Solution of differential equations for multiloop integrals with **Libra**[1] package

Roman N. Lee

Budker Institute of Nuclear Physics, Novosibirsk

Advances in Quantum Field Theory, October 12, 2021

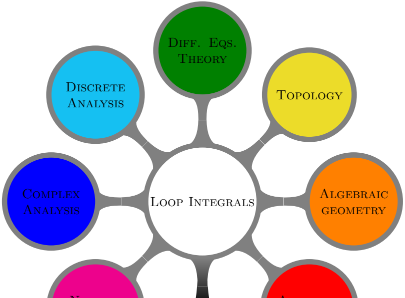
[1] RL, CPC **267** (2021) 108058.

Congratulations!

My congratulations to all heros of the event!
I am very grateful to Vladimir Smirnov for pleasant and fruitful collaboration during many years.

Motivation

- If we want to detect deviations (new physics) from SM, we need to know the predictions of the latter with high precision. In particular, we have to be able to calculate two-loop radiative corrections to various processes [A. Arbuzov's talk yesterday].
- Only quite recently the methods of multiloop calculations have reached the point where they can be really helpful with this goal.
- Besides these practical purposes, multiloop calculations provide a perfect polygon for trying the methods from various mathematical fields: differential equations, complex analysis, number theory



Modern approach to diagrams calculation

1. Consider a family of integrals

$$j(n_1, \dots, n_N) = \int \frac{d^d l_1 \dots d^d l_L}{D_1^{n_1} \dots D_N^{n_N}} .$$

Integrals are functions of kinematic variables x_i and $d = 4 - 2\epsilon$.

Modern approach to diagrams calculation

1. Consider a family of integrals

$$j(n_1, \ldots, n_N) = \int \frac{d^d l_1 \ldots d^d l_L}{D_1^{n_1} \ldots D_N^{n_N}}\,.$$

   Integrals are functions of kinematic variables x_i and $d = 4 - 2\epsilon$.

2. Arrange IBP reduction [Chetyrkin and Tkachov, 1981, Laporta, 2000] to master integrals $\boldsymbol{j} = (j_1, \ldots j_s)^\intercal$ [V. Smirnov's talk today].

Modern approach to diagrams calculation

1. Consider a family of integrals

$$j(n_1,\ldots,n_N) = \int \frac{d^d l_1 \ldots d^d l_L}{D_1^{n_1} \ldots D_N^{n_N}}\,.$$

Integrals are functions of kinematic variables x_i and $d = 4 - 2\epsilon$.

2. Arrange IBP reduction [Chetyrkin and Tkachov, 1981, Laporta, 2000] to master integrals $\boldsymbol{j} = (j_1, \ldots j_s)^\intercal$ [V. Smirnov's talk today].
3. Find differential equations [Kotikov, 1991, Remiddi, 1997] (and/or dimensional recurrences [Tarasov, 1996]) for master integrals

Differential equations

$$\frac{\partial}{\partial x_i}\boldsymbol{j} = M(\boldsymbol{x},\epsilon)\boldsymbol{j}$$

Dimensional recurrences

$$\boldsymbol{j}(\epsilon+1) = R(\boldsymbol{x},\epsilon)\boldsymbol{j}(\epsilon)$$

M and R are $n \times n$ matrices rational in $\boldsymbol{x}$ and ϵ.

Modern approach to diagrams calculation

1. Consider a family of integrals

$$j(n_1,\ldots,n_N) = \int \frac{d^d l_1 \ldots d^d l_L}{D_1^{n_1} \ldots D_N^{n_N}}\,.$$

Integrals are functions of kinematic variables x_i and $d = 4 - 2\epsilon$.

2. Arrange IBP reduction [Chetyrkin and Tkachov, 1981, Laporta, 2000] to master integrals $\boldsymbol{j} = (j_1,\ldots j_s)^\intercal$ [V. Smirnov's talk today].
3. Find differential equations [Kotikov, 1991, Remiddi, 1997] (and/or dimensional recurrences [Tarasov, 1996]) for master integrals

Differential equations

$$\frac{\partial}{\partial x_i}\boldsymbol{j} = M(\boldsymbol{x},\epsilon)\boldsymbol{j}$$

Dimensional recurrences

$$\boldsymbol{j}(\epsilon+1) = R(\boldsymbol{x},\epsilon)\boldsymbol{j}(\epsilon)$$

M and R are $n \times n$ matrices rational in $\boldsymbol{x}$ and ϵ.

4. Find **general solution**.

Modern approach to diagrams calculation

1. Consider a family of integrals

$$j(n_1, \ldots, n_N) = \int \frac{d^d l_1 \ldots d^d l_L}{D_1^{n_1} \ldots D_N^{n_N}} .$$

Integrals are functions of kinematic variables x_i and $d = 4 - 2\epsilon$.

2. Arrange IBP reduction [Chetyrkin and Tkachov, 1981, Laporta, 2000] to master integrals $\boldsymbol{j} = (j_1, \ldots j_s)^\intercal$ [V. Smirnov's talk today].
3. Find differential equations [Kotikov, 1991, Remiddi, 1997] (and/or dimensional recurrences [Tarasov, 1996]) for master integrals

Differential equations

$$\frac{\partial}{\partial x_i}\boldsymbol{j} = M(\boldsymbol{x}, \epsilon)\boldsymbol{j}$$

Dimensional recurrences

$$\boldsymbol{j}(\epsilon + 1) = R(\boldsymbol{x}, \epsilon)\boldsymbol{j}(\epsilon)$$

M and R are $n \times n$ matrices rational in $\boldsymbol{x}$ and ϵ.

4. Find **general solution**.
5. Use other methods for **boundary conditions**.

IBP reduction

Note on phase-space integrals

- Phase-space integrals can be transformed in a standard way into loop integrals with cut propagator:

$$\frac{d^{d-1}p}{(2\pi)^{d-1}2\varepsilon_p} = \frac{d^d p}{(2\pi)^d}2\pi\delta_+\left(p^2-m^2\right) = \frac{d^{d-1}p}{(2\pi)^d}(-i)\oint_{\varepsilon_p} dp_0\left(p^2-m^2\right)^{-1}$$

- As IBP identities are insensitive to the integration contour, provided that it does not lead to surface terms, we can treat the cut propagators in the same way as uncut ones.
- There are two things to take care of: first, shift symmetries which mix cut with uncut denominators should be omitted, second, positive (integer) powers of cut denominators are equal to zero.

ϵ-form

Differential equation

$$\partial \boldsymbol{J}(x) / \partial x = M(x, \epsilon) \boldsymbol{J}(x)$$

Function change

$$\boldsymbol{J}(x) = T(x, \epsilon) \tilde{\boldsymbol{J}}(x)$$

In particular, we can choose master integrals in infinitely many ways.

ϵ-form

Differential equation

$$\partial \boldsymbol{J}(x) / \partial x = M(x, \epsilon) \boldsymbol{J}(x)$$

Function change

$$\boldsymbol{J}(x) = T(x, \epsilon) \tilde{\boldsymbol{J}}(x)$$

In particular, we can choose master integrals in infinitely many ways.

Remarkable observation [Henn, 2013]

There often exists a choice of master integrals such that

$$\partial \tilde{\boldsymbol{J}}(x) / \partial x = \epsilon S(x) \tilde{\boldsymbol{J}}(x)$$

This form makes finding a solution in the form of ϵ-expansion very simple.

ϵ-form

Differential equation

$$\partial \boldsymbol{J}(x)/\partial x = M(x,\epsilon)\boldsymbol{J}(x)$$

Function change

$$\boldsymbol{J}(x) = T(x,\epsilon)\tilde{\boldsymbol{J}}(x)$$

In particular, we can choose master integrals in infinitely many ways.

Remarkable observation [Henn, 2013]

There often exists a choice of master integrals such that

$$\partial \tilde{\boldsymbol{J}}(x)/\partial x = \epsilon S(x)\tilde{\boldsymbol{J}}(x)$$

This form makes finding a solution in the form of ϵ-expansion very simple.

ϵ-form

Differential equation

$$\partial \boldsymbol{J}(x)/\partial x = M(x,\epsilon)\boldsymbol{J}(x)$$

Function change

$$\boldsymbol{J}(x) = T(x,\epsilon)\tilde{\boldsymbol{J}}(x)$$

In particular, we can choose master integrals in infinitely many ways.

Remarkable observation [Henn, 2013]

There often exists a choice of master integrals such that

$$\partial \tilde{\boldsymbol{J}}(x)/\partial x = \epsilon S(x)\tilde{\boldsymbol{J}}(x)$$

This form makes finding a solution in the form of ϵ-expansion very simple.

How to find canonical basis?

ϵ-form

Differential equation

$$\partial \boldsymbol{J}(x)/\partial x = M(x,\epsilon)\boldsymbol{J}(x)$$

Function change

$$\boldsymbol{J}(x) = T(x,\epsilon)\tilde{\boldsymbol{J}}(x)$$

In particular, we can choose master integrals in infinitely many ways.

Remarkable observation [Henn, 2013]

There often exists a choice of master integrals such that

$$\partial \tilde{\boldsymbol{J}}(x)/\partial x = \epsilon S(x)\tilde{\boldsymbol{J}}(x)$$

This form makes finding a solution in the form of ϵ-expansion very simple.

Algorithmic approach [RL, 2015]: general idea

Perform many "elementary" transformations gradually improving properties of the system.

General structure of reduction algorithm

Algorithm proceeds in three major stages, each involving a sequence of "elementary" transformations.

General structure of reduction algorithm

Algorithm proceeds in three major stages, each involving a sequence of "elementary" transformations.

1. *"Fuchsification"*: Eliminating higher-order poles

Input: Rational matrix $M(x, \epsilon)$
Output: Rational matrix with only simple poles on the extended complex plane, $M(x, \epsilon) = \sum_k \frac{M_k(\epsilon)}{x - a_k}$.

General structure of reduction algorithm

Algorithm proceeds in three major stages, each involving a sequence of "elementary" transformations.

1. *"Fuchsification"*: Eliminating higher-order poles

Input: Rational matrix $M(x,\epsilon)$
Output: Rational matrix with only simple poles on the extended complex plane, $M(x,\epsilon)=\sum_k \frac{M_k(\epsilon)}{x-a_k}$.

2. *Normalization*: Normalizing eigenvalues

Input: Matrix from the previous step, $M(x,\epsilon)=\sum_k \frac{M_k(\epsilon)}{x-a_k}$.
Output: Matrix of the same form, but with the eigenvalues of all $M_k(\epsilon)$ being proportional to ϵ.

General structure of reduction algorithm

Algorithm proceeds in three major stages, each involving a sequence of "elementary" transformations.

1. *"Fuchsification"*: Eliminating higher-order poles

Input: Rational matrix $M(x, \epsilon)$
Output: Rational matrix with only simple poles on the extended complex plane, $M(x, \epsilon) = \sum_k \frac{M_k(\epsilon)}{x - a_k}$.

2. *Normalization*: Normalizing eigenvalues

Input: Matrix from the previous step, $M(x, \epsilon) = \sum_k \frac{M_k(\epsilon)}{x - a_k}$.
Output: Matrix of the same form, but with the eigenvalues of all $M_k(\epsilon)$ being proportional to ϵ.

3. *Factorization*: Factoring out ϵ

Input: Matrix from the previous step.
Output: Matrix in ϵ-form, $M(x, \epsilon) = \epsilon S(x) = \epsilon \sum_k \frac{S_k}{x - a_k}$.

Balance

Balance transformation

$$T(x) = \overline{P} + \frac{x - x_2}{x - x_1} P\,,$$

where P is some projector and $\overline{P} = I - P$. When $x_1 = \infty$ or $x_2 = \infty$ omit denominator/numerator.

Balance transformation changes properties (pole order and eigenvalues of matrix residue) of the differential system at $x = x_1$ and $x = x_2$ only.

Balance

Balance transformation

$$T(x) = \overline{P} + \frac{x - x_2}{x - x_1} P\,,$$

where P is some projector and $\overline{P} = I - P$. When $x_1 = \infty$ or $x_2 = \infty$ omit denominator/numerator.

Balance transformation changes properties (pole order and eigenvalues of matrix residue) of the differential system at $x = x_1$ and $x = x_2$ only.

Programs

Further details of the algorithm deserve a special talk, but the good news is that we now have **programs**! At least, three public ones: `epsilon`, `Fuchsia`, `Libra`.

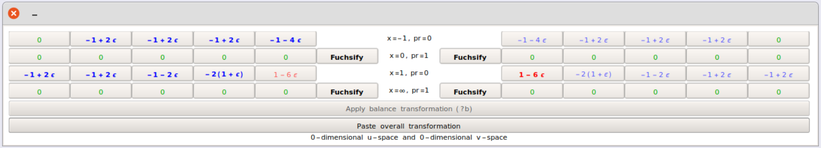
Libra interface for reduction

Automatic tool (useful for simple cases)

In[1]: t=Rookie[M,x,ϵ];

Interactive tool (useful for most cases)

In[1]: t=VisTransformation[M,x,ϵ];



Manual tool (useful for really hard cases)

In[1]: u=GetSubspaces[M,{x,0},ϵ][[1]];v=GetSubspaces[M...

Factoring ϵ

In[2]: t=FactorOut[M,x,ϵ,μ];

General solution

Differential system in ϵ-form

$$\partial_x \boldsymbol{J} = \epsilon S(x) \boldsymbol{J}\,, \qquad S(x) = \sum_k \frac{S_k}{x - a_k}.$$

General solution (evolution operator)

$$U(x, x_0) = \mathrm{Pexp}\left[\epsilon \int_{x_0}^{x} dx_1\, S(x_1)\right]$$

```
In[3]:  U=PexpExpansion[{S,6},x];
```

General solution

Differential system in ϵ-form

$$\partial_x \boldsymbol{J} = \epsilon S(x)\boldsymbol{J}\,, \qquad S(x) = \sum_k \frac{S_k}{x - a_k}.$$

General solution (evolution operator)

$$U(x, x_0) = \sum_n \epsilon^n \iiiint\limits_{x>x_n>\ldots>x_1>x_0} dx_n \ldots dx_1 S(x_n) \ldots S(x_1)$$

```
In[3]:  U=PexpExpansion[{S,6},x];
```

General solution

Differential system in ϵ-form

$$\partial_x \boldsymbol{J} = \epsilon S(x) \boldsymbol{J}\,, \qquad S(x) = \sum_k \frac{S_k}{x - a_k}.$$

General solution (evolution operator)

$$U(x, x_0) = \sum_n \epsilon^n \iiiint\limits_{x>x_n>\ldots>x_1>x_0} dx_n \ldots dx_1 S(x_n) \ldots S(x_1)$$

```
In[3]:  U=PexpExpansion[{S,6},x];
```

Goncharov polylogarithms

So, $U(x, x_0)$ is expressed via Goncharov polylogs [Goncharov, 1998]

$$G(a_n, \ldots, a_1|x) = \iiiint\limits_{x>x_n>\ldots>x_1>0} \frac{dx_1}{x_1 - a_1} \cdots \frac{dx_n}{x_n - a_n}$$

Perfect class of functions: numerical evaluation, analytic continuation, series representation, functional identities, relations to classical polylogs.

Boundary conditions

Suppose we have found a transformation $T(x) = T(x, \epsilon)$ to ϵ-form, $\boldsymbol{j} = T\boldsymbol{J}$. Then we can write

$$\begin{aligned}\boldsymbol{J}(x) &= U(x, x_0)\boldsymbol{J}(x_0),\\ \boldsymbol{j}(x) &= T(x)U(x, x_0)[T(x_0)]^{-1}\boldsymbol{j}(x_0)\end{aligned}$$

But the point x_0 should be somewhat special to simplify the evaluation of $\boldsymbol{j}(x_0)$ as compared to $\boldsymbol{j}(x)$. With no known exceptions, "special" boils down to "singular", i.e., we can expect simplifications for x_0 being a singular point of the differential system. Let it be $x_0 = 0$ for simplicity.

Boundary conditions

Suppose we have found a transformation $T(x) = T(x, \epsilon)$ to ϵ-form, $\boldsymbol{j} = T\boldsymbol{J}$. Then we can write

$$\begin{aligned}\boldsymbol{J}(x) &= U(x, x_0)\boldsymbol{J}(x_0), \\ \boldsymbol{j}(x) &= T(x)U(x, x_0)[T(x_0)]^{-1}\boldsymbol{j}(x_0)\end{aligned}$$

But the point x_0 should be somewhat special to simplify the evaluation of $\boldsymbol{j}(x_0)$ as compared to $\boldsymbol{j}(x)$. With no known exceptions, "special" boils down to "singular", i.e., we can expect simplifications for x_0 being a singular point of the differential system. Let it be $x_0 = 0$ for simplicity.

Problem

$U(x, x_0)$ diverges when x_0 tends to zero. Therefore, we have to consider not the values, but the asymptotics of $\boldsymbol{j}(x_0)$ at $x = 0$.

Boundary conditions

Regularized evolution operator

$$U(x,\underline{0}) = \lim_{x_0\to 0} U(x,x_0)x_0^{\epsilon S_0},$$

where $S_0 = \mathrm{Res}_{x=0}\, S(x)$.

Boundary conditions

Regularized evolution operator

$$U(x,\underline{0}) = \lim_{x_0 \to 0} U(x,x_0) x_0^{\epsilon S_0},$$

where $S_0 = \mathrm{Res}_{x=0}\, S(x)$.

1. $U(x,\underline{0})$ has no divergences.
2. $U(x,\underline{0})$ is a general solution.
3. $U(x,\underline{0}) \to x^{\epsilon S_0}$ when $x \to 0$.

Boundary conditions

Regularized evolution operator

$$U(x,\underline{0}) = \lim_{x_0\to 0} U(x,x_0)x_0^{\epsilon S_0},$$

where $S_0 = \mathrm{Res}_{x=0}\, S(x)$.

1. $U(x,\underline{0})$ has no divergences.
2. $U(x,\underline{0})$ is a general solution.
3. $U(x,\underline{0}) \to x^{\epsilon S_0}$ when $x \to 0$.

Specific solution reads $\boldsymbol{j}(x) = T(x)U(x,\underline{0})\boldsymbol{C}$. The column of boundary constants $\boldsymbol{C}$ can be fixed by evaluating some coefficients in the asymptotics of $\boldsymbol{j}(x)$ when $x \to 0$.

Boundary conditions

Regularized evolution operator

$$U(x,\underline{0}) = \lim_{x_0\to 0} U(x,x_0)x_0^{\epsilon S_0},$$

where $S_0 = \operatorname{Res}_{x=0} S(x)$.

1. $U(x,\underline{0})$ has no divergences.
2. $U(x,\underline{0})$ is a general solution.
3. $U(x,\underline{0}) \to x^{\epsilon S_0}$ when $x \to 0$.

Specific solution reads $\boldsymbol{j}(x) = T(x)U(x,\underline{0})\boldsymbol{C}$. The column of boundary constants $\boldsymbol{C}$ can be fixed by evaluating some coefficients in the asymptotics of $\boldsymbol{j}(x)$ when $x \to 0$.

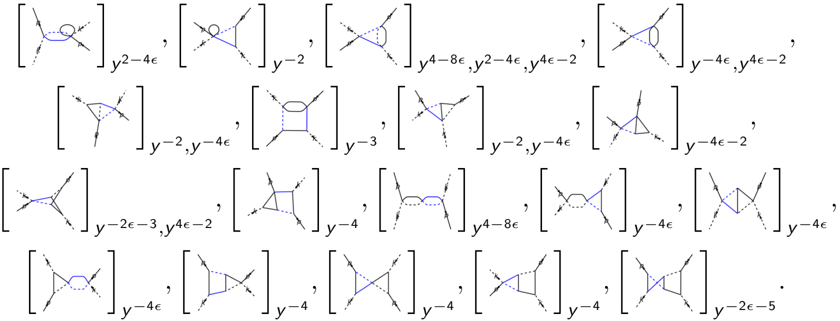
Good news

Libra can determine which asymptotic coefficients, $\boldsymbol{c}$, are sufficient to calculate and find the "adapter" matrix L in $\boldsymbol{C} = L\boldsymbol{c}$.
NB: for regular point, of course, $\boldsymbol{c} = \boldsymbol{j}(0)$ and $L = T^{-1}(0)$.

```
In[4]:  {L,cs}=GetLcs[S,T,{x,0}];
```

Example: boundary conditions for $\sigma_{e^-\gamma\to e^-\gamma}$ @NLO

The following threshold ($s \to 1$, $x \to 0$) asymptotic coefficients are to be calculated:

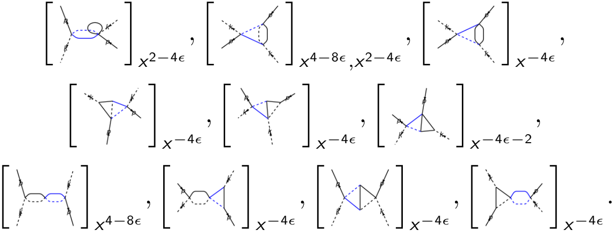


Here $\left[\text{integral}\right]_{x^\alpha}$ denotes the coefficient in front of x^α in the small-x asymptotics of integral ($x \approx \frac{1}{2}\sqrt{s-1}$).

Example: Boundary conditions for $\sigma_{e^-\gamma\to e^-\gamma}$ @NLO

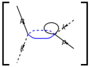
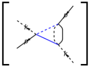
[RL,Lyubyakin,Stocky(2020); RL,Schwartz,Zhang (2021)]

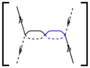
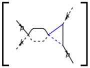
- Selection rule by the fractional power $n\epsilon$ leaves us with 11 possibly nonzero constants.The fractional power -4ϵ corresponds to the hard momentum flowing over the black lines, while -8ϵ — to soft momentum.



Example: Boundary conditions for $\sigma_{e^-\gamma\to e^-\gamma}$ @NLO
[RL,Lyubyakin,Stocky(2020); RL,Schwartz,Zhang (2021)]

- Selection rule by the fractional power $n\epsilon$ leaves us with 11 possibly nonzero constants.The fractional power -4ϵ corresponds to the hard momentum flowing over the black lines, while -8ϵ — to soft momentum.
- Selection rule by integer power reduces the number to 5.

 $_{x^{2-4\epsilon}}$,  $_{x^{4-8\epsilon},x^{2-4\epsilon}}$,

 $_{x^{4-8\epsilon}}$,  $_{x^{-4\epsilon}}$

General solution, Frobenius method

In many applications there is a natural small parameter. E.g., for $e^+e^- \to X$ the electron mass is small, but can not be put to zero. Instead one should expand the regularized evolution operator

$$U(x, \underline{0}) = \lim_{x_0 \to 0} U(x, x_0) x_0^{S_0}$$

in generalized power series. **Libra** has tools for it. It closely follows the approach described in Ref. [RL, Smirnov, and Smirnov, 2018].

$U(x, \underline{0})$ as generalized power series

Recursion data for (matrix) coefficients of U:

```
In[1]:  sdata=SeriesSolutionData[S,x,x];
```

Using data for expanding to fixed order:

```
In[2]:  Uexp=ConstructSeriesSolution[sdata,x,6];
```

Algebraic extensions

- Sometimes, in order to find the transformation to ϵ-form, one has to extend the class of transformations by passing from x to y, such that $x = x(y)$ is some rational function. **Libra** has tool for it:

```
In[1]:  ChangeVar[ds,x→(4 y*y)/(1 - y*y),y];
```

Algebraic extensions

- Sometimes, in order to find the transformation to ϵ-form, one has to extend the class of transformations by passing from x to y, such that $x = x(y)$ is some rational function. **Libra** has tool for it:
  ```
  In[1]:  ChangeVar[ds,x→(4 y*y)/(1 - y*y),y];
  ```
- Moreover, in many cases there is no common rationalizing variable. Thus, **Libra** implements a more powerful way to treat such algebraic extensions, with
  ```
  In[1]:  AddNotation[ds,y → x(1-y*y) - 4 y*y];
  ```
  One may add as many notations as needed, and **Libra** will take care of them (minimizing their appearance, correctly treating their differentiation).

Algebraic extensions

- Sometimes, in order to find the transformation to ϵ-form, one has to extend the class of transformations by passing from x to y, such that $x = x(y)$ is some rational function. **Libra** has tool for it:
  ```
  In[1]:  ChangeVar[ds,x→(4 y*y)/(1 - y*y),y];
  ```
- Moreover, in many cases there is no common rationalizing variable. Thus, **Libra** implements a more powerful way to treat such algebraic extensions, with
  ```
  In[1]:  AddNotation[ds,y → x(1-y*y) - 4 y*y];
  ```
  One may add as many notations as needed, and **Libra** will take care of them (minimizing their appearance, correctly treating their differentiation).
- There is a bunch of functions related to treating the algebraic extensions: `QuolyMod, DiffMod, SeriesCoefficientMod, EValuesMod,` etc. These functions can be used also to treat irreducible denominators, like $x^2 + x + 1$ in a way which do not introduce radicals.

Irreducible cases

- As we know, even with algebraic extensions it is not always possible to reduce the system to ϵ-form. Sometimes the integrals just can not be expressed via polylogs, [see D. Broadhurst's talk yesterday].
- In Ref. [RL and Pomeransky, 2017] the criterion of irreducibility has been derived. The reducibility has been shown to correspond to triviality of some holomorphic vector bundle on the Riemann sphere. Thanks to Birkhoff-Grothendieck theorem, it is possible to constructively decide this. **Libra** has the corresponding tool: the command
  ```
  In[3]:  {T1,T2,T3}= BirkhoffGrothendieck[T,x];
  ```
  decomposes Laurent-polynomial matrix T into the product $T_1 T_2 T_3$, where T_1, T_1^{-1} are polynomial in x, T_3, T_3^{-1} are polynomial in x^{-1}, and $T_2 = \mathrm{diag}(x^{n_1}, \ldots x^{n_k})$. The bundle is trivial iff $T_2 = 1$.

Summary

- Modern multiloop calculation techniques can really help in NNLO calculation useful for the experiments.
- **Libra** can really help in applying the differential equations method. It has tools
  - for the reduction of the differential system to ϵ-form,
  - for the construction of general solution in terms of Goncharov's polylogs,
  - for determining the minimal set of asymptotic coefficients to be evaluated to fix the boundary conditions,
  - for constructing Frobenius expansion,
  - for treating the algebraic extensions,
  - for detecting the irreducible cases.
  - It can work with univariate and multivariate systems.

Outlook

- `Libra` improvements:
  - Improve automatic tool `Rookie[M,x,`ϵ`]`.
  - Better treatment of algebraic extensions, especially, for multivariate case.
- Differential equations method:
  - construct a systematic approach to irreducible cases. In particular, Birkhoff-Grothendieck factorizations seems to be carry a lot of information yet to be properly understood and used.

Thank you!

References

K. G. Chetyrkin and F. V. Tkachov. Integration by parts: The algorithm to calculate β-functions in 4 loops. *Nucl. Phys. B*, 192:159, 1981.

Alexander B Goncharov. Multiple polylogarithms, cyclotomy and modular complexes. *Mathematical Research Letters*, 5:497–516, 1998.

Johannes M. Henn. Multiloop integrals in dimensional regularization made simple. *Phys.Rev.Lett.*, 110(25):251601, 2013. doi: 10.1103/PhysRevLett.110.251601.

A. V. Kotikov. Differential equation method: The Calculation of N point Feynman diagrams. *Phys. Lett.*, B267:123–127, 1991. doi: 10.1016/0370-2693(91)90536-Y. [Erratum: Phys. Lett.B295,409(1992)].

S. Laporta. High precision calculation of multiloop feynman integrals by difference equations. *Int. J. Mod. Phys. A*, 15:5087, 2000.

Ettore Remiddi. Differential equations for Feynman graph amplitudes. *Nuovo Cim.*, A110:1435–1452, 1997.

RL. Reducing differential equations for multiloop master integrals. *J. High Energy Phys.*, 1504:108, 2015. doi: 10.1007/JHEP04(2015)108.

RL and Andrei A. Pomeransky. Normalized Fuchsian form on Riemann sphere and differential equations for multiloop integrals. 2017.

RL, Alexander V. Smirnov, and Vladimir A. Smirnov. Solving differential equations for Feynman integrals by expansions near singular points. *JHEP*, 03:008, 2018. doi: 10.1007/JHEP03(2018)008.

O. V. Tarasov. Connection between feynman integrals having different values of the space-time dimension. *Phys. Rev. D*, 54:6479, 1996. doi: 10.1103/PhysRevD.54.6479.