New tools towards fast evaluation of Feynman integrals

Xiao Liu

University of Oxford

Mainly based on 1711.09572, 1912.09294, 2201.11669 and work in progress

In collaboration with Xin Guan, Yan-Qing Ma, Chen-Yu Wang and Wen-hao Wu

RADCOR 2023 – 16th International Symposium on Radiative Corrections: Applications of Quantum Field Theory to Phenomenology

28 May 2023 ~ 2 June 2023, Crieff, Scotland

Outline

Feynman integrals

➢ Feynman integrals in dimensional regularization

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

- interesting in physics
  - key objects for fixed order computation
  - Standard Model & New Physics
- challenging in mathematics
  - algebraic complexity (integrals reduction)
  - function space (master integrals computation)
- state of the art
  - massive two-loop five-point integrals
  - massive three-loop four-point integrals

➢ Methods and tools

- Reduction:
  - integration-by-parts reduction [Chetyrkin and Tkachov, Nucl. Phys. B, 1981] [Laporta, Int. J. Mod. Phys. A, 2000]
    - 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]
  - syzygy equations [Gluza, Kajda and Kosower, Phys. Rev. D, 2011] [Larsen and Zhang, Phys. Rev. D, 2016]
    - NeatIBP [Wu, Boehm, Ma et al, e-Print: 2305.08783]
  - finite field reconstruction [Manteuffel and Schabinger, Phys. Lett. B, 2015] [Peraro, JHEP, 2016]
    - FiniteFlow [Peraro, JHEP, 2019] FireFly [Klappert and Lange, Comput. Phys. Commun., 2020] RATRACER [Magerya, e-Print: 2211.03572]
  - block-triangular system and Blade

Feynman integrals

- Computation:
  - canonical differential equations [Kotikov, Phys. Lett. B, 1991] [Henn, Phys. Rev. Lett., 2013]
    - CANONICA [Meyer, 2018] epsilon [Prausa, 2017] Fuchsia [Gituliar and Magerya, 2017] Libra [Lee, 2020] DLogBasis [Henn, Mistlberger, Smirnov and Wasser, JHEP 2020]
  - numerical differential equations [Caffo, Czyz, Gunia and Remiddi, Comput. Phys. Commun., 2009] [Czakon, Phys. Lett. B, 2008]
    - DiffExp [Hidding, Comput.Phys.Commun., 2021] SeaSyde [Armadillo, Bonciani, Devoto, Rana and Vicini, Comput.Phys.Commun., 2023]
  - sector decomposition [Binoth and Heinrich, Nucl. Phys. B, 2000]
    - FIESTA [Smirnov and Tentyukov, 2009] [Smirnov, Smirnov and Tentyukov, 2011] [Smirnov, 2014] [Smirnov, 2015] SecDec [Carter and Heinrich, 2011] [Borowka, Carter and Heinrich, 2013] [Borowka, Heinrich, Jones, et al, 2015] pySecDec [Borowka, Heinrich, Jahn, et al, 2018]
  - tropical geometry [Borinsky, 2008.12310] [Borinsky, Munch and Tellander, 2302.08955]
    - feyntrop [Borinsky, Munch and Tellander, 2302.08955]
  - auxiliary mass flow and AMFlow

➢ Other good methods

- Intersection theory [Mastralia and Mizera, JHEP, 2019] [Frellesvig, Gasparotto, Laporta, et al, JHEP 2019]
- Partial fraction decomposition [Boehm, Wittmann, Wu, et al, JHEP, 2020]
- Method of region [Beneke and Smirnov, Nucl. Phys. B, 1998] [Smirnov, Phys. Lett. B, 1999]
- Mellin-Barnes representation [Boos and Davydychev, Theor.Math.Phys., 1991][Smirnov, Phys. Lett. B, 1999]
- Dimensional recurrence and Analyticity [Tarasov, Phys. Rev. D, 1996] [Lee, Nucl. Phys. B, 2010]
- Positivity constraints [Zeng, 2303.15624]
- …

Outline

I. Introduction

II. Block-triangular system and Blade

III. Auxiliary mass flow and AMFlow

IV. Applications

V. Summary and outlook

Block-triangular system

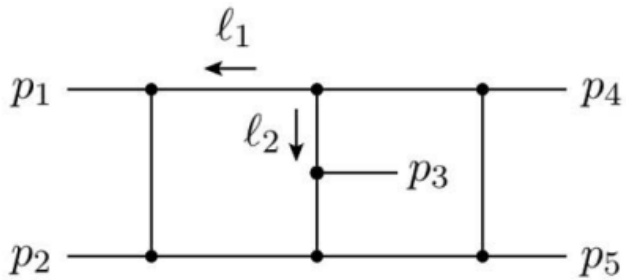
➢ Motivation

- IBP system
  - large: many equations and integrals, typically $10^5\sim$
  - coupled: the systems usually become denser during the solutions
  - time- and resource-consuming
- How to improve?
  - small: as few equations as possible
  - decoupled or semi-decoupled: the system should never become denser
- A possible solution: block-triangular system
- A simple example:
  - 2 equations reducing I_6, I_7 to I_8
  - 2 equations reducing I_4, I_5 to I_8
  - 3 equations reducing I_1, I_2, I_3 to I_8

$$
\begin{aligned}
2I_1 + 3sI_2 + tI_3 + 5I_4 + DI_5 - I_6 + DI_7 + sI_8 &= 0 \\
2sI_1 + tI_2 + DI_3 + stI_4 + I_5 - 6I_6 + sI_7 + DI_8 &= 0 \\
2tI_1 + DI_2 - I_3 - 2sI_4 - 8I_5 + tI_6 - I_7 + sI_8 &= 0 \\
I_4 + I_5 - sI_6 + tI_7 - DI_8 &= 0 \\
I_4 - I_5 - DI_6 + stI_7 + D^2I_8 &= 0 \\
sI_6 + tI_7 - I_8 &= 0 \\
stI_6 - I_7 + I_8 &= 0
\end{aligned}
$$

Relations among Feynman integrals

➢ Feynman integrals are vectors



$$I(\vec{\nu}) = \int \prod_{i=1}^{2} \frac{\mathrm{d}^D \ell_i}{\mathrm{i}\pi^{D/2}} \frac{\mathcal{D}_9^{-\nu_9} \mathcal{D}_{10}^{-\nu_{10}} \mathcal{D}_{11}^{-\nu_{11}}}{\mathcal{D}_1^{\nu_1} \mathcal{D}_2^{\nu_2} \mathcal{D}_3^{\nu_3} \mathcal{D}_4^{\nu_4} \mathcal{D}_5^{\nu_5} \mathcal{D}_6^{\nu_6} \mathcal{D}_7^{\nu_7} \mathcal{D}_8^{\nu_8}}$$

- $\vec{s} = \{s_{12}, s_{23}, s_{34}, s_{45}, s_{15}\}$
- 108 master integrals: $\{M_1, \dots, M_{108}\}$
- each integral can be written as a linear combination
  - $I = c_1 M_1 + \cdots + c_{108} M_{108} \Leftrightarrow (c_1 \ \dots \ c_{108})^{\mathrm{T}}$
  - $c_1, \dots, c_{108}$ are numerically computable from IBP system over finite fields

Relations among Feynman integrals

➢ Relations among $G \equiv \{I_1, I_2, \dots, I_n\}$

$$Q_1 I_1 + Q_2 I_2 + \cdots Q_n I_n = 0$$

- Q_i: polynomials of $\vec{s}$ and D
  - ansatz: $Q_1 = a + bD, \quad Q_2 = cs_{23} + ds_{34}, \; Q_3 = es_{45} + fs_{15}, \; \cdots$
  - $a, b, c, d, e, f, \dots$: unknown rational numbers to be determined (The total number of unknowns is finite!)
- For a given ansatz (search algorithm)
  - compute the numerical reduction coefficients of $I_1, \dots, I_n$ at $D = D_0$ and $\vec{s} = \vec{s}_0$
  - this can provide at most 108 linear constraints for the unknowns

$$a \begin{pmatrix} c_{1,1} \\ \vdots \\ c_{1,108} \end{pmatrix} + bD_0 \begin{pmatrix} c_{1,1} \\ \vdots \\ c_{1,108} \end{pmatrix} + cs_{23,0} \begin{pmatrix} c_{2,1} \\ \vdots \\ c_{2,108} \end{pmatrix} + \cdots = 0$$

  - repeat the above process until there are enough constraints

Reduction

➢ Reducing $G_1 = \{I_1, \dots, I_m\}$ to $G_2 = \{I_{m+1}, \dots, I_n\}$

1. Set $G = G_1 \cup G_2 = \{I_1, \dots, I_n\}$;
2. for a given ansatz of $Q_1, \dots, Q_n$, use the search algorithm to determine all the relations;
3. if the obtained relations are sufficient to reduce G_1, stop; else make another ansatz and go back to step 2.

➢ Making ansatz

- from simple to complicated, for instance:
  - degree-0: $Q_1 = a_1,\ Q_2 = a_2,\ Q_3 = a_3,\ \cdots$
  - degree-1: $Q_1 = a_1 + b_1 D + c_1 s_{23} + d_1 s_{34} + e_1 s_{45} + f_1 s_{15}, Q_2 = a_2 + b_2 D + c_2 s_{23} + d_2 s_{34} + e_2 s_{45} + f_2 s_{15}, Q_3 = \cdots, \cdots$
  - degree-2: $Q_1 = a_1 + b_1 D + c_1 s_{23} + d_1 s_{34} + e_1 s_{45} + f_1 s_{15} + g_1 D^2 + h_1 D s_{23} + i_1 D s_{34} + \cdots$

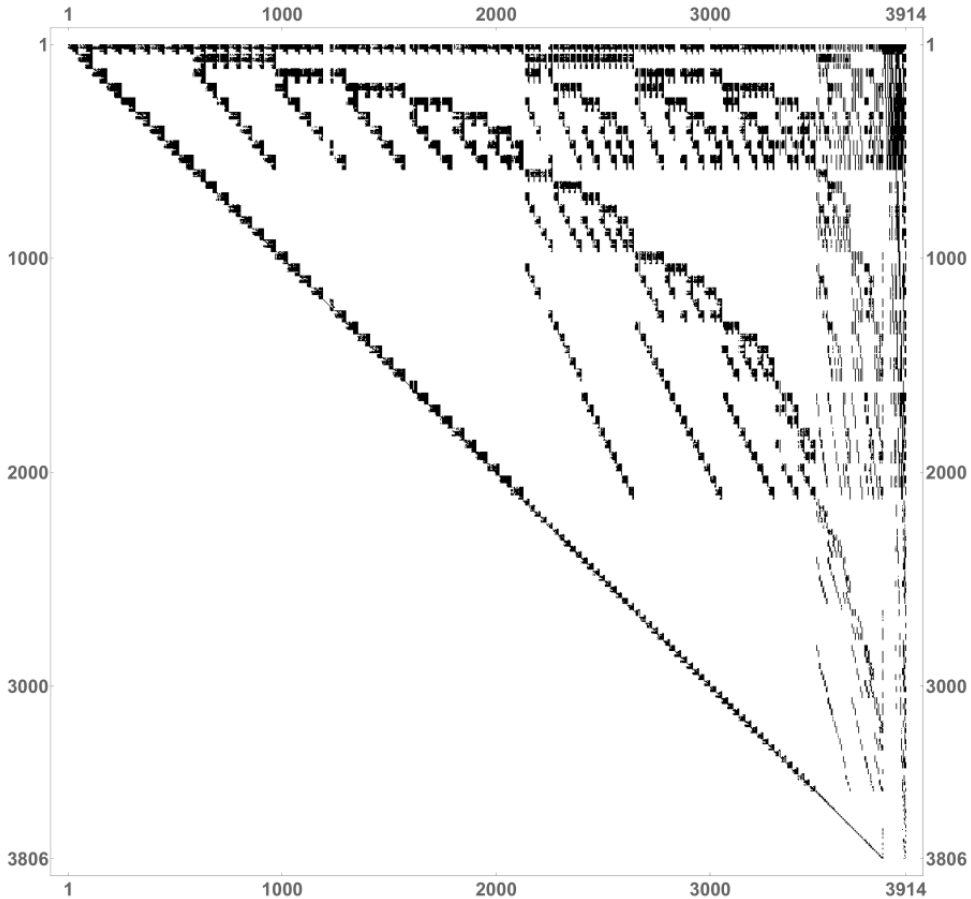
Block-triangular system

➢ Constructing G_1 and G_2 for each block

- Basic construction
  - G_1: target integrals from some sector
  - G_2: integrals in its subsectors
  - if G_1 does not contain dotted integrals, then G_2 as well
- Example (top-sector):
  - G_1: top-sector integrals with up to degree-5 numerators
    - $\{I(1,1,1,1,1,1,1,1,0,0,0), I(1,1,1,1,1,1,1,1,-1,0,0), I(1,1,1,1,1,1,1,1,0,-1,0), \ldots, I(1,1,1,1,1,1,1,1,0,-5,0), I(1,1,1,1,1,1,1,1,0,0,-5)\}$
  - G_2: integrals in subsectors
    - $\{I(0,1,1,1,1,1,1,1,0,0,0), \ldots, I(0,1,1,1,1,1,1,1,0,0,-4), I(1,0,1,1,1,1,1,1,0,0,0), \ldots, I(1,0,1,1,1,1,1,1,0,0,-4), \ldots, I(0,0,1,1,1,1,1,1,0,0,0), \ldots\}$

Results

- IBP system
  - $\sim 3 \times 10^5$ equations
  - numerical sampling over finite fields: 7s per phase-space point
- Block-triangular system
  - 3806 equations reducing 3914 integrals to 108 master integrals
  - 0.17s per phase-space point
  - improvement: $7/0.17 \approx 40$



Blade (in progress)

B

Blade

Project ID: 42197761

Star 7 Fork 0

118 Commits **1** Branch **0** Tags **3.1 MB** Project Storage

Block-triangular form improved Feynman integral decomposition .

- BLock-triAngular form improved Feynman integral DEcomposition
- Available at https://gitlab.com/multiloop-pku/blade
- Installation
  - $ chmod +x auto_install
  - $ ./auto_install

Blade (in progress)

- Basic usage

```
family = dbox;
dimension = 4-2*eps;
loop = {l1,l2};
leg = {p1,p2,p3,p4};
conservation = {p4->-p1-p2-p3};
replacement = {p1^2 -> 0, p2^2 -> 0, p3^2 -> msq, p1 p2 -> s/2, p1 p3 -> t/2, p2 p3 -> (-s-t)/2};
propagator = {(l1)^2,(l1+p1)^2,(l1+p1+p2)^2,l2^2-msq, (l1+l2)^2-msq, (l2-p1-p2)^2-msq, (l2-p3)^2, (l1-p3)^2,(l2+p1)^2};
topsector = {1,1,1,1,1,1,1,0,0};
numeric = {s -> 1};
```

```
BLFamilyDefine[family,dimension, propagator,loop,leg,conservation, replacement,topsector,numeric]
```

```
target={BL[dbox,{1,1,1,1,1,1,1,0,-3}],BL[dbox,{1,1,1,1,1,1,1,-1,-2}],
BL[dbox,{1,1,1,1,1,1,1,-2,-1}],BL[dbox,{1,1,1,1,1,1,1,-3,0}],
BL[dbox,{2,1,1,1,1,1,0,0,-2}], BL[dbox,{2,1,1,1,1,1,0,-2,0}],
BL[dbox,{0,1,2,2,1,1,0,0,-2}], BL[dbox,{0,1,2,2,1,1,0,-2,0}]};
```

```
res = BLReduce[target];
```

```
de = BLDifferentialEquation[{t}]
```

- Define the integral family using “BLFamilyDefine”
- Reduce target integrals using “BLReduce”
- Construct the differential equations using “BLDifferentialEquation”

Outline

I. Introduction

II. Block-triangular system and Blade

III. Auxiliary mass flow and AMFlow

IV. Applications

V. Summary and outlook

Auxiliary mass flow

➢ Motivation

- Traditional methods
  - Analytic: less systematic, case-by-case analysis
  - Numerical: less efficient, hard to obtain high precision results
- A possible solution: semi-analytic (semi-numerical) method
  - reserve analytic structure as far as possible
  - perform quickly-converged numerical computation
- A good candidate: differential equations + power series expansion
  - + systematic boundary conditions

Auxiliary mass flow

➢ Dimensionally regulated Feynman integrals

- integrals with auxiliary mass parameter η

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

- obtain physical integrals through

$$I(\vec{\nu}) = \lim_{\eta \to \mathrm{i}0^-} I_{\text{aux}}(\vec{\nu};\eta)$$

- method of region [Beneke and Smirnov, Nucl. Phys. B, 1998]
  - the only contributing region: $\ell_i^\mu \sim \sqrt{\eta}$

$$\frac{1}{((\ell+p)^2 - m^2 - \eta)^\nu} = \frac{1}{(\ell^2 - \eta)^\nu} \sum_{i=0}^{\infty} \frac{(\nu)_i}{i!} \left(-\frac{2\ell \cdot p + p^2 - m^2}{\ell^2 - \eta} \right)^i$$

- fully massive vacuum integrals [Davydychev and Tausk, Nucl, Phys. B, 1993] [Broadhurst, Eur. Phys. J. C, 1999][Schroder and Vuorinen, JHEP, 2005] [Kniehl, Pikelner and Veretin, JHEP, 2017][Luthe, phdthesis, 2015] [Luthe, Maier, Marquard et al, JHEP, 2017]

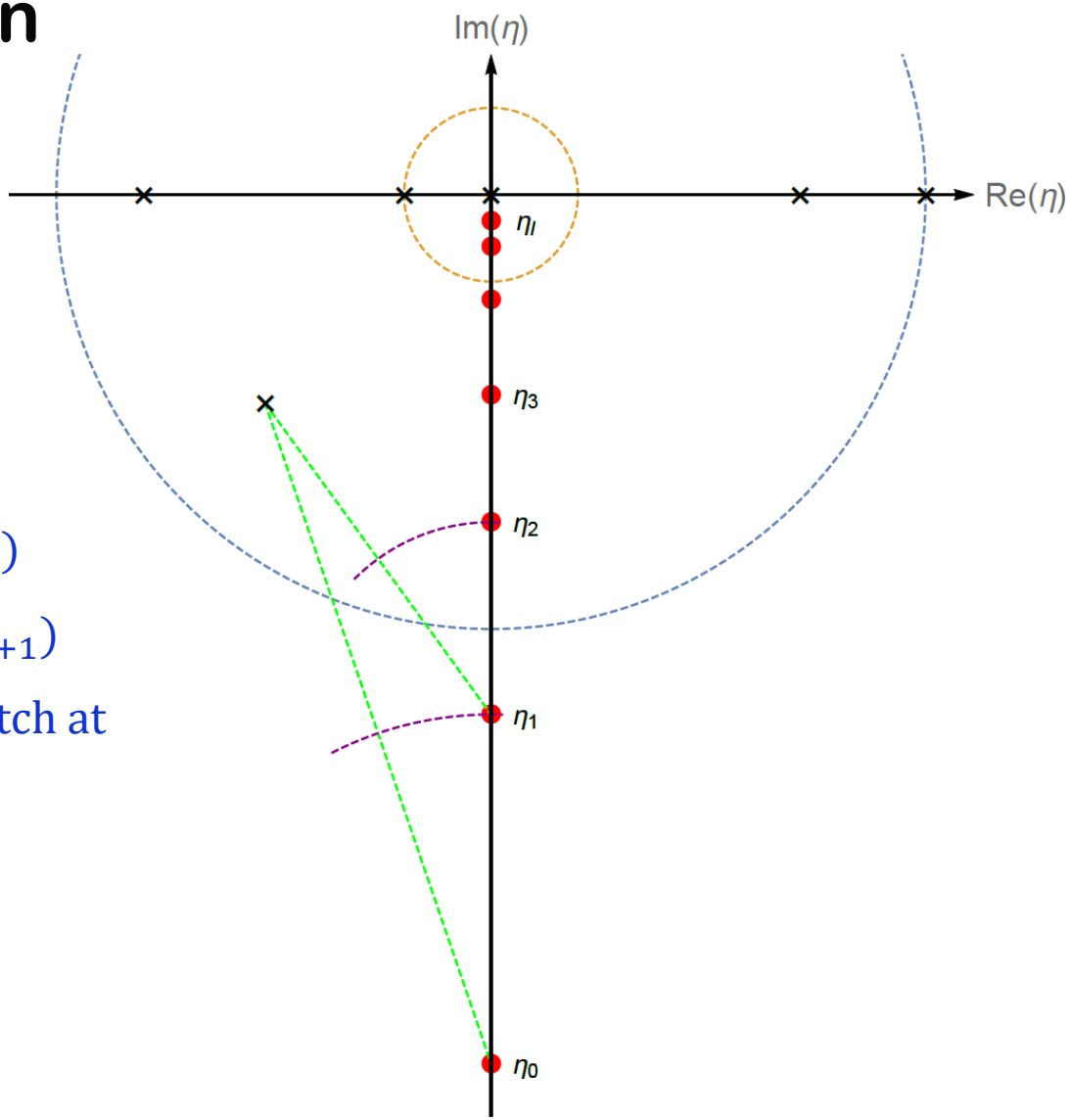
Auxiliary mass flow

➢ Analytic continuation

- differential equations

$$\frac{\partial}{\partial \eta}\vec{\mathcal{I}}_{\text{aux}}(\eta) = A(\eta)\vec{\mathcal{I}}_{\text{aux}}(\eta)$$

- boundary conditions at $\eta = \infty$
- define a path: $\{\eta_0, \eta_1, \dots, \eta_l\}$
- expand at $\eta = \infty$ to estimate $I(\eta_0)$
- expand at $\eta = \eta_i$ to estimate $I(\eta_{i+1})$
- expand formally at $\eta = 0$ and match at $\eta = \eta_l$
  - η_0: outside the larger circle
  - η_l: inside the smaller circle
  - $|\eta_{i+1} - \eta_i| < r_i$



AMFlow

A

AMFlow Free

Project ID: 32748265

Star 10 Fork 2

73 Commits 1 Branch 2 Tags 594 KB Project Storage 1 Release

A proof-of-concept implementation of auxiliary mass flow method.

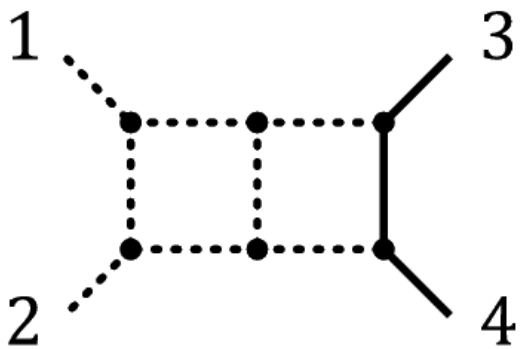
- available at https://gitlab.com/multiloop-pku/amflow
- basic features
  - systematic: works for arbitrary integrals in principle
    - one-loop six-point[Henn, Matijasic and Miczajka, JHEP 2023], two-loop five-point[Badger, Becchetti, Chaubey and Marzucca, JHEP 2023], three-loop three-point [Fael, Lange, Schonwald and Steinhauser, 2302.00693] ...
  - efficient: easy to reach high precision
    - O(100) digits [Badger, Becchetti, Chaubey and Marzucca, JHEP 2023] ..
    - O(1000) digits [Abreu, Becchetti, Duhr and Ozcelik, JHEP 2022]
  - user-friendly: press the button & wait for the results

AMFlow

- examples/automatic_vs_manual

```
(*load the package*)
current = If[$FrontEnd===Null,$InputFileName,NotebookFileName[]]//DirectoryName;
Get[FileNameJoin[{current, "..", "..", "AMFlow.m"}]];
```

```
(*set ibp reducer, could be "FiniteFlow+LiteRed", "Kira" or "FIRE+LiteRed"*)
SetReductionOptions["IBPReducer" -> "Kira"];
```




```
(*configuration of the integral family*)
AMFlowInfo["Family"] = tt;
AMFlowInfo["Loop"] = {l1, l2};
AMFlowInfo["Leg"] = {p1, p2, p3, p4};
AMFlowInfo["Conservation"] = {p4 -> -p1-p2-p3};
AMFlowInfo["Replacement"] = {p1^2 -> 0, p2^2 -> 0, p3^2 -> msq, p4^2 -> msq, (p1+p2)^2 -> s, (p1+p3)^2 -> t};
AMFlowInfo["Propagator"] = {l1^2, (l1+p1)^2, (l1+p1+p2)^2, l2^2, -msq+(l2+p3)^2, (l2+p3+p4)^2, (l1+l2)^2, (l1-p3)^2, (l2+p1)^2};
AMFlowInfo["Numeric"] = {s -> 30, t -> -10/3, msq -> 1};
AMFlowInfo["NThread"] = 4;
```

```
(*SolveIntegrals: computes given integrals with given precision goal up to given eps order*)
(*returned is a list of replacement rules like {j1 → v1, j2 → v2, ...}, where j1, j2, ... are integrals and v1, v2, ... are their results*)
target = {j[tt,1,1,1,1,1,1,1,-3,0], j[tt,1,1,1,1,1,1,1,-2,-1], j[tt,1,1,1,1,1,1,1,-1,-2], j[tt,1,1,1,1,1,1,1,0,-3]};
precision = 20;
epsorder = 4;
auto = SolveIntegrals[target, precision, epsorder];
```

- SetReductionOptions[“IBPReducer” -> “reducer”];
  - “reducer”: “Blade”, “FIRE+LiteRed”, “FiniteFlow+LiteRed”, “Kira”
- SolveIntegrals[targets, precision, epsorder];

AMFlow

- other features
  - compute integrals at any space-time dimension, $D = 6 - 2\epsilon, 3 - 2\epsilon, 11/2 - 2\epsilon$
    - see examples/spacetime_dimension
  - compute integrals with complex masses, $m^2 = 100 - 5i$
    - see examples/complex_kinematics
  - compute phase-space integrals based on reverse unitarity [Anastasiou and Melnikov, Nucl. Phys. B, 2002]
    - $\delta(p_i^2 - m_i^2) \to \frac{1}{[p_i^2 - m_i^2]_c}$
    - see examples/automatic_phase and examples/feynman_prescription

Outline

I. Introduction

II. Block-triangular system and Blade

III. Auxiliary mass flow and AMFlow

IV. Applications

V. Summary and outlook

Applications

➢ $e^+e^- \to q\bar{q}$ @ NNNLO QCD [Xiang Chen, Xin Guan, Chuan-Qi He, **XL** and Yan-Qing Ma, e-Print: 2209.14259]

- Compute the total cross section as a piecewise function of $x \coloneqq 4m^2/s$
  - reduce to master integrals using Blade

$$A = \sum c_i M_i$$

  - construct differential equations of master integrals with respect to x using Blade

$$\frac{\partial}{\partial x} M_i = \sum A_{ij} M_j$$

  - compute the master integrals at a regular point $x = 4/23$ using AMFlow, which serves as boundary conditions

$$M_i(4/23) = \cdots$$

  - solve the differential equations numerically to obtain piecewise functions of master integrals, using the differential equation solver in AMFlow
  - substitute the results of master integrals into the cross section

Outline

I. Introduction

II. Block-triangular system and Blade

III. Auxiliary mass flow and AMFlow

IV. Applications

V. Summary and outlook

Summary and Outlook

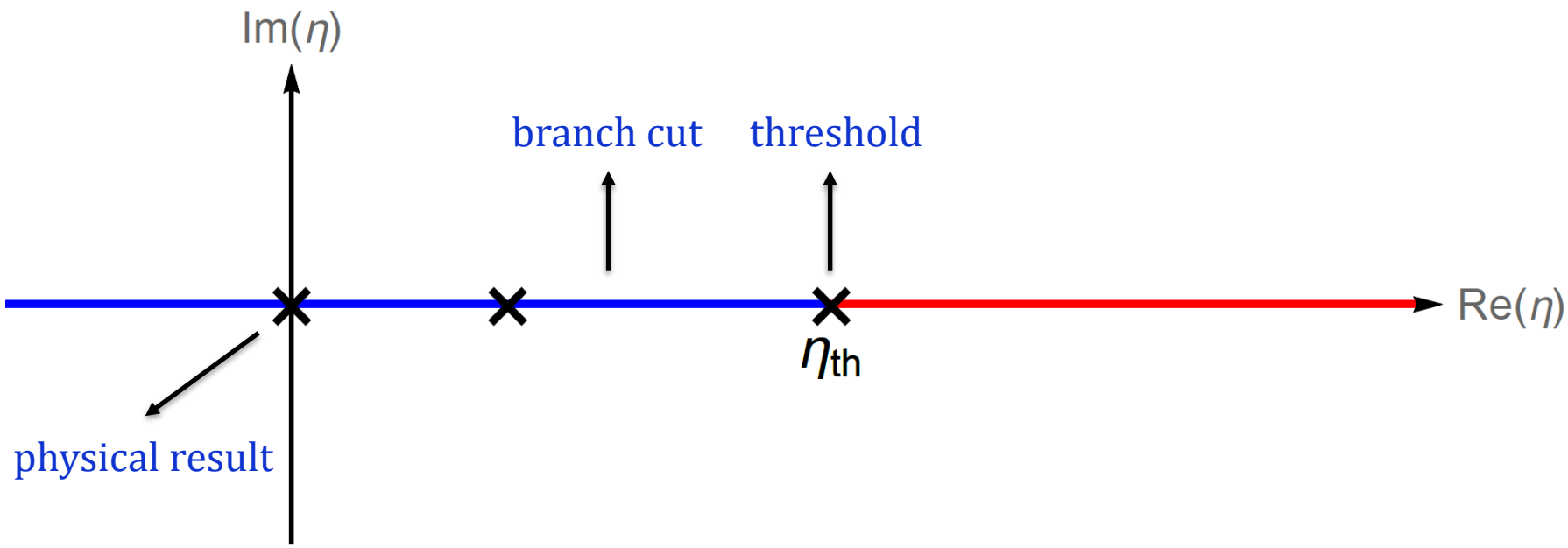
- Integrals reduction: block-triangular system, Blade
- Master integrals computation: auxiliary mass flow, AMFlow
- Solution of univariate problems? Completely automated.
- Solution of multivariate problems?
  - From point to point? OK.
  - More efficient methods? Not known yet.

Thank you!

Auxiliary mass flow

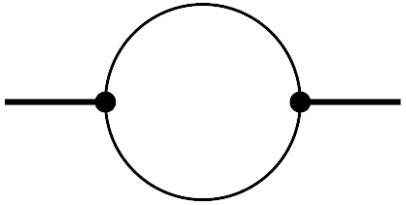
➢ $I(\eta)$ as an analytic function of η

- there should be a maximal threshold $\eta = \eta_{\text{th}}$ on the real axis
  - $I(\eta)$ is real-valued for $\eta > \eta_{\text{th}}$ and complex-valued for $\eta < \eta_{\text{th}}$
- branch cut can be chosen as the straight line connecting $\eta = -\infty$ and $\eta = \eta_{\text{th}}$ along the real axis, such that $I(\eta^*) = I^*(\eta)$



Auxiliary mass flow

➢ A simple example: one-loop massless bubble



$$I(\nu_1,\nu_2) = \int \frac{\mathrm{d}^D\ell}{\mathrm{i}\pi^{D/2}} \frac{1}{(\ell^2)^{\nu_1}((\ell+p)^2)^{\nu_2}}$$

- master integral: $I(1,1)$

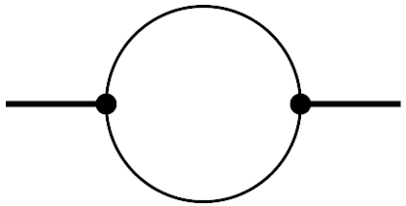
$$I(1,1) = (-p^2 - i0)^{D/2-2} \times \frac{\Gamma\left(2-D/2\right)\Gamma\left(D/2-1\right)^2}{\Gamma(D-2)}$$

$$I(1,1)|_{p^2=1,D=4-2\epsilon} = \frac{1}{\epsilon} + (2-\gamma+i\pi) + O\left(\epsilon^1\right)$$

$$= \frac{1}{\epsilon} + (1.42278 + 3.14159i) + O\left(\epsilon^1\right)$$

Auxiliary mass flow

- insert auxiliary mass



$$I_{\text{aux}}(\nu_1,\nu_2;\eta) = \int \frac{\mathrm{d}^D\ell}{\mathrm{i}\pi^{D/2}} \frac{1}{(\ell^2-\eta)^{\nu_1}((\ell+p)^2-\eta)^{\nu_2}}$$

- master integrals: $\vec{I}_{\text{aux}}(\eta) = \{I_{\text{aux}}(1,0;\eta),\ I_{\text{aux}}(1,1;\eta)\}$
- construct differential equations using IBP reduction

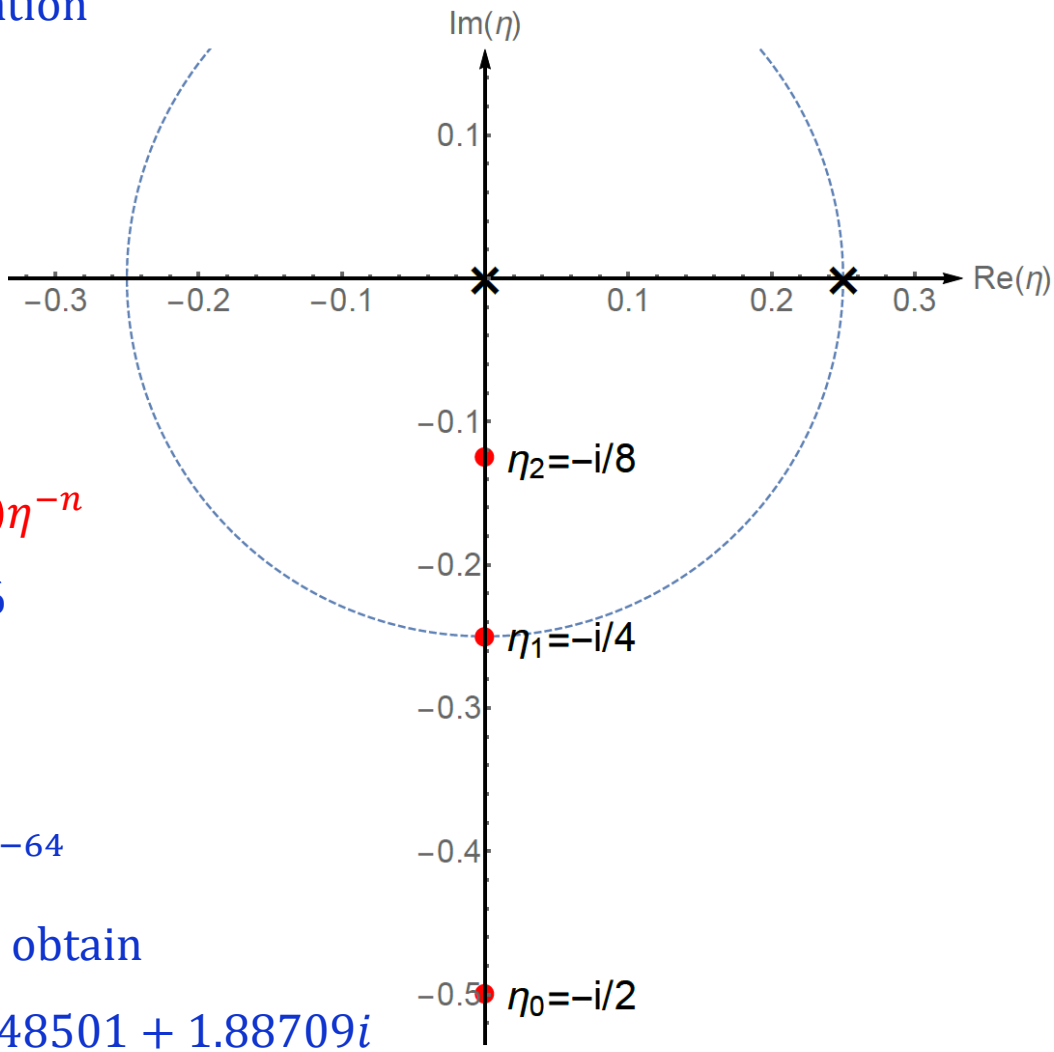
$$\frac{\partial}{\partial\eta}\vec{\mathcal{I}}_{\text{aux}}(\eta) = \begin{pmatrix} \frac{1-\epsilon}{\eta} & 0 \\ \frac{2(\epsilon-1)}{\eta(4\eta-1)} & -\frac{2(2\epsilon-1)}{4\eta-1} \end{pmatrix} \vec{\mathcal{I}}_{\text{aux}}(\eta)$$

- $\eta_{\text{th}} = 1/4$
- boundary conditions

$$I_{\text{aux}}(1,0;\eta) = \eta^{1-\epsilon} \times (-\Gamma(\epsilon-1))$$
$$I_{\text{aux}}(1,1;\eta) \sim \eta^{-\epsilon} \times (\Gamma(\epsilon) + \mathcal{O}(\eta^{-1}))$$

Auxiliary mass flow

- define a path for analytic continuation
  - singularities: $\{0, 1/4\}$
  - $R_{\mathrm{L}} = R_{\mathrm{S}} = 1/4$
  - $\{-i/2,\ -i/4,\ -i/8\}$



- expand near $\eta = \infty$
  - $I_{\mathrm{aux}}(1,1;\eta) = \eta^{-\epsilon} \sum_{n=0}^{\infty} a_n(\epsilon)\eta^{-n}$
    - $a_0(\epsilon) = \epsilon^{-1} - 0.577216$
    - $a_1(\epsilon) = 0.166667$
    - …
    - $a_{100}(\epsilon) = 5.49443 \times 10^{-64}$
  - estimate at $\eta = \eta_0 = -i/2$ to obtain

    $I_{\mathrm{aux}}(1,1;-i/2) = \epsilon^{-1} + 0.0548501 + 1.88709i$

Auxiliary mass flow

- expand near $\eta = \eta_0 = -i/2$
  - $I_{\text{aux}}(1,1;\eta) = \sum_{n=0}^{\infty} a_n(\epsilon)(\eta - \eta_0)^n$
    - $a_0(\epsilon) = \epsilon^{-1} + 0.0548501 + 1.88709i$
    - $a_1(\epsilon) = 0.5714 - 1.77538i$
    - …
    - $a_{100}(\epsilon) = -1.29958 \times 10^{24} + 1.28029 \times 10^{26}$
- estimate at $\eta = \eta_1 = -i/4$ to obtain
  - $I_{\text{aux}}(1,1;-i/4) = \epsilon^{-1} + 0.609168 + 2.13174i$

- expand near $\eta = \eta_1 = -i/4$ and estimate at $\eta = -i/8$ to obtain
  - $I_{\text{aux}}(1,1;-i/8) = \epsilon^{-1} + 0.994236 + 2.42639i$

Auxiliary mass flow

- expand near $\eta = 0$
  - $I_{\text{aux}}(1,1;\eta) = \sum_{n=0}^{\infty} a_n(\epsilon)\eta^n + \eta^{1-\epsilon}\sum_{n=0}^{\infty} b_n(\epsilon)\eta^n$
  - $b_n(\epsilon)$ can be totally determined by sub-topology
    - $b_0(\epsilon) = -2\Gamma(\epsilon - 1)$
    - $b_1(\epsilon) = 4\Gamma(\epsilon - 1)/(\epsilon - 2)$
    - …
  - $a_n(\epsilon)$ cannot be totally determined but can be reduced to $a_0(\epsilon)$
    - $a_1(\epsilon) = 2(2\epsilon - 1)a_0(\epsilon)$
    - $a_2(\epsilon) = 2(2\epsilon - 1)(2\epsilon + 1)a_0(\epsilon)$
    - …
  - match at $\eta = \eta_2 = -i/8$ to obtain $a_0(\epsilon) = \epsilon^{-1} + 1.42278 + 3.14159i$
  - take the limit $\eta \to i0^-$
    - $\lim_{\eta \to i0^-} I_{\text{aux}}(1,1;\eta) = a_0(\epsilon) = \epsilon^{-1} + 1.42278 + 3.14159i$