Latest IBP reduction techniques

Tiziano Peraro (Università di Bologna & INFN)

Precision calculations for future e^+e^- colliders: targets and tools

CERN, 13 June 2022

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

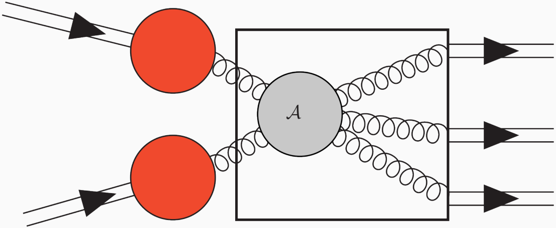
Theory and Phenomenology
of Fundamental Interactions
UNIVERSITY AND INFN · BOLOGNA

INFN
Istituto Nazionale di Fisica Nucleare

New challenges in collider physics

- Theoretical predictions at %-level precision
  - search of new physics
  - test symmetry breaking mechanism of SM
- Crucial high-multiplicity & massive multi-loop processes

Scattering amplitudes



- Main process-dependent part of a physical event
- They can be computed in perturbation theory

$$\mathcal{A} \sim \mathcal{A}_{\text{tree}} + \alpha\, \mathcal{A}_{\text{1-loop}} + \alpha^2\, \mathcal{A}_{\text{2-loops}} + \dots$$

- precision $\Rightarrow$ loops (% level $\sim$ 2 loops)
- energy $\Rightarrow$ high multiplicity and mass scales

State of the art of loop calculations

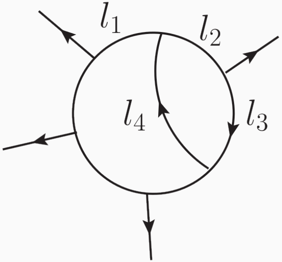
- Tree-level and 1 loop
  - "essentially" solved and automated
- 2 loops
  - great recent progress in 2-loop 5-point processes
    - massless internal and external states (max. 1 off-shell leg)
  - great progress in 4-pt processes with internal masses
- 3 loops
  - recent $2 \to 2$ results in massless QCD
- 4+ loops
  - mostly form factors, anomalous dimensions…
- Complexity: # loops, # legs, external/internal masses

Loop amplitudes

- An integrand contribution to ℓ-loop amplitude

$$\mathcal{A} = \int_{-\infty}^{\infty} \left(\prod_{i=1}^{\ell} d^d k_i \right) \frac{\mathcal{N}}{D_1\, D_2\, D_3 \cdots}$$

  - rational function in the components of loop momenta k_j
  - polynomial numerator $\mathcal{N}$
  - quadratic denominators corresp. to loop propagators



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

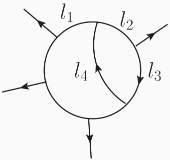
Computing amplitudes: Step 1/3

- Write amplitudes as l.c. of Feynman integrals

$$\mathcal{A} = \sum_j a_j I_j$$

- Dependence on particle-content in rational coeff.s a_j
- The integrals should have a "nice" / "standard" form

$$I = \int_{-\infty}^{\infty} \left(\prod_{i=1}^{\ell} d^d k_i \right) \frac{1}{D_1^{\nu_1} \, D_2^{\nu_2} \, D_3^{\nu_3} \dots}, \qquad \nu_j \lesseqgtr 0$$



$$D_j = \begin{cases} l_j^2 - m_j^2 \\ l_j \cdot v_j - m_j^2 \end{cases}$$

Non-trivial at
high multiplicity

Getting the "nice" form: Integrand reduction

Ossola, Papadopoulos, Pittau (2007)

- generic contribution to a loop amplitude

$$\int_{-\infty}^{\infty} \left(\prod_{i=1}^{\ell} d^d k_i \right) \frac{\mathcal{N}(k_i)}{\prod_j D_j(k_i)},$$

- integrand reduction (integrand as sum of irreducible contributions)

$$\frac{\mathcal{N}(k_i)}{\prod_j D_j(k_i)} = \sum_{T \in \text{topologies}} \sum_{\alpha} \frac{c_{T,\alpha} \, (\boldsymbol{m}_T(k_i))^{\alpha}}{\prod_{j \in T} D_j(k_i)}$$

- the on-shell integrands $\{\boldsymbol{m}_T^{\alpha}\}$
  - form a complete integrand basis
  - can have the "nice"/"standard" form we want
- fit unknown $c_{T,\alpha}$
  - on multiple cuts $\{D_j = 0\}_{j \in T}$ (linear system)
  - black-box polynomial reconstruction in $\{D_j\}$ [T.P. (2019)]

Getting the "nice" form alt.: Physical projectors

- Well-known decomposition of amplitudes in tensors and form factors

$$\mathcal{A} = \sum_j F_j \, T_j$$

  - T_j tensor structures contracted with external polarizations
  - F_j scalar form factors

- Projecting out the form factors

$$F_j = P_j \cdot \mathcal{A} \qquad \text{with} \quad P_j = \sum_k (T^\dagger \cdot T)^{-1}_{jk} \;\; T_k^\dagger$$

  - traditionally impractical for ≥ 5 legs

- Use four-dimensional external polarization states
  [Chen (2019), Tancredi, T.P. (2019,2021)]
  - T_j span physical space for four-dim. polarizations
  - scales much better with multiplicity
  - #indep. structures $\simeq$ #indep. helicity amplitudes

Computing amplitudes: Step 2/3

Chetyrkin, Tkachov (1981), Laporta (2000)

- Feynman integrals obey linear relations, e.g. IBPs

$$\int \Big(\prod_j d^d k_j\Big) \frac{\partial}{\partial k_j^\mu}\, v^\mu \,\frac{1}{D_1^{\nu_1}\, D_2^{\nu_2} \cdots} = 0, \qquad v^\mu = \begin{cases} p_i^\mu & \text{external} \\ k_i^\mu & \text{loop} \end{cases}$$

- Very large and sparse linear systems
- Reduce to linearly independent Master Integrals (MIs) $\{G_1, G_2, \ldots\} \subset \{I_j\}$

$$I_j = \sum_k c_{jk}\, G_k$$

Computing amplitudes: Step 3/3

- The MIs can often be computed analytically
  - in terms of special functions (MPLs, elliptic, . . .)
  - most effective method is differential equations (DEs)
    Kotikov (1991), Gehrmann, Remiddi (2000)

$$\partial_x G_i = \sum_j A_{ij}^{(x)} G_j$$

- Great recent progress on (semi-)numerical methods
  - differential equations (see e.g. DiffExp, AMFlow) (also relies on reduction to MIs)
  - sector decomposition [Binoth, Heinrich (2000)]

Analytic vs algebraic complexity

Analytic complexity

- understanding space of special functions for amplitudes
  (especially with massive internal lines)
- appears in: step 3 (computing MIs, e.g. by solving DEs)

Algebraic complexity

- huge intermediate expressions
- appears in: step 1, step 2 and parts of step 3
  (e.g. deriving DEs or expanding the amplitude)

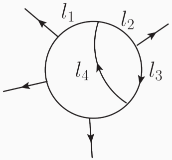
$\Rightarrow$ this talk!

IBP REDUCTION

Definitions: integral families

- An integral family

$$I(\nu_1, \ldots, \nu_n) = \int_{-\infty}^{\infty} \left(\prod_{i=1}^{\ell} d^d k_i \right) \frac{1}{D_1^{\nu_1} \cdots D_n^{\nu_n}}, \qquad \nu_j \lesseqgtr 0$$



$$D_j = \begin{cases} l_j^2 - m_j^2 \\ l_j \cdot v_j - m_j^2 \end{cases}$$

- may include auxiliary denomonators
- all scalar products $k_i \cdot k_j$ and $k_i \cdot p_j$ are lin. comb. of D_j

Definitions: sectors

$$I(\nu_1, \ldots, \nu_n) = \int \frac{1}{D_1^{\nu_1} \cdots D_n^{\nu_n}}$$

- the sector $\mathcal{S}$ of an integral is identified by the set of denominators with positive exponents

$$\mathcal{S}\left(I(\nu_1, \ldots, \nu_n)\right) = \{j | \nu_j > 0\}$$

- linear identities usually generated sector by sector
- symmetries/mappings only valid for specific sectors

Laporta algorithm (in a nutshell)

- Integration by parts (IBP)

$$\int \frac{\partial}{\partial k_j^\mu} v^\mu \frac{1}{D_1^{\nu_1} D_2^{\nu_2} \cdots} = 0, \qquad v^\mu = \begin{cases} p_i^\mu & \text{external} \\ k_i^\mu & \text{loop} \end{cases}$$

  and Lorentz invariance (LI) valid for all sectors

- symmetries/mappings (shifts of k_j that re-map denominators) specific to each sector

Laporta algorithm

- generate many identities for specific choices of v^μ, ν_i
- define an ordering (or weight) for the integrals
- solve the (large and sparse) system of equations
- higher weight integrals in terms of lower weight integrals

Laporta algorithm

Pros

- systematic, general and easy to automate
  - public implementations available (Reduze, Fire, Kira)
- process independent approach

$\Rightarrow$ the most popular and successful reduction method

Cons/Challenges

- linear systems can get huge (with more loops/legs)
- reduction tables can be large/complicated (esp. with more scales)
- large intermediate algebraic expressions

$\Rightarrow$ often a major bottleneck in modern predictions

Symbolic reduction

- The idea: solve the identities for generic exponents
- Example:

$$I(\nu_1,\nu_2) \equiv \int d^dk \, \frac{1}{(k^2)^{\nu_1}\,((k+p)^2)^{\nu_2}}$$

one finds e.g.

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

$$I(\nu_1,\nu_2) = I(\nu_2,\nu_1), \quad \text{etc.}\ldots$$

- shifts of symbolic identities often required (Laporta-like)
- harder to automate (available in LiteRed [R. Lee])
- not widely used

FINITE FIELDS AND RATIONAL RECONSTRUCTION

Finite fields and functional reconstruction

The main idea

- reconstruct analytic results from numerical evaluations
- evaluation over finite fields $\mathcal{Z}_p$ (i.e. modulo prime integers p)

$$\mathcal{Z}_p = \{0, 1, 2, \ldots, p-1\}$$

- use machine-size integers, $p < 2^{64} \Rightarrow$ fast and exact
- collect numerical evaluations and infer analytic result
- first applications
  - linear solvers, IBPs and univariate reconstruction

    Kant (2014), von Manteuffel, Schabinger (2014)
  - helicity amplitudes and multivariate reconstruction T.P. (2016)

Some notable examples

- FinRed (private) [von Manteuffel]
  - reduction for 4-loop form factors [von Manteuffel, Schabinger]
  - 2-loop 5-pt red.s [Buccioni, Caola, von Manteuffel, Tancredi, et al.]
- FiniteFlow(computational framework) [T.P.]
  - 2-loop 4- and 5-point amplitudes (incl. non planar) [Badger, Brønnum-Hansen, Gehrmann, Hartanto, Henn, Zoia, T.P. et al.]
  - Light-like four-loop cusp anomalous dimension [Henn, T.P., Stahlhofen, Wasser]
  - implementation of many new methods (finding identities, DEs. . .)
- Caravel [Abreu, Dormans, Febres Cordero, Ita, Page, Sotnikov, Zeng]
  - analytic five-point amplitudes
- FireFly [Klappert, Lange])
  - used e.g. by Kira Maierhöfer, Usovitsch, Uwer et al.
- Fire 6 [A.V. Smirnov, F.S. Chuharev]
  - 4-loop form factors [Lee, Smirnov, Smirnov, Steinhauser]

The black-box interpolation problem

Given a rational function f in the variables $\boldsymbol{z} = (z_1, \ldots, z_n)$ over $\mathcal{Q}$

- Reconstruct analytic form of f, given a numerical procedure

$$(\boldsymbol{z}, p) \longrightarrow \boxed{\; f \;} \longrightarrow f(\boldsymbol{z}) \bmod p.$$

- evaluate f numerically for several $\boldsymbol{z}$ and p
- efficient multivariate reconstruction algorithms exist
  e.g. T.P. (2016,2019), Klappert, Lange (2019)
- upgrade analytic f over $\mathcal{Q}$ using rational reconstruction algorithm [Wang (1981)] and Chinese remainder theorem

$$\mathcal{Z}_{p_1}, \mathcal{Z}_{p_2}, \ldots \to \mathcal{Z}_{p_1 p_2 \cdots} \to \mathcal{Q}$$

- sidesteps large intermediate expressions & highly parallelizable
- applicable to any rational algorithm

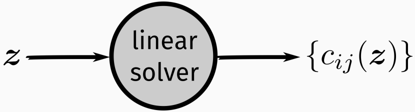
Example: Linear solver

- A $n \times m$ linear system with parametric rational entries

$$\sum_{j=1}^{m} A_{ij}\, x_j = b_i, \quad (i = 1, \ldots, n), \qquad A_{ij} = A_{ij}(\boldsymbol{z}), \quad b_i = b_i(\boldsymbol{z})$$

- input: list of values for paramers $\boldsymbol{z} = (z_1, \ldots, z_n)$
- output: solution $c_{ij} = c_{ij}(\boldsymbol{z})$ such that

$$x_i = \sum_{j \in \text{indep}} c_{ij}\, x_j + c_{i0} \qquad (i \notin \text{indep})$$



- can be part of a larger algorithm

Repeated numerical evaluations over finite fields

Efficient implementations (over finite fields) of numerical algorithms:

- use numerical evaluations to learn/optimize subsequent ones

Example: Linear solver

- learn linearly dependent/independent unknowns
- learn lin. dep. equations (that reduce to $0 = 0$)
- filter equations for needed unknowns
  - often only a subset of the unknown is needed
  - track substitutions during (numerical Gauss elimination)
  - filter out equations that are never substituted (directly or indirectly) into solutions for needed unknowns

Finite fields + Rational reconstruction

- Applicable to any rational algorithm
- Sidesteps appearance of large intermediate expressions
- Massively parallelizable
  - numerical evaluations are independent of each other
  - algorithm-independent parallelization strategy
- Yielded some of the most impressive multi-loop results to date

Reduction to master integrals

- IBPs are large and sparse linear systems
- they reduce Feynman integrals I_j to a lin. indep. set of MIs G_j

$$I_i = \sum_j c_{ij}\, G_j$$

- amplitudes and other multi-loop objects can be reduced mod IBPs

$$\mathcal{A} = \sum_j a_j\, I_j = \sum_{jk} a_j\, c_{jk}\, G_k = \sum_j A_j\, G_j, \quad \text{with } A_j = \sum_k a_k\, c_{kj}$$

- final results for A_k often much simpler than c_{ij}

$\Rightarrow$ solve IBPs numerically and compute A_j via a matrix multiplication

Differential equations for MIs

- The MIs G_k satisfy differential equations

$$\partial_x G_i = \sum_j A_{ij}^{(x)} G_j$$

- Identify MIs G_i (e.g. by solving IBPs numerically)
- Compute their derivatives in terms of (non-master) loop integrals

$$\partial_x G_i = \sum_j a_{ij}^{(x)} I_j$$

- Reduce the needed integrals modulo IBPs: $I_i = \sum_j c_{ij} G_j$
- The final result is given by a matrix multiplication

$$A_{ij}^{(x)} = \sum_k a_{ik}^{(x)} c_{kj}$$

- Reconstruct $A_{ij}^{(x)}$ analytically from its numerical evaluations

Coefficients of the ϵ-expansion

- If MIs are known in terms of special functions f_k

$$G_j = \sum_k g_{jk}(\epsilon, x)\, f_k + \mathcal{O}(\epsilon),$$

we compute

$$\mathcal{A} = \sum_k u_k(\epsilon, x)\, f_k + O(\epsilon), \quad \text{where } u_k(\epsilon, x) = \sum_j A_j(\epsilon, x)\, g_{jk}(\epsilon, x)$$

- what we want is the ϵ-expansion of the $u_k(\epsilon, x)$

$$u_k(\epsilon, x) = \sum_{j=-p}^{0} u_k^{(j)}(x)\, \epsilon^j + \mathcal{O}(\epsilon),$$

- reconstruct in ϵ only and expand to evaluate $u_k^{(j)}(x)$

Partial fractions

- reconstructed results (usually) come out collected and GCD-simplified
- partial fractioning is known to yield simplifications
- multivariate partial fractions require some care
  - uniqueness of result
  - avoid spurious singularities

$$f(x_i) = \frac{n(x_i)}{d_1(x_i)^{a_1} \cdots d_n(x_i)^{a_n}}$$

- modern implementations require some algebraic geometry
  [Abreu, Dormans, Cordero, Ita, Page, Sotnikov (2019)
  Boehm, Wittmann, Wu, Xu, Zhang (2020)
  Heller, von Manteuffel (MultivariateApart,2021)]

  $$1/d_j(x_i) \to q_j \quad \Rightarrow \quad \text{reduction mod } \langle q_1 d_1(x_i) - 1, \ldots, q_n d_n(x_i) - 1 \rangle$$

  ordering: $q_j > x_i$ (exact choice can affect the result)

Partial fractions and rational reconstruction

- Partial fractioned results are simpler, but they require prior analytic knowledge of full result
- Simplifying the reconstruction
  - guess denominator factors e.g. from the "letters" l_k

$$u(x_i) = \frac{n(x_i)}{\prod_k l_k(x_i)^{\alpha_k}}$$

    by reconstructing it over univariate slices $x_i = a_i \tau + b_i$
  - reconstruct w.r.t. one variable, say x_1 (or two, say x_1, x_2)
  - partial fraction univariate (or bivariate) result
  - reconstruct dependence w.r.t. all the other x_i
- applied e.g. $gg \to g + \gamma\gamma$ at 2 loops

  Badger, Brønnum-Hansen, Chicherin, Gehrmann, Hartanto, Henn, Marcoli, Moodie, Zoia, T.P. (2021)

SYZYGIES

Lowering the complexity of IBP systems

- IBP relations contain integrals with higher-powers of propagators

$$0 = \int \frac{\partial}{\partial k_j^\mu} v^\mu \frac{1}{D_1^{\nu_1} D_2^{\nu_2} \cdots} = -\nu_1 \int \left(v^\mu \frac{\partial D_1}{\partial k_j^\mu} \right) \frac{1}{D_1^{\nu_1+1} D_2^{\nu_2} \cdots} + \cdots$$

- many of these don't contribute to the amplitude
- can we build a system without them? [Gluza, Kajda, Kosower (2011)]

$$\sum_j \int \frac{\partial}{\partial k_j^\mu} v_j^\mu \frac{1}{D_1^{\nu_1} D_2^{\nu_2} \cdots} = 0, \qquad v_j^\mu = \sum_m \alpha_{jm} p_m^\mu + \sum_n \beta_{jn} k_n^\mu$$

$$\sum_j v_j^\mu \frac{\partial D_i}{\partial k_j^\mu} = \gamma_i D_i, \qquad \text{for all } i \text{ with } \nu_i > 0$$

$\Rightarrow$ syzygy equations for polynomials
$\alpha_{jm} = \alpha_{jm}(D_i)$, $\beta_{jm} = \beta_{jm}(D_i)$, $\gamma_j = \gamma_j(D_i)$

Syzygy equations

- A syzygy equation has the form

$$\mathbf{f}(\boldsymbol{z}) \cdot \mathbf{g}(\boldsymbol{z}) = \sum_{j=1}^{n} f_j(\boldsymbol{z})\, g_j(\boldsymbol{z}) = 0$$

  - $\mathbf{f} = \{f_1, \ldots, f_n\}$ list of known polynomials
  - $\mathbf{g} = \{g_1, \ldots, g_n\}$ list of unknown polynomials

- can be solved via linear algebra by making an ansatz for g_j [see also Schabinger (2012)]

$$g_j(\boldsymbol{z}) = \sum_{|\alpha|<\alpha_{\max}} c_{j,\alpha}\, \boldsymbol{z}^{\alpha}$$

- if $\mathbf{g}^{(1)}, \ldots, \mathbf{g}^{(M)}$ are generators of the solutions, then any solution

$$\mathbf{g}(\boldsymbol{z}) = \sum_{j=1}^{M} p_j(\boldsymbol{z})\, \mathbf{g}^{(j)}(\boldsymbol{z})$$

  with p_j polynomials

IBPs in the Baikov representation

- The Baikov representation ($k_j^\mu \to z_i = D_i$)

$$I = \int \left(\prod_{i=1}^{\ell} d^d k_i \right) \frac{1}{D_1^{\nu_1} \cdots D_n^{\nu_n}} = C \int dz_1 \cdots dz_n \frac{B(z_1, \ldots, z_n)^\gamma}{z_1^{\nu_1} \cdots z_n^{\nu_n}}$$

  with $B =$ Baikov polynomial (a Gram determinant) and $\gamma = (d - \ell - e - 1)/2$.

- IBPs

$$0 = \sum_j \int \frac{\partial}{\partial z_j} \left(B^\gamma \frac{a_j(z_1, \ldots, z_n)}{z_1^{\nu_1} \cdots z_n^{\nu_n}} \right)$$

$$0 = \sum_j \int \left(\frac{\partial a_j}{\partial z_j} + \underbrace{\frac{\gamma}{B} a_j \frac{\partial B}{\partial z_j}}_{\text{dim. shifted}} - \underbrace{\nu_j \frac{a_j}{z_j}}_{\text{higher powers}} \right) \frac{B^\gamma}{z_1^{\nu_1} \cdots z_n^{\nu_n}}$$

IBPs in the Baikov representation

Ita (2016), Larsen, Zhang (2016)

$$\sum_j \int \left(\frac{\partial a_j}{\partial z_j} + \underbrace{\frac{\gamma}{B} a_j \frac{\partial B}{\partial z_j}}_{\text{dim. shifted}} - \underbrace{\nu_j \frac{a_j}{z_j}}_{\text{higher powers}} \right) \frac{B^\gamma}{z_1^{\nu_1} \cdots z_n^{\nu_n}} = 0$$

- Syzygy equations

$$\sum_j a_j \frac{\partial B}{\partial z_j} = b_0 \, B \qquad (1)$$

$$a_j = z_j \, b_j \qquad \text{(for sectors with } \nu_j > 0\text{)} \qquad (2)$$

for polynomials $b_j = b_j(z_1, \ldots, z_n)$

- closed form solutions for (1) and (2) can be (separately) found [Böhm, Georgoudis, Larsen, Schulze, Zhang (2018)]

IBPs in the Baikov representation

Three different approaches:

1. plug (2) into (1) and solve (1)
2. combine solutions of (1) and (2) (module intersections)
   [Böhm, Georgoudis, Larsen, Schönemann, Zhang (2018)]
3. take solution of (1) and remove higher-powers of denominators via Gauss elimination [von Manteuffel]
   - 1-to-1 map btw. solutions into linear equations

$$\mathbf{g}(\boldsymbol{z}) = \sum_{k\alpha} c_{k\,\alpha} \boldsymbol{z}^{\alpha}\, \hat{e}_k \quad \longleftrightarrow \quad \sum_{k\alpha} c_{k\,\alpha} x_{k\,\alpha} = 0$$

   - Gauss eliminate unwanted terms ⇔ combination of syzygies
   - exploit solvers and rational reconstruction
   - avoid reconstruction of complicated solutions

IBPs in the Baikov representation

- Syzygies yield new parametric identities for each sector
- Then proceed as in traditional Laporta alg.
- Identities can be used in integrand bases
  (see e.g. numerical unitarity [Ita et al. (2016)])
- Can be combined with traditional Laporta identities
  (e.g. to fix incomplete reductions or avoid complex syzygy solutions)
- Up to $\sim 10x$ improvements in efficiency

OTHER APPROACHES

The vector space of Feynman integrals

- A family of Feynman integrals $\{I\}$ is a vector space
- Master Integrals $\{G_j\}$ are a basis of this vector space

$$I = \sum_k c_k \, G_k$$

with rational coefficients $\{c_k\}$

- Can we directly project into this basis?
- A scalar product would do the job

$$c_k = \sum_j (G^{-1})_{kj} \, (G_j \cdot I), \qquad \text{with } G_{jk} \equiv G_j \cdot G_k$$

Intersection theory

- Consider a set of integrals

$$\langle \phi | \sim \int \phi(\boldsymbol{z})\, u(\boldsymbol{z})\, d^n z, \qquad u(\boldsymbol{z}) = \prod_j B_j(\boldsymbol{z})^{\gamma_j}$$

  with γ_j "generic" exponents, u polynomial, ϕ rational

- Invariance under IBPs

$$\phi \to \phi + \mathcal{D}_i\, \xi, \qquad \mathcal{D}_i\, \xi \equiv \left(\partial_{z_i} + \frac{\partial_{z_i} u}{u} \right) \xi$$

- Dual space $|\phi\rangle \sim$ uses $u \to u^{-1}$

- Intersection numbers: a scalar product (univariate case)

$$\langle \phi_L | \phi_R \rangle = \mathrm{Res}_p \Big[\psi_p \phi_R \Big]$$

$$\partial_z \psi_p + \frac{\partial u}{u} \psi_p = \phi_L \qquad \text{close to } z \sim p$$

  with $p \in \{$poles of $\partial_z u/u\}$

A scalar product for Feynman integrals (and more)

Mastrolia, Mizera (2019)

Frellesvig, Gasparotto, Laporta, Mandal, Mastrolia, Mattiazzi, Mizera (2021)

- Intersection theory $\Rightarrow$ requires a suitable integral form
- Example: Baikov representation

$$I \sim \langle \phi | \sim \int u(\boldsymbol{z})\, \phi(\boldsymbol{z})\, d^n \boldsymbol{z}$$

with

$$u = B^\gamma, \qquad \phi = \frac{1}{z_1^{\nu_1} \dots z_n^{\nu_n}}$$

- Int. numbers can be generalized to multivariate case (recursive algorithm)
- has been shown to work in many examples

Intersection Theory and Feynman Integrals: Outlook

- Manifest vector-space structure of integral families
- Some open questions and a few conceptual issues
- Can we implement it in a rational algorithm?
  - poles can be in non-rational positions
  - polynomial algebra + global residues [see e.g. Weinzierl (2020)]
- Missing public/efficient implementations
- An interesting direction to explore

Summary & Outlook

Summary

- IBP reduction is essential for multi-loop predictions
- Great recent progress in Laporta-like approaches
- Finite fields allow to delay analytic reconstruction
  (avoid large intermediate expressions)
- Syzygies reduce complexity of the systems

Outlook

- Improving Laporta-like or parametric reductions
- Exploring alternatives, e.g. approaches for direct reduction

BACKUP SLIDES

Rational reconstruction

Functional reconstruction

Reconstruct the monomials $\boldsymbol{z}^\alpha$ and their coefficients from numerical evaluations of the function (over finite fields)

- from $\mathcal{Q}$ to $\mathcal{Z}_p$:
  $q = a/b \in \mathcal{Q} \quad \longrightarrow \quad q \text{ mod } p \equiv a \times (b^{-1} \text{ mod } p) \text{ mod } p$
- how to go back from $\mathcal{Z}_p$ to $\mathcal{Q}$?
  - rational reconstruction algorithm: given $c \in \mathcal{Z}_n$ find its pre-image $q = a/b \in \mathcal{Q}$ with "small" a, b [Wang (1981)]
  - it's correct when $a, b \lesssim \sqrt{n}$
- make n large enough using Chinese reminder theorem
  - solution in $\mathcal{Z}_{p_1}, \mathcal{Z}_{p_2} \ldots \Rightarrow$ solution in $\mathcal{Z}_{p_1 p_2 \ldots}$

Rational reconstruction: example

- Reconstruct $q = -611520/341$ from its images over finite fields
- $\mathcal{Z}_{p_1}$, with $p_1 = 897473$

$$a_1 = q \text{ mod } p_1 = 13998,$$

$$\text{first guess: } a_1 \xrightarrow{\text{rational rec. over } \mathcal{Z}_{p_1}} g_1 = -411/577$$

- $\mathcal{Z}_{p_2}$, with $p_2 = 909683$

$$a_2 = q \text{ mod } p_2 = 835862$$

$$g_1 \text{ mod } p_2 = 807205 \quad \Rightarrow \quad \text{guess } g_1 \text{ is wrong}$$

- Chinese reminder: $a_1, a_2 \longrightarrow a_{12} \in \mathcal{Z}_{p_1 p_2}$, with $p_1 p_2 = 816415931059$

$$a_{12} \equiv q \text{ mod } p_1 p_2 = 629669763217 \xrightarrow{\text{rational rec. over } \mathcal{Z}_{p_1 p_2}} g_2 = -\frac{611520}{341}$$

- calculation over other fields $\mathcal{Z}_{p_3}, \ldots$ confirm the guess g_2