A new method for the reconstruction of rational functions

Xiao Liu

University of Oxford

Based on e-print: 2306.12262

DESY-HU Theory Seminars
13 July 2023, online

Outline

I. Introduction

II. The method

III. Examples

IV. Summary and outlook

High precision particle physics

➢ Era of Precision

- Theoretical predictions & experimental measurements
- Test particle physics Standard Model & probe signals of New Physics
- Higher order corrections urgently needed

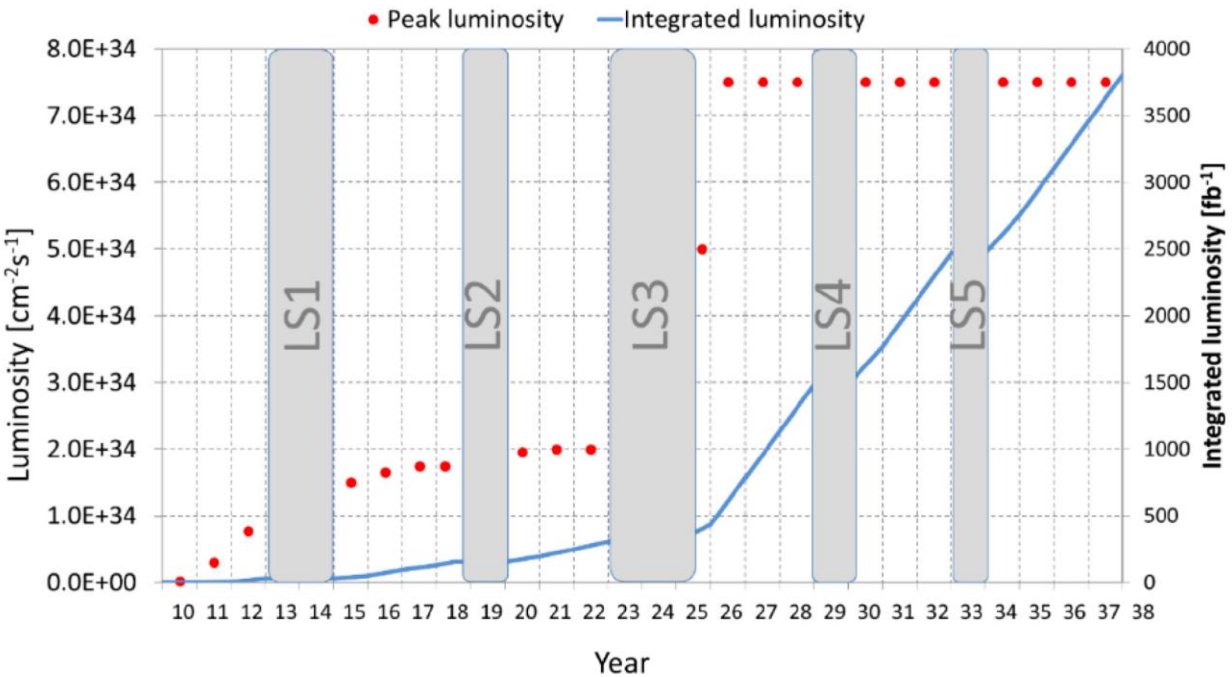


*Figure from [Apollinari, Alonso, Bruning, et al, 2015]

High precision particle physics

State-of-the-art

- NLO revolution [Ossola, Papadopoulos and Pittau, Nucl. Phys. B, 2007] …
- NNLO QCD:
  - $pp \to \gamma\gamma\gamma$ [Chawdhry, Czakon, Mitov, et al, JHEP, 2020]
  - $pp \to \gamma\gamma j$ [Chawdhry, Czakon, Mitov, et al, JHEP, 2021]
  - $pp \to Wb\bar{b}$ [Hartanto, Pncelet, Popescu and Zoia, Phys. Rev. D, 2022]
- Mixed QCD-electroweak
  - $pp \to Z$ [Buccioni, Caola, Delto, et al, Phys. Lett. B, 2020]
  - $pp \to W$ [Behring, Buccioni, Caola, et al, Phys. Rev. D, 2021]
  - $pp \to l^- l^+$ [Buccioni, Caola, Chawdhry, et al, JHEP 2022]
- NNNLO QCD
  - $pp \to H$ [Anastasiou, Duhr, Dulat, et al, Phys. Rev. Lett., 2015] [Mistlberger, JHEP 2018]
  - $pp \to l^- l^+$ [Duhr, Dulat and Mistlberger, Phys. Rev. Lett., 2020] [Duhr, Dulat and Mistlberger, JHEP, 2020] [Duhr, Mistlberger, JHEP, 2022]

➢ Multiloop scattering amplitudes

- Construct the amplitude

$$\mathcal{A} = \sum c_i I_i$$

  - I_i: scalar Feynman integrals in dimensional regularization

$$I(\vec{\nu}) = \int \prod_{i=1}^{L} \frac{\mathrm{d}^D \ell_i}{\mathrm{i}\pi^{D/2}} \frac{\mathcal{D}_{K+1}^{-\nu_{K+1}} \cdots \mathcal{D}_N^{-\nu_N}}{(\mathcal{D}_1 + \mathrm{i}0)^{\nu_1} \cdots (\mathcal{D}_K + \mathrm{i}0)^{\nu_K}}$$

- Compute the scalar integrals: reduction + computation
  - reduction: express all the scalar integrals in terms of a smaller set of independent integrals (master integrals)

$$I_i = \sum_j b_{ij} M_j$$

  - computation: compute the master integrals as expansions in the dimensional regulator $\epsilon = (4 - D)/2$

$$M_j = \sum_{l=-2L} d_{jk} \epsilon^k$$

Feynman integrals

➢ Integration-by-parts (IBP) identities [Chetyrkin and Tkachov, Nucl. Phys. B, 1981]

$$0 = \int \prod_{i=1}^{L} \frac{\mathrm{d}^D \ell_i}{\mathrm{i}\pi^{D/2}} \frac{\partial}{\partial \ell_j^\mu} \left(\frac{v_k^\mu}{\mathcal{D}_1^{\nu_1} \cdots \mathcal{D}_m^{\nu_m}} \right)$$

- Translation invariance of space-time
- Example: one-loop tadpole

$$0 = \int \frac{\mathrm{d}^D \ell}{\mathrm{i}\pi^{D/2}} \frac{\partial}{\partial \ell^\mu} \left(\frac{\ell^\mu}{(\ell^2 - 1)^\nu} \right)$$

$$\Downarrow$$

$$\int \frac{\mathrm{d}^D \ell}{\mathrm{i}\pi^{D/2}} \frac{1}{(\ell^2 - 1)^{\nu+1}} = \frac{D - 2\nu}{2\nu} \int \frac{\mathrm{d}^D \ell}{\mathrm{i}\pi^{D/2}} \frac{1}{(\ell^2 - 1)^{\nu}}$$

  - $I(\nu)$ can be reduced to $I(1)$
  - $I(1)$: master integral
- Number of master integrals is finite. [Smirnov and Petukhov, Lett. Math. Phys., 2011]

Feynman integrals

- Laporta's algorithm [Laporta, Int. J. Mod. Phys. A, 2000]
  - consider IBP identities with numerical indices, rather than symbolic
  - construct a linear system
  - Gaussian elimination
- Computer programs
  - AIR [Anastasiou and Lazopoulos, JHEP, 2004]
  - FIRE [Smirnov, JHEP, 2008] [Smirnov, Smirnov, Comput. Phys. Commun., 2013] [Smirnov, Comput. Phys. Commun., 2015] [Smirnov and Chuharev, Comput. Phys. Commun., 2020]
  - Reduze [Studerys, Comput. Phys. Commun., 2010] [Manteuffel and Studerus, e-Print: 1201.4330]
  - Kira [Maierhofer, Usovitsch and Uwer, Comput. Phys. Commun., 2018] [Klappert, Lange, Maierhofer and Usovitsch, Comput. Phys. Commun., 2021]
  - LiteRed [Lee, 2012] [Lee, 2014]
  - NeatIBP [Wu, Boehm, Ma, et al, 2305.08783]

Finite field techniques

➢ Finite field arithmetic [Manteuffel and Schabinger, Phys. Lett. B, 2015] [Peraro, JHEP, 2016]

- Motivation: intermediate expression swelling
- Solution: numerical sampling + reconstruction
- Finite fields
  - $Z_p = \{\bar{0}, \bar{1}, \ldots, \overline{p-1}\}$ for a prime number p
  - for any $\bar{a} \neq \bar{0}$, there is a unique element $\bar{b}$, such that $\bar{a}\bar{b} = \bar{1}$
    - $\bar{a}^{-1} := \bar{b}$
  - extension to rational numbers
    - $\frac{a}{b} \to \bar{a}\bar{b}^{-1}$
    - examples: $1/3 \bmod 7 = 5,\ 2/7 \bmod 101 = 58$
- Rational numbers reconstruction: Chinese Remainder Theorem + Wang's algorithm [Wang, 1981]

Finite field techniques

- Univariate
  - polynomials: Newton's interpolation formula

$$f(x) = a_0 + (x - x_0)\Big(a_1 + (x - x_1)\Big(a_2 + (x - x_2)\big(\cdots + (x - x_{R-1})a_R\big)\Big)\Big)$$

  - rational functions: Thiele's interpolation formula

$$f(x) = a_0 + (x - x_0)\left(a_1 + (x - x_1)\left(a_2 + (x - x_2)\left(\cdots + \frac{x - x_{R-1}}{a_R}\right)^{-1}\right)^{-1}\right)^{-1}$$

  - $x_0, x_1, \ldots, x_R$: interpolations
  - $a_0, a_1, \ldots, a_R$: unknown parameters to be determined
  - a_j can be determined from $f(x_j)$ and $a_0, \ldots, a_{j-1}$

Finite field techniques

- Multivariate polynomials: recursively applying Newton's formula
- Multivariate rational functions
  - recursively applying Thiele's formula → too many samples required
  - semi-sparse: univariate rational functions + multivariate polynomials
  - $\vec{x} \to \vec{x}t + \vec{a}$

$$f(\vec{x}) \to h(t, \vec{x}) = \frac{\sum_{r=0}^{R} p_r(\vec{x}) t^r}{1 + \sum_{r=1}^{R'} q_r(\vec{x}) t^r}$$

  - $p_r(\vec{x}), q_r(\vec{x})$: degree-$r$ homogeneous polynomials
  - $(t_0, \vec{x}_0), (t_1, \vec{x}_0), \ldots \to h(t, \vec{x}_0) \to p_r(\vec{x}_0), q_r(\vec{x}_0)$
  - $(t_0, \vec{x}_1), (t_1, \vec{x}_1), \ldots \to h(t, \vec{x}_1) \to p_r(\vec{x}_1), q_r(\vec{x}_1)$
  - …
  - $h(t, \vec{x}) \to f(\vec{x})$
- Programs: FiniteFlow [Peraro, JHEP, 2019] FireFly [Klappert and Lange, Comput. Phys. Commun., 2020]

Summary

Refined IBP systems:

syzygy equations [Gluza, Kajda and Kosower, Phys. Rev. D, 2011] [Larsen and Zhang, Phys. Rev. D, 2016]

block-triangular systems [Guan, **XL**, Ma, Chin.Phys.C, 2020]

More powerful linear solver:

Kira [Maierhofer, Usovitsch and Uwer, Comput. Phys. Commun., 2018]

RATRACER [Magerya, e-Print: 2211.03572]

Better interpolation methods [Klappert and Lange, Comput.Phys.Commun. 2020] [Belitsky, Smirnov, Yakovlev, 2023.02511]

more compact ansatz[Badger, Hansen, Chicherin, et al, JHEP 2021][Laurentis, Page, JHEP 2022][Abreu, Laurentis, Ita, et al, 2305.17056]

the method in this talk

$$\text{time} = \frac{\text{time for a single sample} \times \text{number of samples}}{\text{number of CPUs}}$$

Money

Outline

I. Introduction

II. The method

III. Examples

IV. Summary and outlook

➢ A simple observation

- Traditional strategy: reconstructing functions individually & neglecting common structures
- Example

$$f_i(x) = \left(\frac{1+x}{1-x}\right)^{i-1}, \quad i \in [1, 100]$$

  - approximately 200 samples using Thiele's interpolation formula
  - linear relations

$$(1-x)f_{i+1}(x) - (1+x)f_i(x) = 0, \quad i \in [1, 99]$$

  - ansatz + linear fit → 4 samples

$$(a_i + b_i x)f_{i+1}(x) + (c_i + d_i x)f_i(x) = 0$$

- Linear relations → common structures utilized → number of samples reduced

The method

➢ General description

- Goal: all $n-1$ independent relations among k-variate functions $f_1(\vec{x}), \ldots, f_n(\vec{x})$
- Ansatz

$$Q_1(\vec{x})f_1(\vec{x}) + \cdots + Q_n(\vec{x})f_n(\vec{x}) = 0$$

  - $Q_i(\vec{x})$: polynomial of $\vec{x}$ $\Rightarrow$ specifying monomials $x_1^{\alpha_1} \cdots x_k^{\alpha_k}$
- Definition 1:

$$P(\{x_{i_1}, \ldots, x_{i_l}\}, m) := \{x_{i_1}^{\alpha_1} \cdots x_{i_l}^{\alpha_l} \,|\, \sum \alpha_i \le m\}$$

  - $P(\{x_1\},2) = \{1, x_1, x_1^2\}$; $P(\{x_2, x_3\},1) = \{1, x_2, x_3\}$
- Definition 2:

$$P_1 \times P_2 := \{p_1 p_2 \,|\, p_1 \in P_1, p_2 \in P_2\}$$

  - $P(\{x_1\},2) \times P(\{x_2,x_3\},1) = \{1, x_1, x_1^2, x_2, x_1x_2, x_1^2x_2, x_3, x_1x_3, x_1^2x_3\}$
  - $P\left(\{x_{i_1}, \ldots, x_{i_l}\}, m_1\right) \times P\left(\{x_{i_1}, \ldots, x_{i_l}\}, m_2\right) = P\left(\{x_{i_1}, \ldots, x_{i_l}\}, m_1 + m_2\right)$

The method

- Division of variables $\vec{x}$ into r subsets $S_1, \ldots, S_r$ (comments in next slide)
  - $S_1 = \{x_1\}, S_2 = \{x_2, x_3\}$
- For a specific $Q_i(\vec{x})$, given integers $\vec{z} = \{z_1, \ldots, z_r\}$

$$M(\vec{z}) := P(S_1, z_1) \times \cdots \times P(S_r, z_r)$$

  - $M(2,1) = P(S_1, 2) \times P(S_2, 1) = \{1, x_1, x_1^2, x_2, x_1x_2, x_1^2x_2, x_3, x_1x_3, x_1^2x_3\}$
  - $M(\vec{z}_1) \times M(\vec{z}_2) = M(\vec{z}_1 + \vec{z}_2)$
- In practice: use the same $\vec{z}$ for all the Q's
- Algorithm from [Guan, **XL**, Ma, Chin.Phys.C, 2020]
  - 1. start with $\sum z_i = 0$;
  - 2. for each solution of $\vec{z}$, make the ansatz and fit the unknowns
  - 3. test the number of independent relations: if sufficient, terminate; otherwise increase $\sum z_i$ by 1 and go back to step 2.

The method

- Redundant relations: relations from $\vec{z}$ will be redundantly obtained from $\vec{z}'$ if $z_i \leq z_i'$ for any i
  - $z_1 = 1, z_2 = 0$: $(x_1 - 1)f_1 + (3x_1 + 2)f_2 + \cdots = 0$
  - $z_1 = 1, z_2 = 1$: $(x_1x_2 - x_2)f_1 + (3x_1x_2 + 2x_2)f_2 + \cdots = 0$
- Solution: eliminate "solvable" monomials
  - $z_1 = 1, z_2 = 0$: x_1 of f_1 is "solved", $x_1f_1 = f_1 + (-3x_1 - 2)f_2 + \cdots$
  - $z_1 = 1, z_2 = 1$: x_1 and x_1x_2 of f_1 should be "solvable"
  - in general, eliminate $\Lambda_i(\vec{z}) \times M(\vec{z}' - \vec{z})$ for all possible $\vec{z}$

- Comment on the variables division
  - any division works: $S_1 = \{x_1, \ldots, x_k\}$; $S_1 = \{x_1\}, \ldots, S_k = \{x_k\}$; ...
  - but the number of samples can vary significantly
  - do some tests to gain insights

Summary

- Summary
  - build a generator of the numerical samples for the target functions
    - e.g., IBP system + linear solver
  - find the system of all the independent linear relations over a finite field
    - various ansatz of $Q_1(\vec{x})f_1(\vec{x}) + \cdots + Q_n(\vec{x})f_n(\vec{x}) = 0$
    - linear fit = samples ($N_{\text{sample}} \sim N_{\text{unknown}}$) + dense system ($N_{\text{sample}} \times N_{\text{unknown}}$)
  - solve the linear system to obtain explicit solutions
    - traditional rational functions reconstruction strategy
    - additional finite fields (knowledge from the first finite field) + rational numbers reconstruction

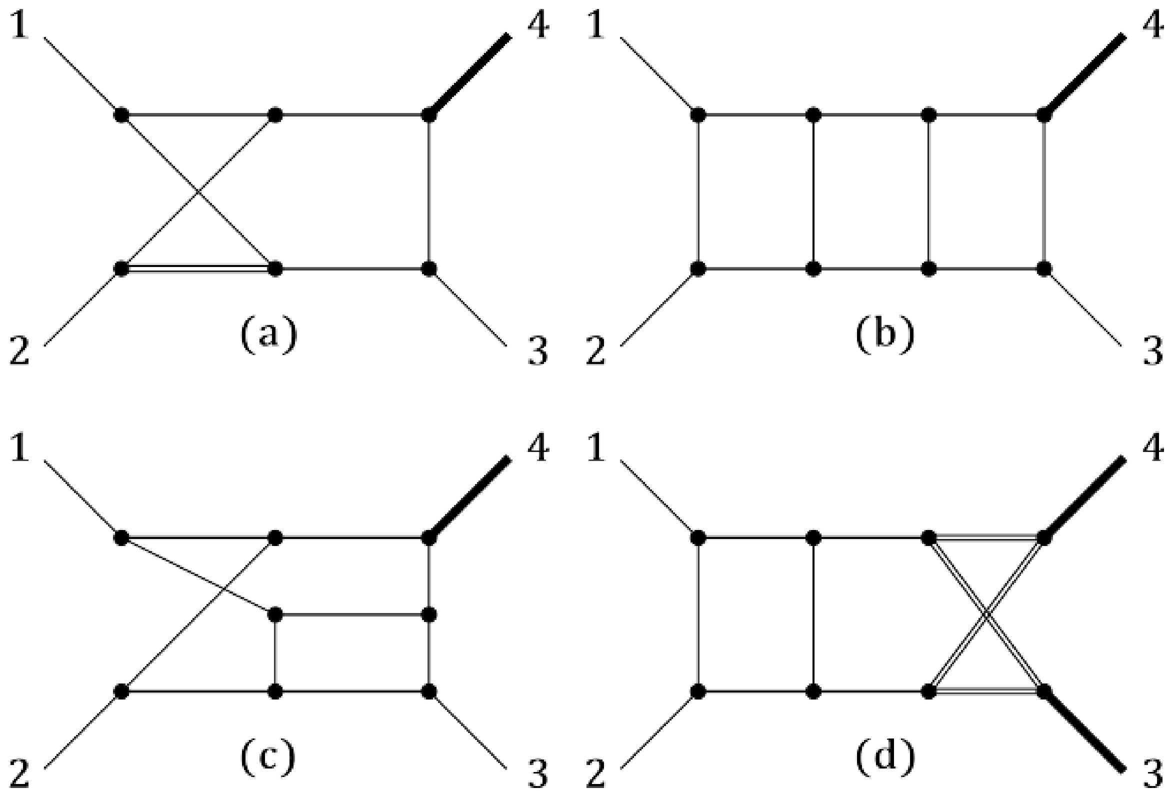
Outline

Examples

- Reduction coefficients of Feynman integrals or amplitudes

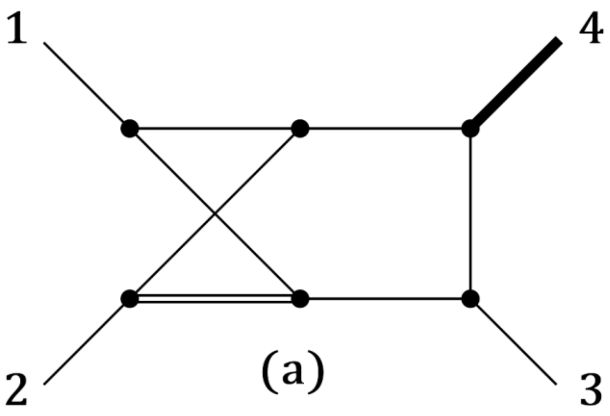
$$\mathcal{A} = f_1\mathcal{M}_1 + \cdots + f_n\mathcal{M}_n$$

  - a common set of denominators reflecting the singularities
  - auxiliary function $f_{n+1} = 1$



Examples

- Topology (a): two-loop amplitude of the mixed QCD-electroweak correction to $pp \to Z + j$ [Bargiela, Caola, Chawdhry, **XL**, to appear]
- Setup
  - $m_Z^2 = 1, m_W^2 = 7/9$
  - remaining: $\{\epsilon, s_{12}, s_{13}\}$
  - 56 master integrals $\Rightarrow$ 56 rational functions
  - LiteRed + FiniteFlow



- Details
  - $S_1 = \{\epsilon\}, S_2 = \{s_{12}, s_{13}\}$
  - $z_1 + z_2 = 6$
  - 1+2 finite fields with 64-bit prime numbers
  - samples: $18326 \times 3 \to 2199 + 1561 \times 2 \Rightarrow$ a factor of 10.3
  - computational cost: 4.6h → 0.44h + 0.03h ⇒ a factor of 9.8

Examples

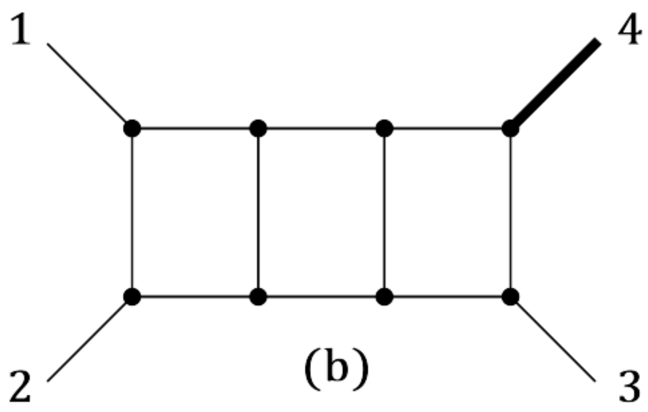
$\vec{z}$	$N_{unknown}$	N_{sample}	$N_{relation}$	$N2_{sample}$
{0, 0}	57	58	0	
{1,0}	114	115	0	
{0,1}	171	172	1	4
{2,0}	171	172	0	
{1,1}	340	341	0	
{0,2}	339	340	1	9
{3,0}	228	229	0	
…				
{2,2}	1014	1015	1	31
…				
{2,3}	1680	1681	20	1394
…				
{3,3}	2198	2199	33	1561
summary		2199	56	1561

Examples

- Number of samples
  - explicit reduction coefficients: total degree 40 for numerator and 39 for denominator
  - linear relations: total degree 6
  - $N_1 = 2199, N_2 = 1561 \ll N_0 = 18326$
- Performance of the systems
  - IBP system (34336 equations): 0.3s per phase-space point
  - Our system (56 equations): 7.5×10^{-4}s per phase-space point
  - 400 times faster $\Rightarrow t_{\text{sol}} = 0.03\text{h} \ll t_{\text{sam}} = 0.44\text{h}$

Examples

- Topology (b): an integral with rank-6 numerator
- Setup
  - $p_4^2 = 1$
  - remaining: $\{\epsilon, s_{12}, s_{13}\}$
  - 83 master integrals $\Rightarrow$ 83 rational functions
  - NeatIBP + FiniteFlow
- Details
  - $S_1 = \{\epsilon\}, S_2 = \{s_{12}, s_{13}\}$
  - $z_1 + z_2 = 8$
  - 1+2 finite fields
  - samples: $48574 \times 3 \rightarrow 6010 + 4599 \times 2 \Rightarrow$ a factor of 9.6
  - computational cost: 78.5h → 8.03h + 0.12h $\Rightarrow$ a factor of 9.6

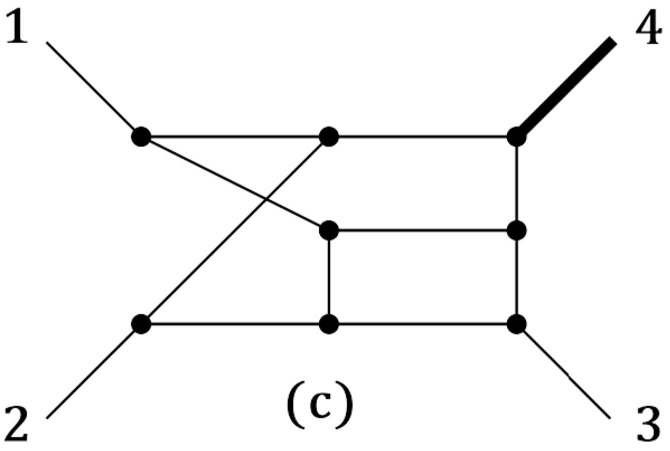


Examples

- Number of samples
  - explicit reduction coefficients: total degree 56 for numerator and 55 for denominator
  - linear relations: total degree 8
  - $N_1 = 6010, N_2 = 4599 \ll N_0 = 48574$
- Performance of the systems
  - IBP system (200074 equations): 1.9s per phase-space point
  - Our system (83 equations): 2.4×10^{-3}s per phase-space point
  - 792 times faster $\Rightarrow\ t_{\text{sol}} = 0.12\text{h} \ll t_{\text{sam}} = 8.03\text{h}$

Examples

- Topology (c): differential equations of master integrals w.r.t. Mandelstam variables [Kotikov, Phys. Lett. B, 1991] [Henn, Phys. Rev. Lett., 2013] : $\frac{\partial}{\partial s_{12}}\vec{M}, \frac{\partial}{\partial s_{13}}\vec{M}$
- Setup
  - $p_4^2 = 1$
  - remaining: $\{\epsilon, s_{12}, s_{13}\}$
  - 280 master integrals
  - LiteRed + FiniteFlow
- Details
  - $S_1 = \{\epsilon\}, S_2 = \{s_{12}, s_{13}\}$
  - $z_1 + z_2 \leq 8$
  - 1+5 finite fields
  - samples: $391937 \times 6 \rightarrow 9612 + 6810 \times 5 \Rightarrow$ a factor of 54
  - computational cost: 450728h* → 8369h + 180h ⇒ a factor of 53

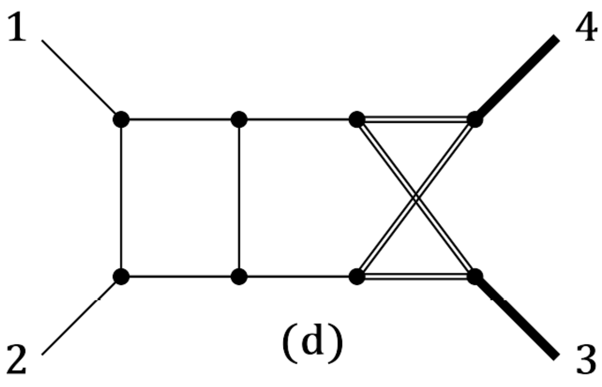


Examples

- Number of samples
  - explicit reduction coefficients: total degree 102 for numerator and 103 for denominator
  - linear relations: total degree 8
  - $N_1 = 9612, N_2 = 6810 \ll N_0 = 391937$
- Performance of the systems
  - IBP system (3461628 equations): 690s per phase-space point
  - Our representative system (280 equations): 0.013s per phase-space point
  - 53077 times faster $\Rightarrow\ t_{\mathrm{sol}} = 180\mathrm{h} \ll t_{\mathrm{sam}} = 8369\mathrm{h}$

Examples

- Topology (d): differential equations with respect to internal squared masses
- Extensively involved in the auxiliary mass flow method [**XL**, Ma, Wang, Phys.Lett.B., 2018] [XL, Ma, Comput.Phys.Commun., 2023]



- Setup
  - $p_3^2 = p_4^2 = 1, s_{12} = 10, s_{13} = -22/9$
  - remaining: $\{\epsilon, m^2\}$
  - 336 master integrals
  - LiteRed + FiniteFlow
- Details
  - $S_1 = \{\epsilon\}, S_2 = \{m^2\}$
  - $z_1 + z_2 \leq 5$
  - 1+32 finite fields
  - samples: $14362 \times 33 \rightarrow 1414 + 1248 \times 32 \Rightarrow$ a factor of 11.5
  - computational cost: 3230h → 281h + 59h ⇒ a factor of 9.5

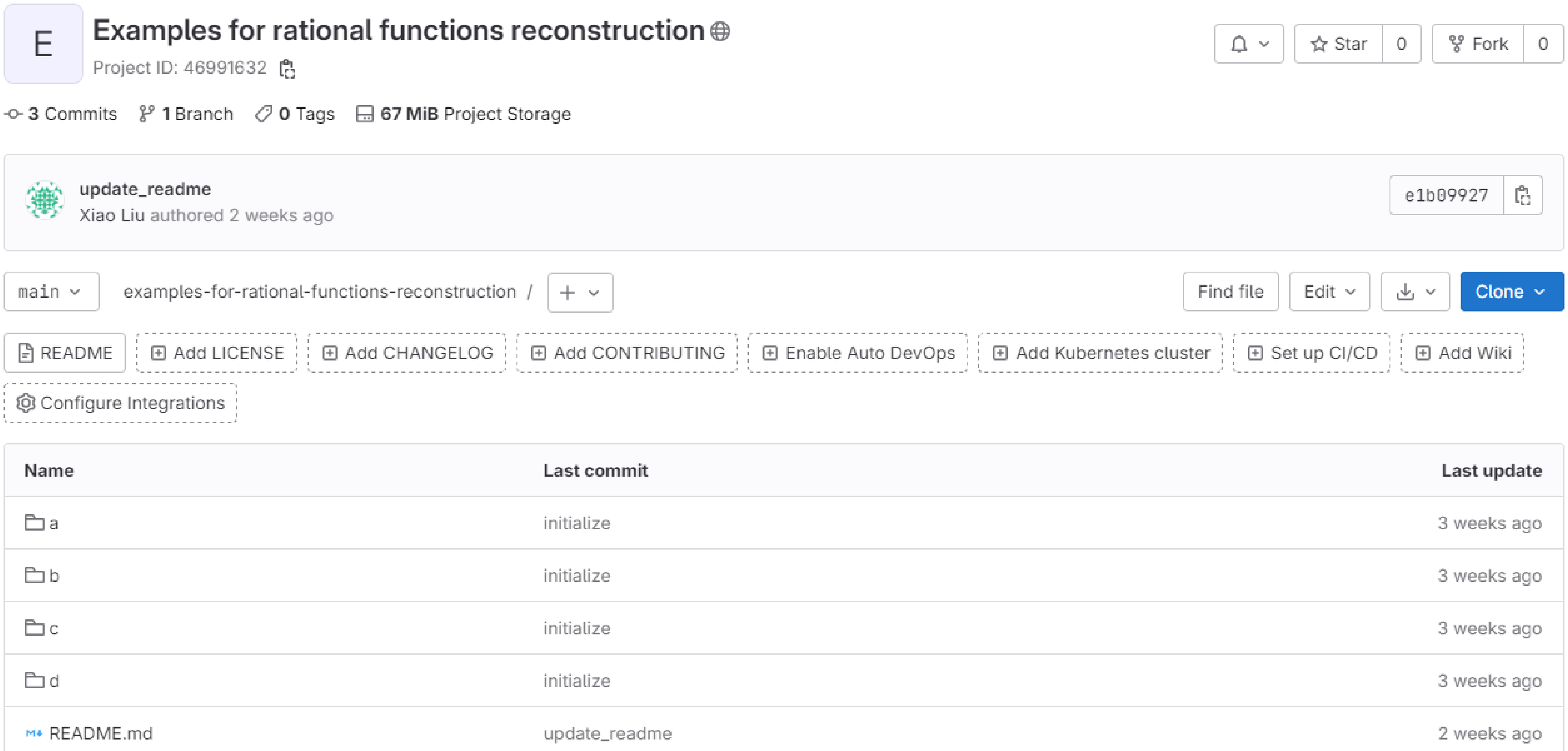
Examples

- Number of samples
  - explicit reduction coefficients: total degree 144 for numerator and 143 for denominator
  - linear relations: total degree 5
  - $N_1 = 1414, N_2 = 1248 \ll N_0 = 14362$
- Performance of the systems
  - IBP system (625070 equations): 24.5s per phase-space point
  - Our representative system (336 equations): 0.019s per phase-space point
  - 1289 times faster, however number of complicated systems increases $\Rightarrow$

$$\frac{t_{\rm sol}=59{\rm h}}{t_{\rm sam}=281h} \sim \frac{1}{5}$$

Examples

- https://gitlab.com/xiaoliu222222/examples-for-rational-functions-reconstruction
  - explicit reduction coefficients & the linear system they satisfy



Outline

I. Introduction

II. The method

III. Examples

IV. Summary and outlook

Summary and Outlook

- A new method for the reconstruction of rational functions is proposed, which works by exploiting all the independent linear relations among the target functions.
- Better scaling behavior
  - improvement factor: univariate $\leq$ 2-variate $\leq$ 3-variate
- The current form of the method is not so good to solve problems with more than 3 variables $\rightarrow$ linear fit becomes dominant and sometimes prohibitive
  - refined ansatz for the relations: sparse or semi-sparse?
  - refined choice of auxiliary functions, rather than a naïve $f_{n+1}(x) = 1$
- $t_{\text{sol}} \ll t_{\text{sam}}$ in most cases $\Rightarrow$ improvements in the generators
  - for cases where $t_{\text{sol}} \geq t_{\text{sam}}$, refined approach to grouping the functions

Thank you!