_3 involve **four additional square roots**:

$$r_2 = \sqrt{(m_3^2 m_4^2 + s_{13}(s_{12}+s_{13}))^2 - 4m_3^2 m_4^2 s_{13}^2} \xrightarrow{m_3, m_4 \to m} \sqrt{(m^4 + s_{13}(s_{12}+s_{13}))^2 - 4m^4 s_{13}^2}$$

$$r_3 = \sqrt{s_{23}((m_3^2 - m_4^2)^2 s_{23} - 4m_3^2 m_4^2 s_{12})} \longrightarrow 2m^2\sqrt{-s_{12}s_{23}}$$

$$r_4 = \sqrt{(m_4^2 - s_{23})((m_4^2 - s_{12})^2(m_4^2 - s_{23}) - 4m_3^2 m_4^2 s_{12})} \longrightarrow \sqrt{(m^2 - s_{23})((m^2 - s_{12})^2(m^2 - s_{23}) - 4m^4 s_{12})}$$

$$r_5 = \sqrt{(m_3^2 - s_{23})((m_3^2 - s_{12})^2(m_3^2 - s_{23}) - 4m_3^2 m_4^2 s_{12})} \longrightarrow \sqrt{(m^2 - s_{23})((m^2 - s_{12})^2(m^2 - s_{23}) - 4m^4 s_{12})}$$

We could not find a transformation to rationalise simultaneously all the square roots $\implies$ try a more **flexible approach to solve the DEs**

The alphabet gets larger!

- 37 letters in the general case (22 in the equal-mass case)
- 21 (11) **new letters** compared to the two-loop planar case
- New **algebraic letters** depending on **two square roots**:

$$r_1 = \sqrt{s_{12}^2 + (m_4^2 - m_3^2)^2 - 2s_{12}(m_4^2 + m_3^2)}$$

$$r_3 = \sqrt{s_{23}((m_3^2 - m_4^2)^2 s_{23} - 4m_3^2 m_4^2 s_{12})}$$

$$W_{r_1 r_3,1} = \frac{s_{23}(m_3^4 + m_4^2) - s_{12}s_{23}(m_3^2 + m_4^2) - 2m_3^2 m_4^2(s_{12} + s_{23}) - r_1 r_3}{s_{23}(m_3^4 + m_4^2) - s_{12}s_{23}(m_3^2 + m_4^2) - 2m_3^2 m_4^2(s_{12} + s_{23}) + r_1 r_3}$$

$$W_{r_1 r_3,1}^{(\mathrm{EM})} = \frac{s_{12}(m^2 + s_{23}) + r_1 r_3}{s_{12}(m^2 + s_{23}) - r_1 r_3}$$

- New **rational letters** with **higher degree**!

$$W_{22} = (m_3^2 m_4^2 + s_{13}(s_{12} + s_{13}))^2 - 4m_3^2 m_4^2 s_{13}^2$$

$$W_{13}^{(\mathrm{EM})} = (m^2 - s_{12})^2(m^2 - s_{23}) - 4m^4 s_{12}$$

[Canko, Syrrakos 2022] hinted to the **stability of the alphabet** with the loop order in the planar case

Evaluation of the MIs

Solve the DEs through **generalised series expansions**

Parametrise a **linear path**

$$\gamma(\eta) = (1-\eta)\vec{x}_i + \eta\vec{x}_f$$

$\Longrightarrow$ DEs A_η in one variable

Boundary point $\vec{x}_i$:

values from `AMFlow`

[Liu, Ma 2022]

Target point $\vec{x}_f$

Evaluation of the MIs

Solve the DEs through **generalised series expansions**

Parametrise a **linear path**

$$\gamma(\eta) = (1-\eta)\vec{x}_i + \eta \vec{x}_f$$

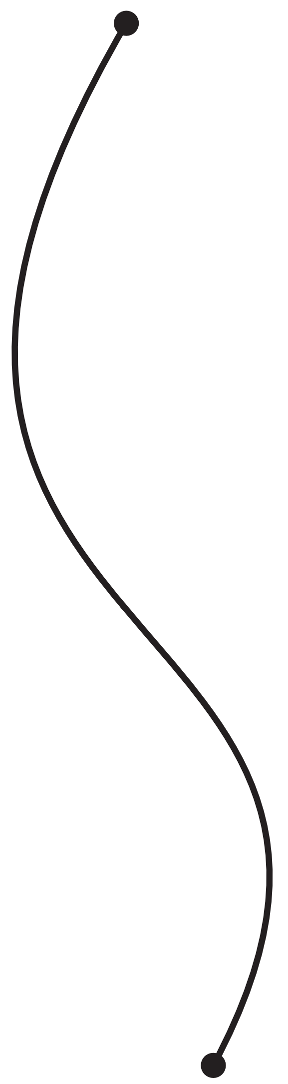
$\Longrightarrow$ DEs A_η in one variable

The solution admits an **expansion**

$$I \backsim I_0 + \sum A_\eta \, \log(\eta - \tau)^{k_1} (\eta - \tau)^{k_2}$$

$\Longrightarrow$ **evolution of the solution** using the DE solver of `AMFlow` and `DiffExp` [Hidding 2020]

Boundary point $\vec{x}_i$:
values from `AMFlow`
[Liu, Ma 2022]



Target point $\vec{x}_f$

Evaluation of the MIs

Solve the DEs through **generalised series expansions**

Parametrise a **linear path**

$$\gamma(\eta) = (1-\eta)\vec{x}_i + \eta\vec{x}_f$$

$\Longrightarrow$ DEs A_η in one variable

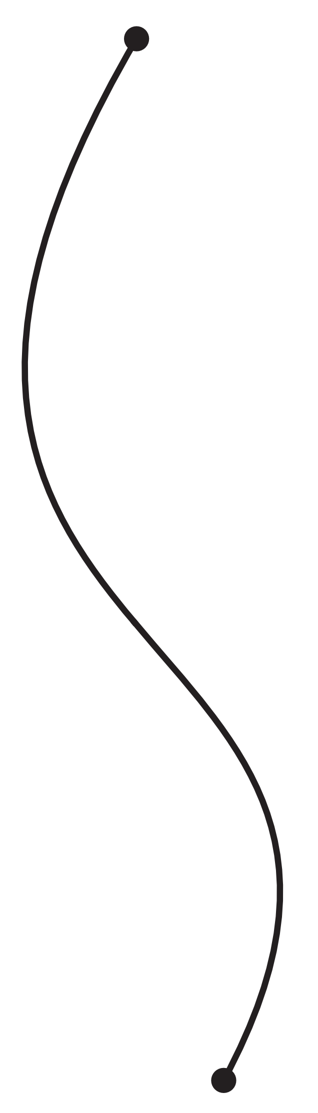
The solution admits an **expansion**

$$I \backsim I_0 + \sum A_\eta \; \log(\eta-\tau)^{k_1}(\eta-\tau)^{k_2}$$

$\Longrightarrow$ **evolution of the solution** using the DE solver of `AMFlow` and `DiffExp` [Hidding 2020]

Evaluation of a single family in the order of minutes per phase space point

Boundary point $\vec{x}_i$: values from `AMFlow` [Liu, Ma 2022]



Target point $\vec{x}_f$

Conclusion

- We computed all the **planar integral families** relevant for **diboson production** at **N3LO QCD** in the leading colour approximation
- For each family, we constructed a basis of MIs satisfying **canonical DEs**
- **New square roots** and letters appear, which were absent at two loops
- **Semi-numerical solution** using `DiffExp` and the DE solver of `AMFlow`
- Future work
  - **Optimisation** of the evaluation of the integrals **for phenomenology**
  - Computation of **three-loop amplitudes** for diboson production

Conclusion

- We computed all the **planar integral families** relevant for **diboson production** at **N3LO QCD** in the leading colour approximation
- For each family, we constructed a basis of MIs satisfying **canonical DEs**
- **New square roots** and letters appear, which were absent at two loops
- **Semi-numerical solution** using `DiffExp` and the DE solver of `AMFlow`
- Future work
  - **Optimisation** of the evaluation of the integrals **for phenomenology**
  - Computation of **three-loop amplitudes** for diboson production

Thank you!