Straightening Out the IBP Equations

2512.05923 [hep-ph]

Junhan William Liu,
Supervised by Prof. Alexander Mitov

December 2025

UNIVERSITY OF CAMBRIDGE

This work has been supported by the University of Cambridge Harding Distinguished Postgraduate Scholars Programme and the STFC DTP research studentship grant ST/Y509140/1.

Introduction

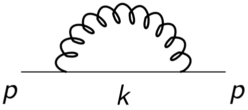
- The LHC is very demanding
  - Rapidly improving experimental precision!
  - NLO, NNLO or even N^3LO computation is necessary

- To have higher order computation of cross sections:
  - **Amplitude computation**
  - IR treatment (phase-space integration)

- To compute the needed amplitude, one needs two things:
  - **Integration-by-Parts (IBP) reduction**
  - Numerical/abstract evaluation of the master integrals

Amplitude computation

- To demonstrate how amplitudes can be computed, we use a simple example: one-loop bubble.



- QCD Feynman rules give:

$$\int d^d k \frac{(2-d)\not{k} + dm}{(k^2 - m^2)(p-k)^2}$$

  - Dimensional regularisation: $d = 4 - 2\epsilon$
  - Contains complicated numerator structure
  - Directly doable at one loop but very hard to evaluate with more loops

Feynman integrals

- The conventional technique is to decompose the matrix element into **Feynman Integrals**:

$$I_{\nu_1,\nu_2} = \int d^d k \frac{1}{(k^2 - m^2)^{\nu_1} (p-k)^{2\nu_2}}$$

  - The same propagators but raised to arbitrary power
- By partial fractioning, the matrix element becomes

$$\left[\frac{(2-d)\not{p}}{2m^2} (I_{0,1} - I_{1,0}) + dmI_{1,1} \right]$$

  - For complicated cases, the decomposition can contain hundreds or even thousands of integrals.
  - Inefficient to compute each one of them

IBP reduction

- There exist relations between Feynman integrals: **Integration-by-Parts** (IBP) identities.
- The IBP identities connect all integrals to a much smaller set of integrals: **master integrals**.
- In this example,

$$\frac{(2-d)\not{p}}{2m^2}(I_{0,1}-I_{1,0})+dmI_{1,1}\to\left(-\frac{(2-d)\not{p}}{2m^2}+\frac{d(d-2)}{2(d-3)m}\right)I_{1,0}$$

$$\{I_{0,1},I_{1,0},I_{1,1}\}\to\{I_{1,0}\}$$

- Once the IBP reduction is done, we are left with just the computation of the master integrals.

IBP equations

Here are the IBP identities for the one-loop bubble

$$\begin{aligned}0 = &-\nu_2\ I_{\nu_1-1,\nu_2+1}\\ &+(d-2\nu_1-\nu_2)\ I_{\nu_1,\nu_2}\\ &-2m^2\ \nu_1 I_{\nu_1+1,\nu_2}\end{aligned}$$

$$\begin{aligned}0 = &-\nu_2\ I_{\nu_1-1,\nu_2+1}\\ &-(\nu_1-\nu_2)\ I_{\nu_1,\nu_2}\\ &+\nu_1\ I_{\nu_1+1,\nu_2-1}\\ &-2m^2\nu_1\ I_{\nu_1+1,\nu_2}\end{aligned}$$

- Two independent equations
- Multi-variable recurrence relations
- Linear, homogenous and with rational coefficients
- Finite number of master integrals: one in this case.
- Highly tangled: the integrals in each equation contain shifts on multiple indices.

Solution of IBP equations

- Best known approach is proposed by Laporta (Laporta, 2000)
  - Substitute the abstract indices with explicit integer values
  - Produce an infinite system of equations, e.g.

$$\ldots, -I_{0,2} + I_{2,0} - 2m^2 I_{2,1} = 0, \ldots$$

  - Use Gauss elimination to solve a relevant part of it
- Limitation for more complicated problems:
  - Time-consuming: Months or even years for supercomputer with RAM consumption of several TB!
  - Huge size of simultaneous linear system of equations
- Numeric solution only:
  - It solves $I_{1,-1}$, $I_{1,-2}$ but not I_{ν_1,ν_2} for arbitrary $\nu_{1,2}$

Diagonalisation of IBP equations

- Diagonal form: changes one index at a time, keeping all other indices fixed (unshifted)

$$I_{\nu_1,\nu_2-1} = \frac{2m^2(d-\nu_1-2\nu_2)(1+d-\nu_1-2\nu_2)}{(2+d-2\nu_1-2\nu_2)(d-\nu_1-\nu_2)} I_{\nu_1,\nu_2}$$

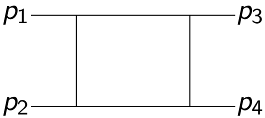
$$I_{\nu_1-1,1} = \frac{2m^2(d-\nu_1-2)(\nu_1-1)}{(d-2\nu_1)(d-1-\nu_1)} I_{\nu_1,1}$$

- Efficient solution: no simultaneous solution of multiple equations!
- One can even solve them for abstract ν_1, ν_2:

$$I_{\nu_1,\nu_2} = \frac{\Gamma(1-d+\nu_1+\nu_2)\Gamma\left(-\frac{d}{2}+\nu_1+\nu_2\right)}{\Gamma\left(2-\frac{d}{2}\right)\Gamma(\nu_1)\Gamma(1-d+\nu_1+2\nu_2)} \times (d-3)(-1)^{\nu_1}(m^2)^{2-\nu_1-\nu_2} I_{1,1}$$

What about multi-master cases?

- What about system with multiple masters?
    - Higher order?
    - Anything alternative?
- Let's consider the following example:



$$I_{\nu_1,\nu_2,\nu_3,\nu_4} = \int d^d k \frac{1}{D_1^{\nu_1} D_2^{\nu_2} D_3^{\nu_3} D_4^{\nu_4}}$$

- **Three** masters: two bubbles and a box

Matrix diagonal form

- One can derive diagonal equations with higher (third) order, but we found it better to do it in an alternative way (more in the backups).
- Consider recurrence in a **different object** but kept **first order**

$$\boldsymbol{V}_{\nu_1-1,\nu_2,\nu_3,\nu_4} = \boldsymbol{W}^{(4)}(\nu_1,\nu_2,\nu_3,\nu_4)\boldsymbol{V}_{\nu_1,\nu_2,\nu_3,\nu_4}$$

$$\begin{pmatrix} I_{\nu_1-1,\nu_2,\nu_3,\nu_4} \\ I_{\nu_1-2,\nu_2,\nu_3,\nu_4} \\ I_{\nu_1-1,\nu_2-1,\nu_3,\nu_4} \end{pmatrix} = \begin{pmatrix} W_{11}^{(4)} & W_{12}^{(4)} & W_{13}^{(4)} \\ W_{21}^{(4)} & W_{22}^{(4)} & W_{23}^{(4)} \\ W_{31}^{(4)} & W_{32}^{(4)} & W_{33}^{(4)} \end{pmatrix} \begin{pmatrix} I_{\nu_1,\nu_2,\nu_3,\nu_4} \\ I_{\nu_1-1,\nu_2,\nu_3,\nu_4} \\ I_{\nu_1,\nu_2-1,\nu_3,\nu_4} \end{pmatrix}$$

- Each line of these vector equations still contains four integrals: "third order" comes back as $\dim(\boldsymbol{V})$.

Matrix diagonal form

- Such equation can be derived for all four indices:

$$\begin{aligned}
\boldsymbol{V}_{\nu_1-1,\nu_2,\nu_3,\nu_4} &= \boldsymbol{W}^{(4)}(\nu_1,\nu_2,\nu_3,\nu_4)\boldsymbol{V}_{\nu_1,\nu_2,\nu_3,\nu_4} \\
\boldsymbol{V}_{1,\nu_2-1,\nu_3,\nu_4} &= \boldsymbol{W}^{(3)}(\nu_2,\nu_3,\nu_4)\boldsymbol{V}_{1,\nu_2,\nu_3,\nu_4} \\
\boldsymbol{V}_{1,1,\nu_3-1,\nu_4} &= \boldsymbol{W}^{(2)}(\nu_3,\nu_4)\boldsymbol{V}_{1,1,\nu_3,\nu_4} \\
\boldsymbol{V}_{1,1,1,\nu_4-1} &= \boldsymbol{W}^{(1)}(\nu_4)\boldsymbol{V}_{1,1,1,\nu_4}
\end{aligned}$$

- The explicit solution of $\boldsymbol{V}_{\nu_1,\nu_2,\nu_3,\nu_4}$ can then be achieved by iterative matrix multiplication

$$\boldsymbol{V}_{\nu_1\nu_2,\nu_3,\nu_4} = \left(\prod_{l=1}^{\nu_1+1}\boldsymbol{W}^{(\mathbf{4})}(l,\nu_2,\nu_3,\nu_4)\right)\left(\prod_{k=1}^{\nu_2+1}\boldsymbol{W}^{(\mathbf{3})}(k,\nu_3,\nu_4)\right)\times \left(\prod_{n=1}^{\nu_3+1}\boldsymbol{W}^{(\mathbf{2})}(n,\nu_4)\right)\left(\prod_{m=1}^{\nu_4+1}\boldsymbol{W}^{(\mathbf{1})}(m)\right)\boldsymbol{V}_{1,1,1,1}$$

Triangular form

- Advantages of the diagonal form:
  - Fast solution to I_{-100}, if you want
  - Analytic continuation to non-integer values
- Disadvantages of diagonal form:
  - Large rational function in $\boldsymbol{W}$
  - **Note:** a better choice of $\boldsymbol{V}$ may significantly simplify $\boldsymbol{W}$
- To improve computational efficiency: **triangular form**
  - No diagonalisation any more
  - No simultaneous solution of multiple equations
  - Much simpler rational functions in the equations

Triangular form

- Efficient reduction in complicated problems:
  - two-loop five-point planar (more benchmarks in the paper)
    - Five years ago, the very same task took more than a year
  - Benchmarked against `NeatIBP` (syzygy) (Wu et al., 2025) using the program `Kira` (Lange, Usovitsch and Wu, 2025)

Fermat	Finite field	
triang.	syzygy	triang.
3.4 h	11.3 h	8.0 h

- Timing breakdown for the Finite field approach

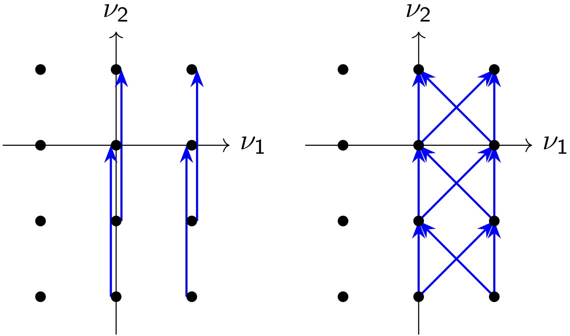
Step per probe	syzygy	triang.
Substitution	2.1%	92.7%
Solution	97.9%	7.3%

Conclusions

- We proposed an automated algorithm that achieved diagonalisation of the IBP system (**never achieved before**)
  - Potential pathway to find analytical solution to the IBP identities (ongoing)
  - Systematic approach to investigate properties of functions (more in the backups)
- As a by-product, we presented the triangular approach
  - Efficiency-oriented
  - A factor of 2 improvement in efficiency compared to the state-of-the-art algorithm (subject to additional improvements)
  - More state-of-the-art computation...

Backup: Matrix diagonal form

- Difference between higher-order diagonal (left) and matrix diagonal (right)



- It is realised with higher-order equation, one will end up with more "master integrals" than necessary (Mitov, 2005, Mitov and Moch, 2006)

Backup: Something to do with Gauss

- Contiguous relations of ${}_2F_1$-hypergeometric function

$$(c-a-1)F(a,b,c)+aF(a+1,b,c)-(c-1)F(a,b,c-1)=0,\ldots$$

- Through standard IBP pipeline, one can find
  - Finite solution
  - Two boundary conditions
    $F(1,1,1)=\frac{1}{1-z}$ $\quad F(1,1,2)=-\frac{\log(1-z)}{z}$
- Same matrix diagonal form can be derived

$$\begin{pmatrix} F(a+2,b,c) \\ F(a+1,b,c) \end{pmatrix} = \begin{pmatrix} \frac{a+bz-cz}{a(1-z)} & \frac{(a-c)(b-c)\,z}{ac(1-z)} \\ \frac{c}{a} & \frac{a-c}{a} \end{pmatrix} \begin{pmatrix} F(a+1,b,c) \\ F(a,b,c) \end{pmatrix}$$

  - Vanishing second column at $a=c$:
  - decoupling from the $\log(1-z)$ (textbook results)