First order factorizable systems of differential equations in one variable

Nikolai Fadeev

Research Institute for Symbolic Computation/RISC Software GmbH

July 27, 2020

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 764850 (SAGEX).

Motivation

Multi-loop computations in gauge theories (example of form factors)

- Find all Feynman diagrams (with particular color, Dirac and momentum structure) associated with the wanted amplitude
- Reduce them to Master Integrals (MI) that depend on some parameter x (typically $s = s(x)$) as well as $\epsilon = \frac{4-D}{2}$ in the dimensional regularisation (dimReg) $\mathcal{I} = (\mathcal{I}_1(x,\epsilon), \ldots, \mathcal{I}_n(x,\epsilon))$

Motivation

Multi-loop computations in gauge theories (example of form factors)

- Find all Feynman diagrams (with particular color, Dirac and momentum structure) associated with the wanted amplitude
- Reduce them to Master Integrals (MI) that depend on some parameter x (typically $s = s(x)$) as well as $\epsilon = \frac{4-D}{2}$ in the dimensional regularisation (dimReg) $\mathcal{I} = (\mathcal{I}_1(x,\epsilon), \dots, \mathcal{I}_n(x,\epsilon))$

Question: How to compute explicitely those $\mathcal{I}$?

Motivation

Multi-loop computations in gauge theories (example of form factors)

- Find all Feynman diagrams (with particular color, Dirac and momentum structure) associated with the wanted amplitude
- Reduce them to Master Integrals (MI) that depend on some parameter x (typically $s = s(x)$) as well as $\epsilon = \frac{4-D}{2}$ in the dimensional regularisation (dimReg) $\mathcal{I} = (\mathcal{I}_1(x,\epsilon), \dots, \mathcal{I}_n(x,\epsilon))$

Question: How to compute explicitely those $\mathcal{I}$?

- Integration by part/differentiation methods give us a system in the $\mathcal{I}$ MI:

$$\frac{d}{dx}\mathcal{I}(x,\epsilon) = \mathcal{M}(x,\epsilon)\mathcal{I}(x,\epsilon) + \mathcal{R}(x,\epsilon)$$

Motivation

Multi-loop computations in gauge theories (example of form factors)

- Find all Feynman diagrams (with particular color, Dirac and momentum structure) associated with the wanted amplitude
- Reduce them to Master Integrals (MI) that depend on some parameter x (typically $s = s(x)$) as well as $\epsilon = \frac{4-D}{2}$ in the dimensional regularisation (dimReg) $\mathcal{I} = (\mathcal{I}_1(x,\epsilon), \ldots, \mathcal{I}_n(x,\epsilon))$

Question: How to compute explicitely those $\mathcal{I}$?

- Integration by part/differentiation methods give us a system in the $\mathcal{I}$ MI:

$$\frac{d}{dx}\mathcal{I}(x,\epsilon) = \mathcal{M}(x,\epsilon)\mathcal{I}(x,\epsilon) + \mathcal{R}(x,\epsilon)$$

We now need some algorithm to solve this system of first order differential equations.

First method [arXiv:1810.12261] (Blümlein, Schneider and al.)

- Differentiation of MI $\rightarrow \underbrace{\text{elements of } \mathcal{I}}_{\mathcal{M}\mathcal{I}} + \underbrace{\text{new integrals (Base Integrals – BI)}}_{\mathcal{R} \text{ (inhomogeneous part)}}$

$$\frac{d}{dx}\mathcal{I}(x,\epsilon) = \mathcal{M}(x,\epsilon)\mathcal{I}(x,\epsilon) + \mathcal{R}(x,\epsilon) \tag{1}$$

Assumptions

- $\mathcal{M} \in \mathcal{M}_n(\mathbb{K}(\epsilon, x))$ ($\mathbb{K}$ is a computable subfield of $\mathbb{R}$) and $\mathcal{M}$ has no poles around $\epsilon = 0$ (can be achieved by index-shift)
- Each element of $\mathcal{R} = (\mathcal{R}_1, \ldots, \mathcal{R}_n)$ and $\mathcal{I} = (\mathcal{I}_1, \ldots, \mathcal{I}_n)$ can be Laurent-expanded around $\epsilon = 0$ as

$$\exists k \in \mathbb{N} \quad \forall i \in \{1, \ldots, n\} \quad \mathcal{R}_i = \sum_{j=-k}^{\infty} \mathcal{R}_i^{(j)} \epsilon^j, \quad \mathcal{I}_i = \sum_{j=-k}^{\infty} \mathcal{I}_i^{(j)} \epsilon^j$$

- The given coefficients $\mathcal{R}_i^{(j)}$ are polynomial expressions in terms of **hyperexponential functions** and **iterated integrals over hyperexponential functions**.
- The searched coefficients $\mathcal{I}_i^{(j)}$ are assumed to be from the same function space.

Definition – Hyperexponential functions

A function $f(x)$ over some field $\mathbb{K}$ is called **hyperexponential** if there exists some non zero $r(x) \in \mathbb{K}(x)$ such that

$$\frac{\partial_x f(x)}{f(x)} = r(x) \Leftrightarrow \exists l \in \mathbb{K},\ f(x) = e^{\int_l^x \mathrm{d}y r(y)}$$

An iterated integral over an hyperexponential function is an integral

$$I(x) = \int_{l_0}^{x} \mathrm{d}x_1 f_1(x_1) \int_{l_1}^{x_1} \mathrm{d}x_2 f(x_2) \cdots \int_{l_{n-1}}^{x_{n-1}} \mathrm{d}x_n f(x_n) \text{ with } l_0, \ldots, l_n \in \mathbb{K}$$

Such that for all $i \in \{1, \ldots, n\}$, $f_i(x)$ is hyperexponential.

Examples: e^x, $\sqrt{1-x}$, $f(x)^q$, where $f(x) \in \mathbb{K}(x)$ and $q \in \mathbb{Q}$, etc.
Notes:

- Closed under multiplication, inversion, differentiation
- Rather general setting

Remarks:

1. One can distinguish the MI sector-wise, the sector being a set of maximum number of non-vanishing propagators in a single Feynman graph.
2. The DE of some MI only depends on MI from the same sector or subsectors.

Remarks:

1. One can distinguish the MI sector-wise, the sector being a set of maximum number of non-vanishing propagators in a single Feynman graph.
2. The DE of some MI only depends on MI from the same sector or subsectors.

Step 1 (preprocessing step)

Using IBP methods, we "triangularize" the system in n smaller sub-systems where the ith subsystem of coupled MI depend only on themselves and on those from the systems 1 through $i-1$.

Remarks:

1. This is more of a preprocessing step and is *per se* not included in the algorithm.
2. We will denote from now on the subsystems with a tilde, i.e. we study now a subsystem MI $\tilde{\mathcal{I}} = (\tilde{I}_1, ..., \tilde{I}_n)$ $(n \lesssim 10)$

We consider now the following subsystem

$$\frac{d}{dx}\tilde{\mathcal{I}} = \tilde{\mathcal{M}}\tilde{\mathcal{I}} + \tilde{\mathcal{R}} \tag{2}$$

Given the assumptions, we can expand:

$$\tilde{\mathcal{M}} = \sum_{j=0}^{\infty} \tilde{M}^{(j)}\epsilon^j, \quad \tilde{\mathcal{R}} = \sum_{j=-k}^{\infty} \mathcal{R}_i^{(j)}\epsilon^j, \quad \tilde{\mathcal{I}} = \sum_{j=-k}^{\infty} \mathcal{I}_i^{(j)}\epsilon^j \tag{3}$$

Step 2

By plugging (3) into (2), we can collect the powers of ϵ and we get a series of differential equation systems (DES) only in x:

$$(2) \Rightarrow \begin{cases} \frac{d}{dx}\tilde{\mathcal{I}}^{(-l)} &= & \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l)} + \tilde{\mathcal{R}}^{(-l)} & \epsilon^{-l} \\ \frac{d}{dx}\tilde{\mathcal{I}}^{(-l+1)} &= & \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l+1)} + (\tilde{\mathcal{M}}^{(1)}\tilde{\mathcal{I}}^{(-l)}) + \tilde{\mathcal{R}}^{(-l+1)} & \epsilon^{-l+1} \\ & & \vdots & \vdots \\ \frac{d}{dx}\tilde{\mathcal{I}}^{(k)} &= & \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(k)} + \left(\sum_{i=1}^{k+l}\tilde{\mathcal{M}}^{(i)}\tilde{\mathcal{I}}^{(k-i)}\right) + \tilde{\mathcal{R}}^{(k)} & \epsilon^{k} \\ & & \vdots & \vdots \end{cases} \tag{4}$$

Step 3

We solve the system iteratively in a bottom-up fashion.

To illustrate this step, let us detail a little bit what we need to do, taking the first subsystem

$$\frac{d}{dx}\tilde{\mathcal{I}}^{(-l)} = \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l)} + \tilde{\mathcal{R}}^{(-l)} \tag{5}$$

Remark: In particular, the study of this case is sufficient in the sense that for the general equation for ϵ^k, only the inhomogeneous solution changes.

– We use first the package `OreSys` in order to transform the 1st order DES into a higher order differential equation (HODE) for only one MI, i.e.

$$(5) \overset{\text{OreSys}}{\Longrightarrow} \begin{cases} \sum_{k=0}^{m} p_k(x)\partial_x^k \tilde{\mathcal{I}}_1^{(-l)}(x) = r(x) & \text{HODE in } \tilde{\mathcal{I}}_1^{(-l)}(x),\ p_i(x) \in \mathbb{K}(x) \\ r(x) = \sum_{i=0}^{p}\sum_{j=1}^{m} r_{i,j}(x)\partial_x^i \tilde{\mathcal{R}}_j^{(-l)}(x) & p \in \mathbb{N},\ r_{i,j}(x) \in \mathbb{K}(x) \\ \tilde{\mathcal{I}}_j^{(-l)}(x) = \sum_{i=0}^{m-1} a_{k,i}(x)\partial_x^i \tilde{\mathcal{I}}_1^{(-l)}(x) + \rho_k(x) & j \in \{2,\dots,m\},\ a_{k,i}(x) \in \mathbb{K}(x),\ \rho_k(x) \text{ like } r(x) \end{cases} \tag{6}$$

- We solve first the homogeneous HODE from (6)

Assumption

The HODE can be factored out in the following form (algorithms to check whether this case is possible):

$$(\partial_x - \hat{p}_1(x))(\partial_x - \hat{p}_2(x)) \cdots (\partial_x - \hat{p}_m(x))y(x) = 0 \tag{7}$$

With $\hat{p}_i(x) \in \mathbb{K}(x)$.

If this is the case, then by defining the following hyperexponential functions

$$\forall k \in \{1, \dots, m\},\ h_k(x) = e^{\int_{\ell_k}^{x} \mathrm{d}y \hat{p}_k(y)} \text{ and } \ell_k \in \mathbb{K}$$

One can check that the set of homogeneous solutions is the following vector space

$$\mathcal{S}_h = \left\{ \sum_{i=1}^{m} c_i y_i(x) \mid c_i \in \mathbb{K},\ y_i(x) = h_1(x) \int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)} \cdots \int_{\ell_{i-2}}^{x_{i-2}} \mathrm{d}x_{i-1} \frac{h_i(x_{i-1})}{h_{i-1}(x_{i-1})} \right\}$$

Remarks:

1. The function $y_i(x)$ are iterated integrals over hyperexponential functions $\frac{h_k(x)}{h_{k-1}(x)}$ and are called **d'Alembertian solutions**.
2. Given that the HODE factors in first-order factors, the function `SolveDE[diff_eqn_in_f,f[x],x]` of the package `HarmonicSums` of J. Ablinger will find the full set of solutions described above with all possible simplifications (in particular down to expressions involving harmonic polylogarithms and cyclotomic harmonic polylogarithms).

Remarks:

1. The function $y_i(x)$ are iterated integrals over hyperexponential functions $\frac{h_k(x)}{h_{k-1}(x)}$ and are called **d'Alembertian solutions**.
2. Given that the HODE factors in first-order factors, the function `SolveDE[diff_eqn_in_f,f[x],x]` of the package `HarmonicSums` of J. Ablinger will find the full set of solutions described above with all possible simplifications (in particular down to expressions involving harmonic polylogarithms and cyclotomic harmonic polylogarithms).

- The particular solution of the HODE from (6) is

$$g(x) = h_1(x)\int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)} \cdots \int_{\ell_{m-2}}^{x_{m-2}} \mathrm{d}x_{m-1} \frac{h_m(x_{m-1})}{h_{m-1}(x_{m-1})} \int_{\ell_{m-1}}^{x_{m-1}} \mathrm{d}x_m \frac{r(x_m)}{h_m(x_m)}$$

So that finally the general solution of the HODE is of the form

$$\boxed{\tilde{\mathcal{I}}_1^{(-l)}(x) = g(x) + c_1 y_1(x) + \cdots + c_m y_m(x)} \tag{8}$$

- We use physical initial conditions (IC) or results from other computations in order to fix the constants c_i.
- We use the solution (8) of the HODE as well as

$$\forall k \in \{2, \ldots, m\},\ \tilde{\mathcal{I}}_k^{(-l)}(x) = \sum_{i=0}^{m-1} a_{k,i}(x)\partial_x^i \tilde{\mathcal{I}}_1^{(-l)}(x) + \rho_k(x)$$

  From (6) in order to get all the solutions $\mathcal{I}_i^{(-l)}$ at order $k = -l$.
- We can finally plug the whole vector of solution $\tilde{\mathcal{I}}^{(-l)} = (\tilde{\mathcal{I}}_1^{(-l)}, \ldots, \tilde{\mathcal{I}}_n^{(-l)})$ into

$$\frac{d}{dx}\tilde{\mathcal{I}}^{(k)} = \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(k)} + \left(\sum_{i=1}^{k+l} \tilde{\mathcal{M}}^{(i)}\tilde{\mathcal{I}}^{(k-i)}\right) + \tilde{\mathcal{R}}^{(k)} \tag{9}$$

  For $k = -l + 1$ and reiterate the process up to the wanted k

Remark: Since only the homogeneous side of (9) changes, it suffices each time to compute only the new inhomogeneous solution g (which still requires to do most time consuming steps of simplifications of the integral of g).

Second method [arXiv:1912.04390] (Blümlein, Marquard, Schneider)

General ideas (1)

- One uncouples <u>first</u>, i.e. without expanding in ϵ, the general subsystem $\partial_x \tilde{\mathcal{I}} = \tilde{\mathcal{M}}\tilde{\mathcal{I}} + \tilde{\mathcal{R}}$ leading to a HODE equation of the form

$$\alpha_0(x,\epsilon)\tilde{\mathcal{I}}_1(x,\epsilon) + \cdots + \alpha_m(x,\epsilon)\partial_x^m \tilde{\mathcal{I}}_1(x,\epsilon) = \beta(x,\epsilon,\text{inhomogeneous part})$$

- One plugs in it an ansatz of the form

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{k=-l}^{\infty} \tilde{\mathcal{I}}_1^{(k)}(x)\epsilon^k, \quad \beta(x,\epsilon) = \sum_{k=-l}^{\infty} \beta_k(x)\epsilon^k$$

And then, provided that all α_i don't vanish at the same time (which one can guarantee), sending $\epsilon \to 0$ we get the HODE for the first coefficient only in ϵ

$$\alpha_0(x,0)\tilde{\mathcal{I}}_1^{(-l)}(x) + \cdots + \alpha_d(x,0)\partial_x^m \tilde{\mathcal{I}}_1^{(-l)}(x) = \beta_{-l}(x)$$

General ideas (2)

- One can use then once more the `SolveDE` function of `HarmonicSums` of J. Ablinger in order to solve for the $\tilde{\mathcal{I}}_1^{(-l)}(x)$ coefficient.
- Use the uncoupling equations relating the other $\tilde{\mathcal{I}}_i^{(-l)}$ to $\tilde{\mathcal{I}}_1^{(-l)}$ and its derivatives in order to get them order by order.
- Plug this result in the initial HODE, and reiterate to get all coefficients order by order.

Remark: It is also possible to expand $\tilde{\mathcal{I}}_1$ into x and not ϵ, and then work with finite recurences using C. Schneider's package `Sigma`.

Conclusion

- We presented two different methods that allow us to solve first-order DE in a rather general setting, each with its pros and cons (depending on the size of the systems, poles in ϵ, etc.)
- A completely automatic code is under development
- We will try to enhance the linear differential equation solver (or the corresponding linear difference equation solver) to tackle even more general classes of inputs