SERIES EXPANSION OF HYPERGEOMETRIC FUNCTIONS ABOUT THEIR PARAMETERS USING MultiHypExp*

Souvik Bera

Indian Institute of Science, Bengaluru, Karnataka 560012, India
souvikbera@iisc.ac.in

Received 27 December 2023, accepted 28 December 2023, published online 11 March 2024

We describe our algorithm to find the series expansion of multivariate hypergeometric functions (MHFs) in ϵ that lie in the Pochhammer parameters and its Mathematica implementation MultiHypExp.

DOI:10.5506/APhysPolBSupp.17.2-A13

1. Introduction

Multivariate hypergeometric functions (MHFs) are widespread in many branches of physics and mathematical physics. In the context of high-energy physics, the Feynman integrals can be expressed in terms of MHFs. In dimensional regularization, the dimensional parameter $\epsilon = (4-D)/2$ appears linearly in the Pochhammer parameters. On the other hand, the ratios of the scales involved in the integral take their place as the arguments of the MHFs. Such MHFs are often expressed as a Laurent series in ϵ. Thus, efficient algorithms to find the series expansion of MHFs about their parameters are necessary. Here, we discuss the algorithm proposed in [1] and its Mathematica realization MultiHypExp [2] with an example of the single variable Gauss ${}_2F_1$ function for the purpose of illustration.

2. Our algorithm

The steps of the algorithm proposed in [1] are summarized below.

— *Step 1*: Distinguish the type of series expansion (Taylor or Laurent types) by examining the Pochhammer parameters of the given MHF (say $F(\epsilon)$).

* Presented at the XLV International Conference of Theoretical Physics "Matter to the Deepest", Ustroń, Poland, 17–22 September, 2023.

— *Step 2*: Find the series expansion of $F(\epsilon)$, if it is of the Taylor type.

— *Step 3*: If the series expansion is of Laurent type, find a secondary function, say $G(\epsilon)$ that can be related to $F(\epsilon)$ by a differential operator $H(\epsilon)$ as

$$F(\epsilon) = H(\epsilon) \bullet G(\epsilon) \tag{1}$$

and $G(\epsilon)$ can be expanded in the Taylor series following *Step 2*. Here, the symbol $\bullet$ means the action of the differential operator $H(\epsilon)$ on the function $G(\epsilon)$.

— *Step 4*: Find the corresponding differential operator $H(\epsilon)$.

— *Step 5*: Perform the series expansion of the operator $H(\epsilon)$; and apply it on the Taylor expansion of $G(\epsilon)$ and collect different powers of ϵ.

We now discuss each of the steps below.

2.1. Step 1: Determination of the type of the series expansion

The series expansion of an MHF may be of Laurent series if any of the following situations appear:

1. When one or more lower Pochhammer parameters (*i.e.*, Pochhammer parameters in the denominator) are of the form: $(-p + q\epsilon)_p$;
2. When one or more upper Pochhammer parameters (*i.e.*, Pochhammer parameters in the numerator) are of the form: $(p + q\epsilon)_{-p}$,

where p is a non-negative integer. We call a Pochhammer parameter *singular* if it satisfies any of the above two conditions. Note that these are necessary but not sufficient conditions.

2.2. Step 2: Taylor expansion of MHF

In [1], the coefficients of the Taylor series expansion of the given MHF are expressed as MHFs with higher summation fold but having the same number of arguments as in the given MHF. We modify *Step 2* of the algorithm from [1] in order to express the series coefficients in terms of multiple polylogarithms (MPLs). We follow the following steps:

— At first, the set of partial differential equations (PDEs) of the given MHFs $F(\epsilon)$ is obtained. Let,

$$\boldsymbol{F} := F(\boldsymbol{a};\boldsymbol{b};\boldsymbol{x}) = \sum_{\boldsymbol{m}\in\mathbb{N}_0^r} \frac{(\boldsymbol{a})_{\mu\cdot\boldsymbol{m}}}{(\boldsymbol{b})_{\nu\cdot\boldsymbol{m}}} \frac{\boldsymbol{x}^{\boldsymbol{m}}}{\boldsymbol{m}!} = \sum_{\boldsymbol{m}\in\mathbb{N}_0^r} A(\boldsymbol{m})\boldsymbol{x}^{\boldsymbol{m}} . \tag{2}$$

The annihilators L_i of $F(\boldsymbol{a};\boldsymbol{b};\boldsymbol{x})$ are given by [3]

$$L_i = \left[h_i(\boldsymbol{\theta})\frac{1}{x_i} - g_i(\boldsymbol{\theta})\right] , \qquad (3)$$

where $\boldsymbol{\theta} = \{\theta_1, \dots, \theta_n\}$ is a vector containing Euler operators $\theta_i = x_i \partial_{x_i}$ and

$$\frac{A(\boldsymbol{m}+\boldsymbol{e_i})}{A(\boldsymbol{m})} = \frac{g_i(\boldsymbol{m})}{h_i(\boldsymbol{m})} , \qquad i = 1, \dots, n , \qquad (4)$$

where $\boldsymbol{e_i}$ is a unit vector with 1 in its i^{th} entry.

— The PDEs are brought to the Pfaffian form: $\mathrm{d}\boldsymbol{g} = \Omega \boldsymbol{g}$, where $\Omega = \sum_{i=1}^n \Omega_i \mathrm{d}x_i$ and the vector $\boldsymbol{g}$ contains the function $\boldsymbol{F}$, and its derivatives

$$\boldsymbol{g} = (\boldsymbol{F} \ \ , \ \ \boldsymbol{\theta_i} \bullet \boldsymbol{F} \ \ , \ \ \boldsymbol{\theta_i \theta_j} \bullet \boldsymbol{F}, \dots)^T . \qquad (5)$$

— The Pfaffian system is brought to the canonical form [4, 5], where the parameter ϵ is factored out: $\mathrm{d}\boldsymbol{g}' = \epsilon \Omega' \boldsymbol{g}'$.

— With a suitable boundary condition, the system is solved in order-by-order in ϵ.

2.3. Step 3: Construction of the secondary function

The secondary function $G(\epsilon)$ related to $F(\epsilon)$ can be obtained by performing the following replacements of the Pochhammer parameters of $F(\epsilon)$:

1. When one or more lower Pochhammer parameters of $F(\epsilon)$ are singular, then $(-p + q\epsilon)_p \to (1 + q\epsilon)_p$;
2. When one or more upper Pochhammer parameters of $F(\epsilon)$ are singular, then $(p + q\epsilon)_{-p} \to (q\epsilon)_{-p}$.

2.4. Step 4: The differential operator

In [6], a general algorithm based on the Gröbner basis techniques is provided by Takayama to find differential operators that relate two MHFs with Pochhammer parameters differed by integer values. For our purpose, we need the step-down operator for the lower Pochhammer parameters and the step-up operator for the upper Pochhammer parameters

$$F(\boldsymbol{a};\boldsymbol{b};\boldsymbol{x}) = \frac{1}{b_i}\left(\sum_{j=1}^{n}\nu_{ij}\theta_{x_j} + b_i\right)\bullet F(\boldsymbol{a};\boldsymbol{b}+\boldsymbol{e_i};\boldsymbol{x}) = H_-(b_i)\bullet F(\boldsymbol{a};\boldsymbol{b}+\boldsymbol{e_i};\boldsymbol{x})\,,$$

$$\begin{aligned} F(\boldsymbol{a},\boldsymbol{b},\boldsymbol{x}) &= \frac{1}{a_i-1}\left(\sum_{j=1}^{n}\mu_{ij}\theta_{x_j} + a_i - 1\right)\bullet F(\boldsymbol{a}-\boldsymbol{e_i},\boldsymbol{b},\boldsymbol{x}) \\ &= H_+(a_i)\bullet F(\boldsymbol{a}-\boldsymbol{e_i},\boldsymbol{b},\boldsymbol{x})\,. \end{aligned}$$

2.5. Step 5: Action of the differential operator

In the final step, we apply the differential operator found in *Step 4* on the Taylor expansion of $G(\epsilon)$ obtained in *Step 3*.

3. Example: Gauss ${}_2F_1$ function

Let us consider the following Gauss hypergeometric functions, whose series expansion we wish to find:

$$F(\epsilon) :=\ {}_2F_1(\epsilon,-\epsilon;\epsilon-1;x) = \sum_{m=0}^{\infty}\frac{(\epsilon)_m(-\epsilon)_m}{(\epsilon-1)_m}\frac{x^m}{m!}\,. \tag{6}$$

We notice that the lower Pochhammer parameter of $F(\epsilon)$ is singular. Therefore, the series expansion may be of Laurent type. Therefore, we construct the secondary function. The secondary function $G(\epsilon)$, which is related to $F(\epsilon)$ and expandable in the Taylor series, can be found by replacing the singular Pochhammer parameter,

$$G(\epsilon) :=\ {}_2F_1(\epsilon,-\epsilon;\epsilon+1;x) = \sum_{m=0}^{\infty}\frac{(\epsilon)_m(-\epsilon)_m}{(\epsilon+1)_m}\frac{x^m}{m!}\,. \tag{7}$$

Next, we go on to find the Taylor series expansion of $G(\epsilon)$. By constructing a vector $\boldsymbol{g} = (G(\epsilon),\theta_x G(\epsilon))^T$ and making good use of the ordinary differential equation of Gauss ${}_2F_1$, we obtain the following Pfaffian system:

$$\mathrm{d}\boldsymbol{g} = \Omega\boldsymbol{g} \quad , \quad \Omega = \begin{pmatrix} 0 & \frac{1}{x} \\ \frac{\epsilon^2}{x-1} & \frac{\epsilon}{x-1}-\frac{\epsilon}{x} \end{pmatrix}. \tag{8}$$

Further, the Pfaffian system can be brought to the canonical form by the transformation matrix, which can be obtained using CANONICA [7]

$$T = \begin{pmatrix} 1 & 0 \\ 0 & \epsilon \end{pmatrix} \tag{9}$$

which reads

$$\mathrm{d}\boldsymbol{g}' = \epsilon \Omega' \boldsymbol{g}' \,, \qquad \Omega' = \begin{pmatrix} 0 & \frac{1}{x} \\ \frac{1}{x-1} & \frac{1}{x-1} - \frac{1}{x} \end{pmatrix} . \tag{10}$$

This system can now be solved order by order in ϵ with the boundary condition given by: $\boldsymbol{g}(x=0) = (1,0)^T$.

Thus, we find the Taylor series expansion of $G(\epsilon)$ as

$$G(\epsilon) = 1 + \epsilon^2 G(0,1;x) + \epsilon^3(-G(0,0,1;x) + G(0,1,1;x)) + O\left(\epsilon^4\right) \,, \tag{11}$$

where Gs are the MPLs [8, 9] defined as

$$G(a_1, \ldots, a_n; z) = \int_0^z \frac{dt}{t - a_1} G(a_2, \ldots, a_n; t) \tag{12}$$

with $G(z) = 1$, and a_i and z being complex-valued variables.

Next, we obtain the differential operator that relates the two Gauss hypergeometric functions: $F(\epsilon) = H(\epsilon) \bullet G(\epsilon)$.

The required differential operator that relates the two Gauss hypergeometric functions can be easily obtained

$$H(\epsilon) = \frac{(\epsilon(2x-1) - x + 1)}{(\epsilon - 1)\epsilon(x-1)} \theta_x + \frac{\epsilon(2x-1) - x + 1}{(\epsilon - 1)(x-1)} .$$

Finally, we apply the differential operator $H(\epsilon)$ on the Taylor expansion of $G(\epsilon)$ to find the series expansion of $F(\epsilon)$

$$F(\epsilon) = 1 + \epsilon \left[G(1;x) - \frac{x}{x-1} \right] + \epsilon^2 \left[-\frac{x}{x-1} G(1;x) + G(1,1;x) - \frac{x}{x-1} \right] .$$

The result is consistent with the result obtained using the HypExp [10] package.

4. MultiHypExp package

We now discuss the usage of the two commands of the package MultiHypExp, which can be downloaded from the following url: `https://github.com/souvik5151/MultiHypExp`

It is suitable for Mathematica v11.3 and beyond. The package depends on the following packages: HolonomicFunctions [11, 12], CANONICA [7], HYPERDIRE [13–15], PolyLogTools [16], HPL [17, 18] .

These dependencies must be called before loading the package MultiHypExp. The usages and the implementation of these packages are discussed in detail in [2].

The package consists of two commands:

— `SeriesExpand`: To find the series expansion of certain MHFs about their integer-valued Pochhammer parameters;

— `ReduceFunction`: To find the reduction formula of certain MHFs in terms of MPLs.

We give examples of the usage of these commands below.

The first three coefficients of the series expansion of Appell $F_3(\epsilon,\epsilon,\epsilon,1+\epsilon;1+\epsilon;x,y)$ about the parameter ϵ can be obtained by calling the `SeriesExpand` command in the following way:

```
In[]:= SeriesExpand[F3,{e,e,e,1+e,1+e},{x,y},e,3]
```

In the standard form, the output of the above command reads

$$F_3(\epsilon,\epsilon,\epsilon,1+\epsilon;1+\epsilon;x,y) = 1 - G(1,y)\epsilon + (-G(0,1,x) + G(1,1,y))\epsilon^2$$

```
In[]:= ReduceFunction[F3,{1,1,2,2,3},{x,y}]
```

To find the reduction formula of Appell $F_3(1,1,2,2;3;x,y)$ in terms of MPLs, we call the `ReduceFunction` command as above. The output can be written in the standard form, after converting MPLs to ordinary logarithms as

$$F_3(1,1,2,2;3;x,y) = -\frac{2(-xy + x + y + \log(1-x) + \log(1-y))}{(x+y-xy)^2}\,.$$

5. Conclusion

We have presented an efficient algorithm to find a series expansion of MHFs and discussed its implementation in Mathematica. We conclude with some remarks. The algorithm is applicable when the parameter ϵ appears linearly inside the Pochhammer parameters of the MHFs. Furthermore, the series expansion of a given MHF is not applicable to its singular locus. The package is capable of finding the series expansion of certain MHFs around integer values of the Pochhammer parameters. Sometimes the MHFs are needed to series-expand around rational values of parameters, which requires some non-trivial change of variables (see [19]) or may require functions beyond MPLs. We plan to explore the possibilities of finding a series expansion of MHFs for such cases.

REFERENCES

[1] S. Bera, *Nucl. Phys. B* **989**, 116145 (2023), arXiv:2208.01000 [math-ph].

[2] S. Bera, *Comput. Phys. Commun.* **297**, 109060 (2024), arXiv:2306.11718 [hep-th].

[3] H. Bateman, «Higher Transcendental Functions», 1953.

[4] J.M. Henn, *Phys. Rev. Lett.* **110**, 251601 (2013), arXiv:1304.1806 [hep-th].

[5] J.M. Henn, *J. Phys. A: Math. Theor.* **48**, 153001 (2015), arXiv:1412.2296 [hep-ph].

[6] N. Takayama, *Japan J. Appl. Math.* **6**, 147 (1989).

[7] C. Meyer, *Comput. Phys. Commun.* **222**, 295 (2018), arXiv:1705.06252 [hep-ph].

[8] A.B. Goncharov, arXiv:math.AG/0103059.

[9] A.B. Goncharov, arXiv:1105.2076 [math.AG].

[10] T. Huber, D. Maître, *Comput. Phys. Commun.* **175**, 122 (2006), arXiv:hep-ph/0507094.

[11] C. Koutschan, *ACM Commun. Comput. Algebra* **43**, 119 (2010).

[12] C. Koutschan, *Math. Comput. Sci.* **4**, 259 (2010).

[13] V.V. Bytev, M.Y. Kalmykov, B.A. Kniehl, *Comput. Phys. Commun.* **184**, 2332 (2013), arXiv:1105.3565 [math-ph].

[14] V.V. Bytev, M.Y. Kalmykov, S.-O. Moch, *Comput. Phys. Commun.* **185**, 3041 (2014), arXiv:1312.5777 [math-ph].

[15] V.V. Bytev, B.A. Kniehl, *Comput. Phys. Commun.* **189**, 128 (2015), arXiv:1309.2806 [math-ph].

[16] C. Duhr, F. Dulat, *J. High Energy Phys.* **2019**, 135 (2019), arXiv:1904.07279 [hep-th].

[17] D. Maître, *Comput. Phys. Commun.* **174**, 222 (2006), arXiv:hep-ph/0507152.

[18] D. Maître, *Comput. Phys. Commun.* **183**, 846 (2012), arXiv:hep-ph/0703052.

[19] M.A. Bezuglov, A.I. Onishchenko, arXiv:2312.06242 [hep-th].