A new method for the reconstruction of rational functions

Xiao Liu

University of Oxford

Based on e-print: 2306.12262

MathemAmplitudes 2023: QFT at the Computational Frontier
26 September 2023, Padova

➢ Reduction of Feynman integrals

- Integration-by-parts system [Chetyrkin and Tkachov, 1981] [Laporta, 2000]

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

  - AIR, FIRE, Reduze, Kira, LiteRed, NeatIBP, Blade talks by Yan-Qing Ma, Johann Usovitsch, Mao Zeng, Yang Zhang
- Multivariate partial fraction
  - MultivariateApart, Pfd-parallel talks by Andreas V Manteuffel and Yang Zhang
- Finite field reconstruction [Manteuffel and Schabinger, 2015] [Peraro, 2016]
  - FiniteFlow [Peraro, 2019], FireFly [Klappert and Lange, 2020]

Rational Functions

Refined IBP systems:
Syzygy equations [Gluza, Kajda and Kosower, 2011] [Larsen and Zhang, 2016]
Block-triangular systems [Guan, **XL**, Ma, 2020]

More powerful linear solver:
Kira [Maierhofer, Usovitsch and Uwer, 2018]
RATRACER [Magerya, 2211.03572]

Better interpolation methods [Klappert and Lange, 2020] [Belitsky, Smirnov, Yakovlev, 2023.02511]

More compact ansatz [Badger, Hansen, Chicherin, et al, 2021][Laurentis, Page, 2022][Abreu, Laurentis, Ita, et al, 2305.17056] and talk by Giuseppe De Laurentis

↑ ↑

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

The method

➢ 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

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 $\vec{x}$
- Generalized algorithm from [Guan, **XL**, Ma, 2020]

  1. For a given ansatz, fit the unknowns to obtain the linear relations

  2. Test the number of independent relations: if sufficient, terminate; otherwise make a new ansatz and go back to step 1.
- Example (2-variate):
  - $a_1 f_1 + \cdots + a_n f_n = 0$
  - $(a_1 + b_1 x_1 + c_1 x_2) f_1 + \cdots + (a_n + b_n x_1 + c_n x_2) f_n = 0$
  - $(a_1 + b_1 x_1 + c_1 x_2 + d_1 x_1^2 + e_1 x_1 x_2 + g_1 x_2^2) f_1 + \cdots = 0$

The method

- More carefully
  - Divide the variables into subsets and assign a monomial degree for each set
  - $S_1 = \{x_1\}, S_2 = \{x_2\}$
    - $\vec{z} = \{0,0\} \rightarrow a_1 f_1 + \cdots + a_n f_n = 0$
    - $\vec{z} = \{1,0\} \rightarrow (a_1 + b_1 x_1) f_1 + \cdots + (a_n + b_n x_1) f_n = 0$
    - $\vec{z} = \{0,1\} \rightarrow (a_1 + b_1 x_2) f_1 + \cdots + (a_n + b_n x_2) f_n = 0$
    - $\vec{z} = \{1,1\} \rightarrow (a_1 + b_1 x_1 + c_1 x_2 + d_1 x_1 x_2) f_1 + \cdots = 0$
  - Naïve division: $S = \{x_1, x_2\}$
- Change values of $\vec{z} \rightarrow$ various ansatz made
  - $\sum z_i = 0 \rightarrow \{0, \ldots, 0\}$
  - $\sum z_i = 1 \rightarrow \{1, 0, \ldots, 0\}, \{0, 1, 0, \ldots, 0\}, \ldots, \{0, \ldots, 0, 1\}$
  - $\sum z_i = 2 \rightarrow \{2, 0, \ldots, 0\}, \{1, 1, 0, \ldots, 0\}, \{0, 2, 0, \ldots, 0\}, \ldots, \{0, \ldots, 0, 2\}$
  - …

The method

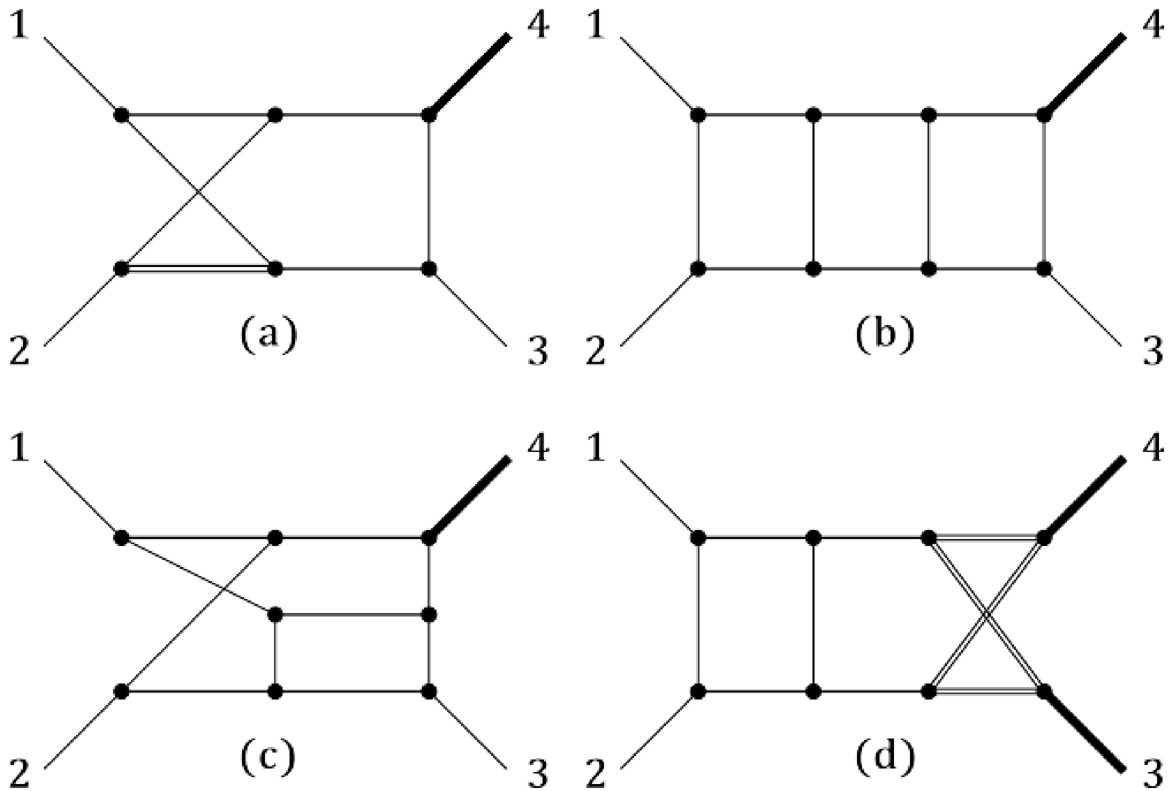
- Summary
  - Build a generator of the numerical samples for the target functions
    - e.g., IBP system + linear solver
  - Find the system of all the independent linear relations over a finite field
    - Various ansatz of $Q_1(\vec{x})f_1(\vec{x}) + \cdots + Q_n(\vec{x})f_n(\vec{x}) = 0$
    - Linear fit = samples ($N_{\text{sample}} \sim N_{\text{unknown}}$) + dense solve ($N_{\text{sample}} \times N_{\text{unknown}}$)
  - Solve the linear system to obtain explicit solutions
    - Traditional rational functions reconstruction strategy
    - Additional finite fields + rational numbers reconstruction
- Computational cost: typically dominated by samples generation during linear fit

Examples

- Reduction coefficients of Feynman integrals or amplitudes

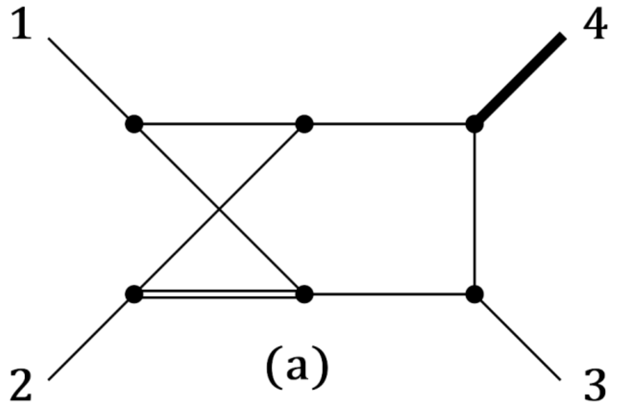
$$\mathcal{A} = f_1 \mathcal{M}_1 + \cdots + f_n \mathcal{M}_n$$

  - 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



Examples

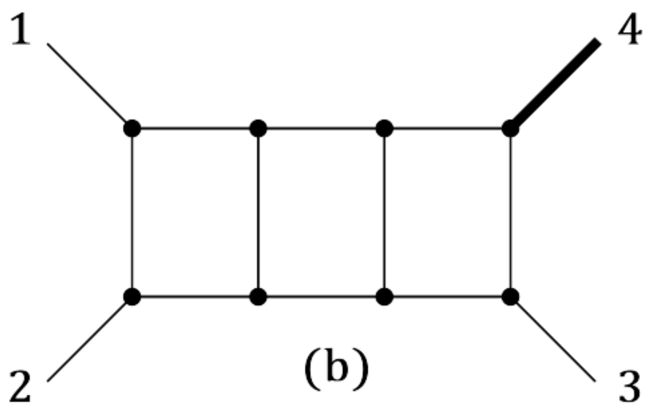
$\vec{z}$	N_{unknown}	N_{sample}	N_{relation}	$N2_{\text{sample}}$
{0, 0}	57	58	0	-
{1,0}	114	115	0	-
{0,1}	171	172	1	4
{2,0}	171	172	0	-
{1,1}	340	341	0	-
{0,2}	339	340	1	9
{3,0}	228	229	0	-
…				
{2,2}	1014	1015	1	31
…				
{2,3}	1680	1681	20	1394
…				
{3,3}	2198	2199	33	1561
summary		2199	56	1561

Examples

- Number of samples
  - 18326×3 given by FiniteFlow
  - $2199 + 1561 \times 2$ by the new method → a factor of 10.3
  - Explicit reduction coefficients: total degree 40 for numerator and 39 for denominator
  - Linear relations: total degree 6
- Computational cost
  - 4.6 CPU hours by FiniteFlow
  - 0.44 CPU hour (samples generation) + 0.03 CPU hour (explicit solution) by the new method → a factor of 9.8
  - 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

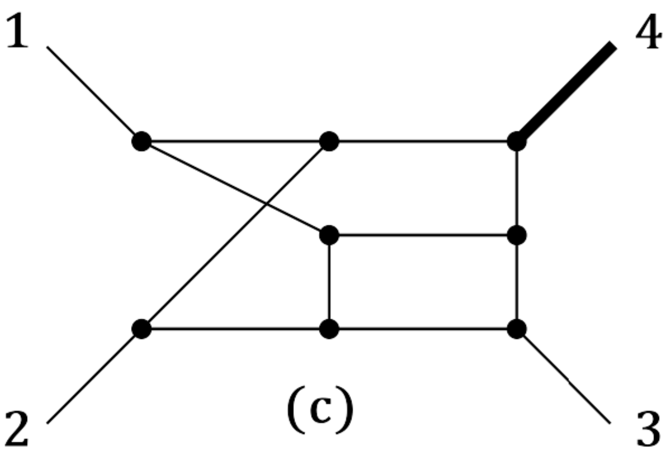
Examples

- 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
  - Number of samples: $48574 \times 3 \rightarrow 6010 + 4599 \times 2 \Rightarrow$ a factor of 9.6
  - Computational cost: $78.5 \rightarrow 8.03 + 0.12$ CPU hours $\Rightarrow$ a factor of 9.6



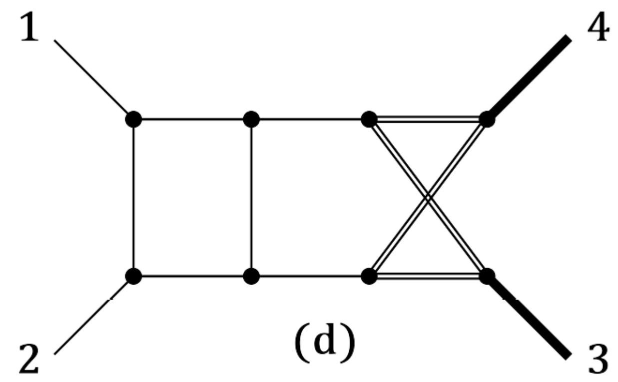
Examples

- Topology (c): differential equations of master integrals w.r.t. Mandelstam variables [Kotikov, 1991] [Henn, 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
  - Number of samples: $391937 \times 6 \rightarrow 9612 + 6810 \times 5 \Rightarrow$ a factor of 54
  - Computational cost: 450728* $\rightarrow 8369 + 180$ CPU hours $\Rightarrow$ a factor of 53



Examples

- Topology (d): differential equations with respect to internal squared masses
- Extensively involved in the auxiliary mass flow method [**XL**, Ma, Wang, 2018]
- Setup
  - $p_3^2 = p_4^2 = 1, s_{12} = 10, s_{13} = -22/9$
  - Remaining: $\{\epsilon, m^2\}$
  - 336 master integrals
  - LiteRed + FiniteFlow



(d)

- Details
  - $S_1 = \{\epsilon\}, S_2 = \{m^2\}$
  - $z_1 + z_2 \leq 5$
  - 1+32 finite fields
  - Number of samples: $14362 \times 33 \rightarrow 1414 + 1248 \times 32 \Rightarrow$ a factor of 11.5
  - Computational cost: $3230 \rightarrow 281 + 59$ CPU hours $\Rightarrow$ a factor of 9.5

Examples

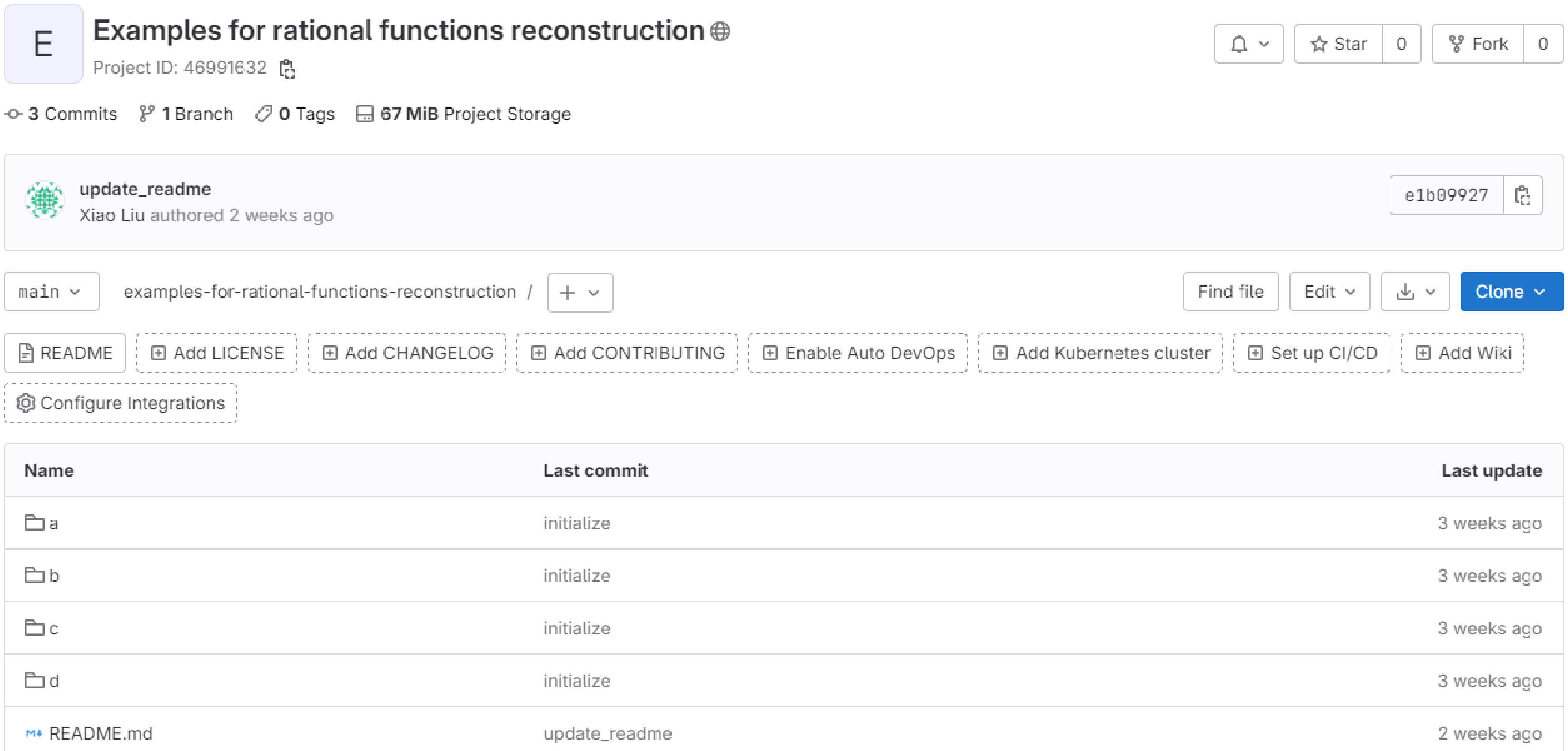
- More details
  - Topology (c) and (d): a representative integral

Topology	d_{Num}	d_{Den}	d_{Rel}	N_{IBP}	N_{Rel}	t_{IBP}/s	t_{Rel}/s
(a)	40	39	6	34336	56	0.3	0.00075
(b)	56	55	8	200074	83	1.9	0.0024
(c)	102	103	8	3461628	280	690	0.013
(d)	144	143	5	625070	336	24.5	0.019

  - $d_{\text{Rel}} \ll d_{\text{Num}}, d_{\text{Den}}$
  - $t_{\text{Rel}} \ll t_{\text{IBP}}$, $t_{\text{IBP}}/t_{\text{Rel}}$ for each topology: 400, 792, 53077, 1289

Examples

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



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
- Time for solving negligible in most cases $\Rightarrow$ improvements in the generators of rational functions also yield benefits
- 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 $f_{n+1}(x) = 1$

Thank you!