JOHANNES KEPLER
UNIVERSITY LINZ

Author
Nikolai Fadeev
11947983

Submission
Research Institute for Symbolic Computations

Supervisor and First Evaluator
Univ.-Prof. Dr.
Carsten Schneider

Second Evaluator
Prof. Dr.
Johannes Blümlein

May 2025

Computer algebra for special functions

Doctoral Thesis

to confer the academic degree of

Doktor der Naturwissenschaften

in the Doctoral Program

Naturwissenschaften

JOHANNES KEPLER UNIVERSITY LINZ
Altenberger Straße 69
4040 Linz, Austria
jku.at

Abstract

Calculations done in different mathematical areas — such as computer algebra, combinatorics, number theory, differential equations — and physical areas — such as particle physics — give rise to a plethora of problems involving special functions that need to be dealt with efficiently. In this PhD, we concentrated on two such particular problems.
In the first part of this PhD thesis, we explored the relation between iterated binomial sums, an extension of general harmonic sums, and their integral representations, in order to compute their asymptotic expansions. To do that in a fully automatic way, we created a dedicated package, `RICA`. Using Mellin representations, we first formalised and extended a classical recursive method to compute Mellin inverses of such sums, and together with it implemented several methods to compute asymptotic expansions of such integrals. In the process, we introduced and explored a new class of functions related to Mellin convolutions. Those allowed us to automatically compute asymptotic expansions for more general classes of sums in a new and efficient way, while providing a way to get symbolic representations for the constants appearing in the calculation of the Mellin inversions.
In the second part of this PhD thesis, we studied first order inhomogeneous systems of differential equations involving an extra parameter ϵ coming from particle physics computations. Since usually those systems could only be solved up to some order in ϵ, we aimed at developing a method to optimise the solving task of such systems. We studied an approach centered on the minimisation of the ϵ order in the expansion of the inhomogeneous part. In particular, we proposed a method based on separating the system in smaller subsystems called triangularization, before analysing each of those individually using different uncoupling schemes, selected priorization of equations and through comparisons of ϵ-orders. This method has been implemented in a package called `SystemAnalysis`.

Kurzfassung

Berechnungen in verschiedenen mathematischen Bereichen — wie Computeralgebra, Kombinatorik, Zahlentheorie, Differentialgleichungen — und physikalischen Bereichen — wie der Teilchenphysik — führen zu einer Fülle von Problemen mit speziellen Funktionen, die effizient bearbeitet werden müssen. In dieser Dissertation haben wir uns auf zwei dieser speziellen Probleme konzentriert.
Im erste Teil der Doktorarbeit untersuchten wir die Beziehung zwischen iterierten Binomialsummen, einer Erweiterung der allgemeinen harmonischen Summen, und ihrer Integraldarstellungen, um ihre asymptotischen Erweiterungen zu berechnen. Um dies zu automatisieren, haben wir ein spezielles Paket, `RICA`, entwickelt. Unter Verwendung von Mellin-Darstellungen haben wir zunächst ein klassisches rekursives Verfahren zur Berechnung von Mellin-Inversen solcher Summen formalisiert und erweitert und zusammen mit diesem mehrere Methoden zur Berechnung asymptotischer Expansion solcher erweiterten Integrale implementiert. Dabei haben wir eine neue Klasse von Funktionen eingeführt und erforscht, die mit Mellin-Faltungen zusammenhängen. Diese erlaubten es uns, automatisch asymptotische Erweiterungen für allgemeinere Klassen von Summen auf eine neue und effiziente Weise zu berechnen und gleichzeitig eine Möglichkeit zu schaffen, symbolische Darstellungen für die Konstanten zu erhalten, die bei der Berechnung der Mellin-Inversionen auftreten.
Im zweite Teil der Doktorarbeit untersuchten wir inhomogene Systeme von Differentialgleichungen erster Ordnung, die einen zusätzlichen Parameter ϵ enthalten, der aus Berechnungen der Teilchenphysik stammt. Da diese Systeme in der Regel nur bis zu einer gewissen Ordnung in ϵ gelöst werden können, haben wir eine Methode entwickelt, um die Lösung solcher Systeme zu optimieren. Wir untersuchten einen Ansatz, der sich auf die Minimierung der ϵ-Ordnung in der Expansion des inhomogenen Teils konzentriert. Insbesondere haben wir eine Methode vorgeschlagen, die auf der Aufteilung des Systems in kleinere Teilsysteme, die sogenannte Triangularisierung, basiert, bevor jedes dieser Teilsysteme einzeln mit Hilfe verschiedener Entkopplungsschemata, ausgewählter Priorisierung von Gleichungen und durch Vergleiche der ϵ-Ordnungen analysiert wird. Diese Methode wurde in einem Paket namens `SystemAnalysis` implementiert.

Acknowledgements

Throughout my thesis, I have received tremendous help from many people and without them I would never have managed to finish it.

First and foremost I would like to deeply thank Carsten Schneider, who has been the best advisor I could ask for. He gave me wonderful projects that I thouroughly enjoyed working on. His deep insight on many subjects always helped me to distinguish what was more relevant to concentrate on, when I had several possible research directions and was unsure about which one to choose, and his great kindness was an irreplacable moral support during difficult times.

I would also like to sincerely thank Johannes Blümlein, without whom this thesis probably would not have even started, and who was an equally wonderful advisor. Working with him at DESY Zeuthen was a great honor. His extremely vast knowledge, profound understanding of interactions between physics and mathematics, and precise guidance were invaluable to help me better understand the intricacies of asymptotics and their applications in particle physics, and better organise myself to reach objectives more efficiently.

During my time at Wolfram, Devendra Kapadia always guided me with precision and kindness in my project, and Oleg Marichev taught me many mathematical subjects that broadened my horizons and gave me new ideas — and for that, I am very thankful to both of them. I would like to express my gratitude to Lorenzo Quintavalle, Manuel Accettulli Huber, Luke Corcoran, Stefano de Angelis, with whom we had great fun working on the different projects at Wolfram, and discovering together the American "way of life" and culture. Together with them, I am grateful to all the other SAGEX people who I met during those years and who helped me one way or another broaden my horizons, from the scientific and the human side.

I am also extremely thankful to Jakob Ablinger and Silviu Radu, with whom I had many meaningful discussions and work sessions. Jakob has always had great insights, he supported and helped me in moments of doubt, and Silviu shaped my approach to research in mathematics in general thanks to his great pedagogical talent. Working with them has always been a pleasure.

Throughout all my years there, I have come to see RISC as a second home, and for that I would also like to thank Abilio de Freitas, Georg Ehling, Jakob Obrovsky, Ralf Hemmecke, Teimuraz Kutsia and many others with whom we had exciting discussions about mathematics, physics, computer science, politics or even more metaphysical questions during coffee breaks. I am grateful to Tanja Gutenbrunner and Ramona Oehme-Pöchinger who were always there whenever some administrative problem popped up, and helped me solve them very efficiently. Last but not the least, my deepest and sincerest gratitude goes to Ralf Wahner not only for his help whenever some technical problems arose, but more particularly for his friendliness, kindness and tremendous support in most difficult times.

I would like also to thank all the great people that I had the opportunity to meet during my travels to conferences abroad or in Austria, such as Ali Uncu, Christoph Koutschan, Pierrez Lairez and many others, with whom we had great interactions and discussions about mathematics and many other things.

Of course, I would like to thank all my friends who were always there for me and supported me throughout this long journey, in particular Richard, Michaël, Melan, Louis, Aurélie, Axel-Adrien, Anthony, Guillaume, Adrien, Jérémie, Valentin, Ben, Evelina, Lisa, Andriana, Elaine, Rodrigo and all the others. On a more funnier note, my thanks also go to Ryuukishi, who created stories and characters that fascinated many generations, and gave me at least partly inspiration for the name of the `RICA` package.

Finally, I would not be who I am without my mother, my father, my two brothers and my grand-mother; without their tremendous support I would simply not be here now. I want to dedicate this thesis to my father, who left us recently, but whose soul and creative spirit never did. Since my young age, he taught me to see beauty in the world, be it spiritual or physical, and the original reason I came to be interested in mathematics in middle school was because I aesthetically liked the symbol of integration, $\int$. Dostoevsky said in the *Idiot* that "Beauty will save the world!", and it is therefore probably not a coincidence that my thesis involved studies of special functions, so heavily relying on it. I am sure that up above, where all the beauty and love of this world come from, this fact brings a smile to my father's face.

This PhD thesis was supported by the European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Grant Agreement No. 764850 'SAGEX' and the Austrian Science Fund (FWF) grant 10.55776/P33530.

Contents

1. Introduction **5**

2. Asymptotics and special functions **9**

2.1. From harmonic to binomial sums 10

2.1.1. Basic properties of the Mellin transform 15

2.1.2. Polylogarithms, harmonic polylogarithms and generalized polylogarithms 21

2.2. Methods to compute Mellin convolutions 32

2.2.1. General convolution properties of Mellin transforms 32

2.2.2. Harmonic sums and plus-prescriptions 36

2.3. A simple recursive algorithm for Mellin inversion of iterated sums . 46

2.3.1. The classical method versus our recursive method 47

2.3.2. The general method 49

2.3.3. An example with RICA 58

2.3.4. Generalized convolution polylogarithms 62

2.3.5. The problem of extended convolution multiplication formulas 69

2.4. A theoretical overview of asymptotics and series expansions 79

2.4.1. About series expansions and asymptotics 80

2.4.2. Integration and asymptotics 88

2.4.3. A note about singular integrands 111

2.5. Computing asymptotic expansions using Mellin representations . . . 112

2.5.1. Factorial series approach (FSA) 114

2.5.2. Geometric series approach (GSA) 117

2.5.3. Change of variables (COV) 120

2.5.4. Asymptotics and GCPL functions 135

2.5.5. Asymptotics and numerics 155

2.5.6. Some examples of asymptotics 159

3. An overview to solve first order systems of differential equations **167**

3.1. Solving hypotheses 168

3.2. Triangularization 171

3.3. Solving iteratively the system 172

3.3.1. Defining recursive sub-systems 172

3.3.2. Uncoupling the system 173

3.3.3. Solving the higher-order differential equation for the order k 174
3.3.4. Solving the system at order k and iterating . . . 177
3.4. Solving iteratively the system using the Sigma package . . . 178
3.4.1. Further assumptions on the subsystem . . . 178
3.4.2. Uncoupling the system . . . 179
3.4.3. Order by order solving . . . 180
3.4.4. Solving for the other master integrals and iterating . . . 185
3.4.5. Alternative application of the method . . . 185
3.4.6. The large moments method . . . 186

4. Optimising the analysis of differential systems 189
4.1. Preprocessing the system . . . 190
4.2. How to proceed . . . 194
4.3. Selecting the equations to solve for an irreducible system . . . 194
4.3.1. Criterion for optimal uncoupling . . . 194
4.3.2. Uncoupling the system and selecting the best equations — a naive approach . . . 196
4.3.3. Refining the uncoupling scheme . . . 209
4.3.4. Recurrences as selection scheme . . . 220
4.4. Computing the corrections . . . 221
4.4.1. ϵ-order correction . . . 224
4.4.2. Equation selection method and correction computing . . . 226
4.5. Global corrections and trees . . . 229
4.5.1. Method of combining the subsystem corrections . . . 233

5. Conclusion 239

A. The RICA package, a userguide 243
A.1. Basic objects and transformations . . . 243
A.1.1. Hwb . . . 243
A.1.2. ConvertGLbToGL . . . 244
A.1.3. ConvertGLToGLb . . . 244
A.1.4. ConvertHwb . . . 245
A.1.5. GLbToGLCb . . . 245
A.1.6. GLbToHwb . . . 246
A.1.7. GLCb . . . 247
A.1.8. GLCbToIntegral . . . 247
A.1.9. GSToHSS . . . 249
A.1.10. ReduceGLCb . . . 250
A.1.11. TransformRootGL . . . 251

A.2. Mellin convolutions . . . 252
A.2.1. ExtendedMellinConvolve . . . 252
A.2.2. SumToMellin . . . 253
A.3. Asymptotic expansions . . . 254
A.3.1. AsymptoticsMellint . . . 254
A.3.2. AsymptoticsSum . . . 255
A.3.3. GLCbSeries . . . 256
A.4. Useful functions . . . 257
A.4.1. FindPSLQRelations . . . 257
A.4.2. GLCbToLatex . . . 258
A.4.3. VerifyExpansion . . . 259

B. Implementation of the analysis of differential systems **261**
B.1. Using SystemAnalysis . . . 261
B.2. Examples . . . 264

Conventions

We introduce here some general conventions that will be used throughout this text:

- To denote a **derivative with respect to the variable** x, we will use either $\frac{\mathrm{d}}{\mathrm{d}x}$, $(\bullet)'$ or the more compact d_x notation.
- $\mathbb{N}$ denotes the **set of non negative integers**, for $a \in \mathbb{N}$, $\mathbb{N}_{\geq a} := \mathbb{N} \backslash \{0, \ldots, a\}$ and the special case $\mathbb{N}^\star := \mathbb{N}_{\geq 1}$ is the set of **positive integers**.
- $\mathbb{Z}$ denotes the **set of all integers**; for $a \in \mathbb{Z}$, $\mathbb{Z}_{\geq a} := \{a, a+1, \ldots, \}$, $\mathbb{Z}_{\leq a} := \{\ldots, a-1, a\}$, and $\mathbb{Z}^\star := \mathbb{Z} \backslash \{0\}$.
- $\mathbb{K}$, $\mathbb{F}$ will denote **computable fields**.
- We will denote with $\mathbb{K}[X]$ the **ring of polynomials over** the field $\mathbb{K}$ and with $\mathbb{K}(X)$ the **field of rational functions over** the field $\mathbb{K}$.
- If $q \in \mathbb{K}[X]$, we denote the **set of roots** of q with $V(q) \subset \bar{\mathbb{K}}$, where $\bar{\mathbb{K}}$ denotes the **algebraic closure** of the field $\mathbb{K}$ (with $\bar{\mathbb{R}} = \mathbb{C}$ as specific case).
- We will use $\mathbb{R}$ to denote **real numbers** and $\mathbb{C}$ to denote **complex numbers**.
- We denote with $C_n(a, r)$, where $n, r > 0$ and $a \in \mathbb{K}^n$, the **n-sphere** of center a and radius r. In particular C_1 is a circle and C_2 is the standard sphere.
- We will denote with sign(xà the **sign function**, of a real number x, i.e.,

$$\operatorname{sign}(x) := \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}.$$

- We will denote with $(a)_n$ or $(a)_{\overline{n}}$ the **rising factorial** or **Pochhammer symbol**, i.e.,

$$(a)_{\overline{n}} = (a)_n := a(a+1)\cdots(a+n-1), \quad (a)_{\overline{0}} = (a)_0 := 1$$

- We will denote with $(a)^{\underline{n}}$ the **falling factorial**, i.e.,

$$(a)^{\underline{n}} := a(a-1)\cdots(a-n+1), \quad (a)^{\underline{0}} := 1$$

- We will denote the **set of hyperexponential functions over** $\mathbb{K}$ with $\mathcal{H}^e(\mathbb{K})$ (for more details see Definition 3.1.1).
- For an **alphabet** $\mathcal{A} = \{f_1, \ldots, f_n\}$ with **letters** f_i, we write a **word** $\mathcal{A}$ as $\vec{w} = f_{i_1}, \ldots, f_{i_j}$, $j \in \mathbb{N}^\star$. (see Definition 2.1.1).
- We denote the **generalized polylogarithms** with $\mathrm{H}_{\vec{w}}(x)$, where $\vec{w}$ is a word, if the iterated integrals are defined with 0 as a lower bound, and $\mathrm{H}^*_{\vec{w}}(x)$ if the iterated integrals are defined with 1 as a the higher bound (see Definition 2.1.13).
- We denote the **Mellin transform of a function** $f(x)$ **at the point** $n \in \mathbb{N}$ with $\mathbf{M}\left[f(x)\right](n)$ (for more details see Definition (2.1.8)).
- We denote the **Mellin transform of a function** $f(x)$ **in the +-prescription at the point** $n \in \mathbb{N}$ with $\mathbf{M}^+\left[f(x)\right](n)$ or $\mathbf{M}\left[[f(x)]_+\right](n)$ (for more details see Definition 2.1.23).
- We denote the **Mellin convolution of two functions** $f(x)$ **and** $g(x)$ with $f(x) \star g(x)$ (for more details see Definition (2.1.28)).
- If $f(x)$ is a function, we denote the **iterated Mellin convolution** of $f(x)$ with itself p-times as $f(x)^{\star p}(x)$, i.e.,

$$\begin{aligned} f(x)^{\star 0}(x) &:= 1, \\ f(x)^{\star 1}(x) &:= f(x), \\ f(x)^{\star p}(x) &:= f(x) \star f(x)^{\star (p-1)}. \end{aligned}$$

- We denote a function whose p^{th}-derivative is continuous in its definition domain with $\mathcal{C}^k$ and say it is of **class** $\mathcal{C}^p$. A function that is simply continuous on its definition domain is $\mathcal{C}^0$.
- The **Riemann zeta-function** ζ is defined for $s \in \mathbb{C}$ such that $\mathrm{Re}(s) > 1$ as

$$\zeta(s) := \sum_{n=1}^{\infty} \frac{1}{n^s}$$

  and its analytic continuation elsewhere on the complex plane.
- $\zeta_n \in \mathbb{R}$ denotes the **values of the Riemann zeta-function at** $n \in \mathbb{N}_{\geq 2}$.
- $\mathbf{a}_p$, $a \in \mathbb{K}$, denotes the **sequence** $\underbrace{a, \ldots, a}_{p \text{ times}}$.

- The **Heaviside function** Θ is defined as:

$$\Theta(x) := \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} .$$

- If T is a **distribution** and φ a test function from the associated function space, we denote T acting on φ as $\langle T, \varphi \rangle$. If T is induced by a smooth locally integrable function f over $U \subseteq \mathbb{R}$ then, unless stated otherwise, its action on φ will be defined as:

$$\langle T, \varphi \rangle := \int_U \mathrm{d}x \, f(x)\varphi(x).$$

- The **Dirac δ-distribution**, also sometimes called **Dirac δ-function** is defined as a distribution that acts on a proper test function f, typically a smooth function over $\mathbb{R}$ with compact support, in the following way:

$$\langle \delta, f \rangle := f(0) = \int_{-\infty}^{+\infty} \mathrm{d}x \, \delta(x) f(x)$$

  where the integral representation is a convenient abuse of notation.

- We denote with $\delta_{i,j}$ the **Kronecker delta** function defined by

$$\delta_{i,j} := \begin{cases} 0 & \text{if} \quad i \neq j \\ 1 & \text{if} \quad i = j \end{cases} .$$

- We denote the **cyclotomic polynomial of order** n with $\Phi_n(x)$ (see (2.1.91)).

- We represent **hypergeometric functions** (PFQ) as ${}_pF_q$ and **regularized hypegeometric functions** as ${}_p\tilde{F}_q$ (see Defintion 2.3.5).

We also provide below a list of the common abreviations that we will use:

- **BI**: base (case) integral (see p.168)
- **CM**: coefficient matrix (see p.4)
- **FBI**: full base (case) integral (see p.229)
- **GCPL**: general convolution polylogarithm (see Definition 2.3.3)
- **GPL**: general polylogarithm (see p.21)
- **GRPL**: general root polylogarithm (see Definition 2.1.13)
- **HODE**: higher order differential equation (see p.173)
- **HPL**: harmonic polylogarithm (see Definition 2.1.15)
- **IBI**: induced base (case) integral (see Remark 4.1.2)
- **IBP**: integration by parts (see p.167)
- **IOH**: iterated integral over hyperexponential functions (see Definition 3.1.2)
- **LF**: linear factor (see p.195)
- **LHS**: left-hand side (of an, equation or relation in general) (see p.70)
- **MI**: master integral (see p.168)
- **MIV**: master integral vector (see p.189)
- **RHS**: right-hand side (of an equation or relation in general) (see p. 189)
- **RHSV**: right-hand side vector (see p.189)
- **RMI**: root master integrals (see Definition 4.5.1)

1. Introduction

When Augustine of Hippo, better known as Saint Augustine, asked himself in the *Confessions* the question "*What is time?*", his own answer was "*If nobody asks me, I know; but if I were desirous to explain it to one that should ask me, plainly I do not know.*". While luckily for us the statement does not fully transpose, it still holds some tinge of truth when one replaces the name *time* with *special functions*. The rapid development of mathematics in the 18^{th} and 19^{th} centuries prompted researchers to coin the term for new classes of functions that were not related to polynomial, trigonometric, hyperbolic functions and their inverses. However, starting with only certain solutions of differential equations appearing in physics — such as the Airy functions in optics, the sine integral in signal processing or spherical harmonics for quantum mechanics among many others —, their span became much larger and prone to changes as boundaries between mathematical fields became fuzzier and new connections were made.

The aim of this PhD thesis is the development of computer algebra methods to explore special functions that arise in the connections of particle physics calculations and related problems, e.g., in combinatorics or number theory. The first part will be addressed in Chapter 2 of the thesis. The second part of the thesis covered in Chapters 3 and 4 pushes further and refines available tools to solve coupled systems of ordinary differential equations that arise in the context of calculating Feynman diagrams. In the following we give a general overview of the different parts and highlight the main contributions.

Chapter 2: In computer algebra, the study of nested sums done by Schneider using difference field theory [1] and recurrence solving [2–6] led to connections between the recurrences setting and the differential equations setting, in particular connecting harmonic sums and harmonic polylogarithms. The subject was studied already by Blümlein and Kurth [7], Remiddi and Vermaseren [8] and later generalized by Ablinger, Blümlein and Schneider [9–13], with extensions to generalized and cyclotomic polylogarithms as well as root-valued polylogarithms. On the sum side, a study of generalized harmonic sums, also denoted as S-sums has been done in particular by Moch, Uwer and Weinzierl in [14] and, giving detailed proofs, by Ablinger, Blümlein and Schneider in [12]. More extended classes, including in particular the binomial sums, were already discussed in physics contexts by Kalmykov

and Veretin in [15] for infinite sums and substantially extended by Ablinger, Blümlein, Raab and Schneider in [16]. One idea to connect sum representations and integral representations, i.e., specific classes of special functions, was to use the Mellin transform [17,18], to relate recurrences and differential equations. More recently, in [13] Behring, Blümlein and Schönwald also proposed a method using computation of generating functions and complex analysis techniques and differential equations. But in [16] and also [19], Ablinger, Blümlein, Raab and Schneider took a very different approach that did not rely on the differential equations setting to tackle new classes of sums called binomial sums (Section 2.1) due to their relevance in particle physics, as seen already at two loops for self-energy diagrams in [15,20,21] and then more widely at three loops [22,23]. This class of sums, characteristic for its $\binom{2n}{n}$ central binomial coefficient, has also appeared in combinatorial problems, such as in [24] where Krattenthaler and Schneider have studied certain classes of double nested sums containing binomial coefficients and absolute values. More general binomial sums in the combinatorics setting have also been studied in [25–27]. Finite binomial sums became our first object of study. In Chapter 2, we will concentrate on the method introduced in [16,19]. After formalising it properly, we implemented it in a Mathematica package called `RICA`, for **R**ule **I**nduced **C**onvolutions for **A**symptotics. The aim was not only to be able to automatically represent certain classes of sums as specific integrals (Section 2.3), but also to be able to compute asymptotic expansions of those to enable further studies (Section 2.5), such as analytic continuations of the sums to the complex plane. Asymptotic expansions of finite binomial sums were considered in references [16,28] before. They rely on partial integrations, independently of the alphabet, see also [29]. In order to represent sums as integrals, we used once again Mellin representations and the method that relies on recursive computations of Mellin convolutions of the summands of the iterated binomial sums sketched in [16], that we will properly detail in Sections 2.2 and 2.3. For asymptotics, several methods have been implemented, among which there is a rather general approach that relies on change of variables, and term by term integration enabled by Watson's lemma [16,30] and extended versions (Section 2.5.3). In the midst of doing so, we will define new classes of special functions, the generalized convolution polylogarithms (Section 2.2.1), and elaborate that those can be used in an efficient and simple way to tackle asymptotic computations of not only binomial sums introduced in [16] but even more extended classes of sums (Section 2.5.4). In particular, we will be able to perform inversion and asymptotic expansions of binomial sums such as those appearing in [16], for instance:

$$\sum_{i=1}^{n}\binom{2i}{i}(-2)^i\sum_{j=1}^{i}\frac{\left(\frac{1}{2}\right)^j}{j}\sum_{k=1}^{j}\frac{1}{k^2}\underset{n\to+\infty}{=}\frac{5(-8)^n\zeta_3}{9\sqrt{\pi}\sqrt{n}}-\frac{65(-1)^n2^{3n-3}\zeta_3}{81\sqrt{\pi}n^{3/2}}-\frac{65(-1)^n2^{3n-7}\zeta_3}{243\sqrt{\pi}n^{5/2}}+\cdots$$

In general the sums treated in [16] cover only two central binomial coefficients letters. Nested binomial sums with more than two binomial coefficients did not

appear in physical computations yet, even though the convolution formulas given in [16] allow for more binomial terms. Utilizing our generalized convolution polylogarithms we succeed in tackling more general sums involving more than two central binomial coefficients, such as

$$\sum_{i=1}^{n}\binom{2i}{i}\sum_{j=1}^{i}\frac{2^j}{j\binom{2j}{j}}\sum_{k=1}^{j}\binom{2k}{k}\underset{n\to+\infty}{=}\frac{2^{3n+6}}{21\sqrt{\pi}n^{3/2}}+\frac{211\ 2^{3n+3}}{441\sqrt{\pi}n^{5/2}}+\frac{76739\ 2^{3n-1}}{3087\sqrt{\pi}n^{7/2}}+\cdots$$

Finally, in [11] higher powers of central binomial coefficients have been considered, linked to elliptic structures. These are incomplete elliptic integrals, still maintaining first order factorizable iterative objects on the sum side and partially on the integral side. Due to our new approach we are able to perform asymptotic expansions of such objects without having to deal with explicit incomplete elliptic function analysis. It is expected that applications exist in which sums like

$$\sum_{i=1}^{n}\binom{2i}{i}\sum_{j=1}^{i}\binom{2j}{j}^2\underset{n\to+\infty}{=}\frac{2^{6n+10}}{945\pi^{3/2}n^{3/2}}-\frac{239\ 2^{6n+7}}{99225\pi^{3/2}n^{5/2}}+\cdots \tag{1.0.1}$$

$$\sum_{i=1}^{n}\binom{2i}{i}^2\sum_{j=1}^{i}\frac{1}{j\binom{2j}{j}}\underset{n\to+\infty}{=}\frac{2^{4n+4}}{45\sqrt{3}n}-\frac{11\ 2^{4n+2}}{675\sqrt{3}n^2}+\cdots \tag{1.0.2}$$

$$\sum_{i=1}^{n}\binom{2i}{i}^5\sum_{j=1}^{i}\frac{1}{j^2\binom{2j}{j}^2}\sum_{k=1}^{j}\binom{2k}{k}^3\underset{n\to+\infty}{=}\frac{\pi^5 2^{12n+8}}{773955n^5}+\frac{71\pi^5 2^{12n+6}}{126773829n^6}+\cdots \tag{1.0.3}$$

will play an important role. We remark that these sums could not be tackled with the differential equation approach given in [17, 18]. Alternatively, one may have used the recurrence approach to compute a basis of the space of asymptotic expansions [31, 32]. However, this approach has difficulties to determine the appearing constants. More precisely, it is very hard to extract only the first numerical digits of them. In our approach the constants can be given symbolically in terms of integrals that often can be simplified or guessed by PSLQ methods [33, 34] to special constants such as $\sqrt{3}$ or $\zeta_3=\sum_{n=1}^{\infty}\frac{1}{n^3}$ as given above. If this is not possible, the constants can be determined at least to very high precision.

Chapter 3: In particle physics, trying to probe matter leads frequently to interesting problems involving diverse special functions. In particular, trying to compute how particles interact leads often to studies of differential equations or more generally, of inhomogeneous systems of differential equations, where the variables in which we want to solve are called master integrals [35–42]. In Chapter 3 we review different classical methods to solve first order systems of differential equations. Due to physical reasons, the systems often involve one extra "small" parameter ϵ

and cannot be solved analytically where ϵ arises inside of the special functions. In some cases a basis transformation of the system enables one to read off a solution in form of an ϵ-expansion where the coefficients are given explicitely in terms of special functions [43–48]. If this fails, two alternative very flexible methods have been elaborated. One then either can expand in this parameter and then uncouple them [48][1] (Section 3.3) or uncouple directly those systems and then solve order by order in ϵ [49, 50] (Section 3.4). Both approaches have been succesfully applied in many physics problems [22, 28, 51–60]. Because both methods rely on differential and recurrence equation solving, in the case it cannot be achieved, a method called the large moment method, has been introduced in [61–63] and applied for example in [62, 64–68].

Chapter 4: Based on the solving strategies introduced in Chapter 3, we pushed the ideas of [48, 62] implemented in `SolveCoupledSystems` further and develop a method that aims to optimise the solving task of such systems. In particular, because the inhomogeneous part of the differential systems also contains some integrals, called base case integrals, that depend on ϵ, our aim is to minimise as much as possible the order in ϵ up to which we need to give the expansions of those integrals in order to solve the system up to the desired order given by the user. Our method is split in three parts. In the first part, we preprocess the systems in order to split it into simpler systems (Section 4.1). In the second part, we focus on optimising the ϵ-order expansion only for those subsystems (Sections 4.3 and 4.4). Finally, we recombine all this information from the subsystems to obtain corrections for the full system (Section 4.5). Besides optimizing the ϵ-orders of the expansion we also provide techniques to find the minimal orders of the underlying linear recurrences. This new feature shall be also relevant for the currently used large moment method [62, 64–68]. The method has been implemented in a small package called `SystemAnalysis`, and we illustrate with some small examples how it works (Section B). It is expected that this refined analysis will speed up substantially future calculations in the context of integration by part methods.

All the underlying algorithms from Chapters 2 and 4 are implemented in the available packages `RICA` [69] and `SystemAnalysis`. Details of the usage are given within the chapters as well as in the Appendix A and Appendix B.

In Chapter 5, we outline future possible directions for further extensions.

[1] In reference [48] no special bases is required. It applies to the single-variable case.

2. Asymptotics and special functions

The chapter concentrates on the connection between nested sums, specials functions and their asymptotics, stemming originally from particular physics problems and then extending towards possible mathematical generalisations as the study progresses. More specifically, in the context of particular physics computations in perturbation theory within QCD and QED [16,28,65], we encounter finite and infinite nested binomial and inverse binomial sums that need to be dealt with in a systematic way. This kind of analytic calculations of Feynman integrals aims at high precision predictions for particle physics experiments for processes both with massless partons but also massive quarks in QCD. They involve on the mathematical side the computation of increasingly complex iterated integrals. In the process of doing so, many different classes of functions arise, for example harmonic sums and harmonic polylogarithms [12,70] and generalisations thereof, see the survey [71]. The package `HarmonicSums`, developed by Jakob Ablinger [9,10,12,72–74], allows one to manipulate efficiently such objects, and in particular find closed forms or simpler representations, compute (inverse) Mellin transforms, asymptotics, etc. thereof. In particular, Mellin inversion is performed through solving of differential equations [17]. Nevertheless, extensions of nested generalized harmonic sums weighted by the binomial coefficient $\binom{2n}{n}$ cannot always be dealt with in this case, and a different approach to treat them has been presented in [16]. This method relies on recursive computations of Mellin convolutions of the summands of the iterated binomial sums. The main advantage of this approach is that by defining a set of possible general cases, i.e., classes of functions for which we can compute in general the Mellin inverse and/or the Mellin convolution, the computation of the Mellin inverse can be made rather fast and straightforward. Moreover, by identifying new general cases and adding them to our "dictionary", we can easily extend the classes of iterated sums we can deal with, even beyond the binomial case. In the first part of this chapter, after introducing the binomial sums and the associated special functions, we will present in detail the algorithm that allows one to compute inverse Mellin transforms of those. All of those methods, originally only sketched in [16], have been precisely formalised and implemented in a package that we wrote, called `RICA` (short for **R**ule **I**nduced **C**onvolutions for **A**symptotics) [69].

Another advantage in representing the nested binomial sums as Mellin integrals

is that one can perform asymptotic expansions at $n \to +\infty$, where n denotes the Mellin parameter, which helps us to obtain for example analytic continuations of those nested sums [16]. There exist several possible methods to perform asymptotic expansions of functions of the form $\int_0^1 \mathrm{d}x\, x^n f(x)$, i.e., defined as Mellin transforms of some function f, depending on the regularity of f. In particular, there is a rather general method for factorial series [75,76] that relies on changes of variables, and term by term integration enabled by Watson's lemma [16,30]. In the second part of the chapter, we present the asymptotics methods, all of which have been properly formalised and implemented in `RICA` together with several variations or simpler methods to speed up computations when possible.

The theory presented in [16] encounters limitations as in general a Mellin inverse and therefore asymptotics could not be obtained for all of the binomial sums. Therefore, throughout this chapter, we study some additional cases, in particular extended binomial sums containing rational functions (Section 2.2.1), or certain specific classes involving plus-prescriptions (Section 2.2.2). Furthermore, we introduced a new class of functions, the generalized convolution polylogarithms, to deal with cases when a standard generalized polylogarithm representation could not be obtained. Due to the structure of this class of new functions, we could in particular prove a theorem that allowed us to use them to compute new asymptotics of binomials sums that could not be achieved with the methods of [16] alone. These results are presented in particular in Section 2.3 and Section 2.4. We note that these classes of sums cannot be tackled by using other available techniques such as given in [17]. Here one relies on the fact that the underlying differential equations must be solved in closed form; which usually fails for sums such as (1.0.1) — (1.0.3). All of these new tools have been implemented in the `RICA` package.

2.1. From harmonic to binomial sums

We start with a definition that will be useful for binomial sums as well as for their functional counterparts, the generalized polylogarithms[2].

Definition 2.1.1 (Letter, alphabet, word)**.** *We call a* ***letter*** *a function a_i that is defined at least on $[0;1]$ and is integrable on $]0;1[$, the* ***alphabet*** *$\mathcal{A}$ the collection of all such considered letters a_i, i.e., $\mathcal{A} = \{a_1, \ldots, a_n\}$, and a* ***word of length n*** *a sequence of n such letters $a_{i_1}, a_{i_2}, \ldots, a_{i_n}$; the empty set is denoted by $\emptyset$.*

Remark 2.1.1. *Throughout this work, whenever we write $\mathbb{K}$ we denote, unless explicitely stated otherwise, a computable subfield of $\mathbb{R}$ or $\mathbb{C}$, and we note $\mathbb{K}^\star := \mathbb{K}\backslash\{0\}$.*

[2] See Subsection 2.1.2 for more details.

Remark 2.1.2. *The function notation in letters is later needed for the definition of iterated integrals. For the sum case, the letters are actually constants, i.e., elements from a field $\mathbb{K}$ (often $\mathbb{K} \subseteq \mathbb{R}$).*

Definition 2.1.2 (Binomial sums)**.** *Let $n \in \mathbb{N}$. We define a* ***binomial sum*** *(or* ***iterated binomial sum****, IBS) as follows:*

$$\mathcal{BS}_{(\emptyset)}(n) = 1 \tag{2.1.1}$$

$$\mathcal{BS}_{(a_1)}(n) = \sum_{i=1}^{n} a_1(i; b, c, m) \tag{2.1.2}$$

$$\mathcal{BS}_{(a_1,\ldots,a_k)}(n) = \sum_{i_1=1}^{n} a_1(i_1; b_1, c_1, m_1)\mathcal{BS}_{(a_2,\ldots,a_k)}(i_1) \tag{2.1.3}$$

where we define the ***letters*** *a_p of the binomial sum as:*

$$a_p(i; b, c, m) = \binom{2i}{i}^b \frac{c^i}{i^m},\ b \in \{-1, 0, 1\},\ c \in \mathbb{K}^\star,\ m \in \mathbb{N}. \tag{2.1.4}$$

k is called the ***depth*** *of the sum.*

Remark 2.1.3. *We can also write the definition in a non-recursive way as:*

$$\mathcal{BS}_{(a_1,\ldots,a_k)}(n) = \sum_{i_1=1}^{n} a_1(i_1; b_1, c_1, m_1) \sum_{i_2=1}^{i_1} a_2(i_2; b_2, c_2, m_2) \cdots \sum_{i_k=1}^{i_{k-1}} a_k(i_k; b_k, c_k, m_k). \tag{2.1.5}$$

We will say that $\mathcal{BS}_{(a_1,\ldots,a_k)}(n)$ is of **depth k**. For practical examples, we will also write the arguments of the letters instead of the letters themselves in the indices of the binomial sum

$$\mathcal{BS}_{(a_1,\ldots,a_k)}(n) = \mathcal{BS}_{((b_1,c_1,m_1),\ldots,(b_k,c_k,m_k))}(n). \tag{2.1.6}$$

In this case letters are triplets $(b_i, c_i, m_i) \in \{-1, 0, 1\} \times \mathbb{K}^\star \times \mathbb{N}$ instead of the a_i.

Example 2.1.1. *Typical examples of binomial sums are:*

$$\mathcal{BS}_{((1,1,0),(1,1,0))}(n) = \sum_{i=1}^{n} \binom{2i}{i} \sum_{j=1}^{i} \binom{2j}{j}$$

$$\mathcal{BS}_{((-1,4,3))}(n) = \sum_{i=1}^{n} \frac{1}{\binom{2i}{i}} \frac{4^i}{i^3}$$

$$\mathcal{BS}_{((1,-1,1),(0,1,2))}(n) = \sum_{i=1}^{n} \binom{2i}{i} \frac{(-1)^i}{i} \sum_{j=1}^{i} \frac{1}{j^2}$$

$$\mathcal{BS}_{\left((1,2,3),\left(-1,\frac{1}{2},1\right),(0,0,2)\right)}(n) = \sum_{i=1}^{n} \binom{2i}{i} \frac{2^i}{i^3} \sum_{j=1}^{i} \frac{\left(\frac{1}{2}\right)^j}{j\binom{2j}{j}} \sum_{k=1}^{j} \frac{1}{k^2}.$$

More generic summands can also be considered in some cases, in order to include not only coefficients of the form $1/i^m$, but more generally certain kinds of rational functions. This class would in particular include coefficients of the form

$$\frac{c^n}{(2i+1)\binom{2n}{n}}.$$

Definition 2.1.3 (Set of roots)**.** *Let $q \in \mathbb{K}[X]$. We denote the* ***set of roots*** *of q with*

$$V(q) = \{r \in \mathbb{K} \mid q(r) = 0\}$$

with $V(q) \subset \bar{\mathbb{K}}$, where $\bar{\mathbb{K}}$ is the algebraic closure of the field $\mathbb{K}$.

Definition 2.1.4 (Extended binomial sums)**.** *Let $n \in \mathbb{N}^\star$. We define an* ***extended binomial sum*** *as follows:*

$$\overline{\mathcal{BS}}_{(\emptyset)} = 1 \tag{2.1.7}$$

$$\overline{\mathcal{BS}}_{(\alpha)}(n) = \sum_{i=1}^{n} \alpha(i; b, c, p, q) \tag{2.1.8}$$

$$\overline{\mathcal{BS}}_{(\alpha_1,\ldots,\alpha_k)}(n) = \sum_{i_1=1}^{n} \alpha_1(i_1; b_1, c_1, p_1, q_1) \overline{\mathcal{BS}}_{(\alpha_2,\ldots,\alpha_k)}(i_1) \tag{2.1.9}$$

where we define the ***letters*** *α_p of the extended binomial sum as:*

$$\alpha_p(i; b, c, p, q) = \binom{2i}{i}^b c^i \frac{p(i)}{q(i)},\ b \in \{-1, 0, 1\},\ c \in \mathbb{K}^\star,\ p, q \in \mathbb{K}[X],\ \deg p \leq \deg q,\ V(q) \cap \mathbb{N} = \emptyset. \tag{2.1.10}$$

Remark 2.1.4. *While for the general definition of the extended binomial sums we consider all roots except non-negative integers, for technical reasons explained in Section 2.2.1 we will eventually restrict to $V(q) \subset \mathbb{R}_- \cap \mathbb{K}$.*

Remark 2.1.5. *Extensions over letters containing linear or square factors in linear representations, i.e., according to the fundamental theorem of algebra [77] the factorisation of any polynomial over*[3] *$\mathbb{R}$, have been studied in [78]. The difference in our case is that in principle we accept factorisations over $\bar{\mathbb{R}} = \mathbb{C}$.*

[3] Or more precisely any computable subfield $\mathbb{K} \subset \mathbb{R}$.

Example 2.1.2. *Typical examples of extended binomial sums are:*

$$\sum_{i=1}^{n} \frac{2^i}{(3i+2)\binom{2i}{i}}$$

$$\sum_{i=1}^{n} \frac{(-2)^i}{(2i+1)\binom{2i}{i}} \sum_{j=1}^{i} \binom{2j}{j} \frac{(-1)^j}{j^3}$$

$$\sum_{i=1}^{n} \frac{1+i}{i^2+5i+6} \sum_{j=1}^{i} \frac{(-2)^j}{j\binom{2j}{j}} \sum_{k=1}^{j} \frac{3^k}{k^2};$$

for examples of such sums, we refer also to [79, Section 4.2].

We can also consider two very important subclasses of binomial sums [9], which are obtained by restricting some of the a_i letter parameters b_i, c_i, m_i, as well as one particular subclass of the extended binomial sums [7, 10].

Definition 2.1.5 (Harmonic Sums)**.** *For $k \in \mathbb{N}^\star$, $n \in \mathbb{N}$ and $a_i \in \mathbb{Z}^\star$ with $i \in \{1, \ldots, k\}$ we define [8, 9]*

$$\mathrm{S}_\emptyset(n) = 1 \tag{2.1.11}$$

$$\mathrm{S}_a(n) = \sum_{i=1}^{n} \frac{\mathrm{sign}(a)^i}{i^{|a|}} \tag{2.1.12}$$

$$\mathrm{S}_{a_1,\ldots,a_k}(n) = \sum_{i=1}^{n} \frac{\mathrm{sign}(a_1)^{i_1}}{i_1^{|a_1|}} \mathrm{S}_{a_2,\ldots,a_k}(i_1). \tag{2.1.13}$$

*k is called the **depth** and $w = \sum_{i=0}^{k} |a_i|$ the **weight** of the **harmonic sums** $\mathrm{S}_{a_1,\ldots,a_k}$.*

Remark 2.1.6. *We can unfold the recursive definition of an harmonic sum and then write it as*

$$\mathrm{S}_{a_1,\ldots,a_k}(n) = \sum_{i_1=1}^{n} \frac{\mathrm{sign}(a_1)^{i_1}}{i_1^{|a_1|}} \sum_{i_2=1}^{i_1} \frac{\mathrm{sign}(a_2)^{i_2}}{i_2^{|a_2|}} \cdots \sum_{i_k=1}^{i_{k-1}} \frac{\mathrm{sign}(a_k)^{i_k}}{i_k^{|a_k|}}.$$

Remark 2.1.7. *In our binomial sum convention, a harmonic sum corresponds to the case where the letters take the form $a_p = a_p(i; 0, 1, m)$, so that*

$$\mathrm{S}_{m_1,\ldots,m_k}(n) = \mathcal{BS}_{((0,1,m_1),\ldots,(0,1,m_k))}(n). \tag{2.1.14}$$

Example 2.1.3. *Typical examples of harmonic sums are:*

$$\mathrm{S}_2(n) = \sum_{i=1}^{n} \frac{1}{i^2}$$

$$\mathrm{S}_{1,1}(n) = \sum_{i=1}^{n} \frac{1}{i} \sum_{j=1}^{i} \frac{1}{j}$$

$$\mathrm{S}_{-1,2,3}(n) = \sum_{i=1}^{n} \frac{(-1)^i}{i} \sum_{j=1}^{i} \frac{1}{j^2} \sum_{k=1}^{j} \frac{1}{k^3}.$$

The second class of sums extends the class of harmonic sums to include geometric type terms in the numerator, that are already studied in [14]. Further algorithmic properties are explored in [9,12].

Definition 2.1.6 (S-sums)**.** *For $a_i \in \mathbb{N}^\star$, $n \in \mathbb{N}$ and $x_i \in \mathbb{K}^\star$ we define*

$$\mathrm{S}_{\emptyset}(\emptyset; n) = 1 \tag{2.1.15}$$

$$\mathrm{S}_{a}(x; n) = \sum_{i=1}^{n} \frac{x^i}{i^a} \tag{2.1.16}$$

$$\mathrm{S}_{a_1,\ldots,a_k}(x_1, \ldots, x_k; n) = \sum_{i_1=1}^{n} \frac{x^{i_1}}{i_1^{a_1}} \mathrm{S}_{a_2,\ldots,a_k}(x_2, \ldots, x_k; n) \tag{2.1.17}$$

*where k is called the **depth** and $w = \sum_{i=0}^{k} a_i$ is called the **weight** of the **S-sum** $\mathrm{S}_{a_1,\ldots,a_k}(x_1, \ldots, x_k; n)$.*

Remark 2.1.8. *We can also write the S-sums in a non-recursive way as*

$$\mathrm{S}_{a_1,\ldots,a_k}(x_1, \ldots, x_k; n) = \sum_{i_1=1}^{n} \frac{x_1^{i_1}}{i_1^{a_1}} \sum_{i_2=1}^{i_1} \frac{x_2^{i_2}}{i_2^{a_2}} \cdots \sum_{i_k=1}^{i_{k-1}} \frac{x_k^{i_k}}{i_k^{a_k}}. \tag{2.1.18}$$

Example 2.1.4. *Examples of S-sums are:*

$$\mathrm{S}_{2}(-1; n) = \sum_{i=1}^{n} \frac{(-1)^i}{2^i}$$

$$\mathrm{S}_{1,2}\left(\frac{1}{2}, -1; n\right) = \sum_{i=1}^{n} \frac{\left(\frac{1}{2}\right)^i}{i} \sum_{j=1}^{i} \frac{(-1)^j}{j^2}$$

$$\mathrm{S}_{2,1,2}\left(2, \frac{1}{2}, 1\right) = \sum_{i=1}^{n} \frac{2^i}{i^2} \sum_{j=1}^{i} \frac{\left(\frac{1}{2}\right)^j}{j} \sum_{k=1}^{j} \frac{1}{k^2}.$$

Finally, we consider the case when we restrict the rational functions in extended binomial sums to linear terms in the denominator, which corresponds to cyclotomic sums [10], which can be seen themselves as extensions of generalized harmonic sums.

Definition 2.1.7 (Cyclotomic sums)**.** *For $n \in \mathbb{N}^\star$, we define the* ***cyclotomic sums*** *as follows:*

$$\mathrm{S}_{\{\emptyset\}}(\emptyset; n) = 1 \tag{2.1.19}$$

$$\mathrm{S}_{\{a,b,c\}}(s; n) = \sum_{i=1}^{n} \frac{s^i}{(ai+b)^c} \tag{2.1.20}$$

$$\mathrm{S}_{\{a_1,b_1,c_1\},\dots,\{a_k,b_k,c_k\}}(s_1,\dots,s_k; n) = \sum_{i_1=1}^{n} \frac{s_1^{i_1}}{(a_1 i_1 + b_1)^{c_1}} \mathrm{S}_{\{a_2,b_1,c_1\},\dots,\{a_k,b_k,c_k\}}(s_2,\dots,s_k; n) \tag{2.1.21}$$

where $a_i, c_i \in \mathbb{N}^\star$, $b_i \in \mathbb{N}$, $s_i \in \{-1, 1\}$, $a_i > b_i$. The ***weight*** *of the sum is $w = \sum_{i=1}^{k} c_i$, and $\{a_i, b_i, c_i\}$ denotes lists and not sets.*

Example 2.1.5. *Examples of cyclotomic sums are:*

$$\mathrm{S}_{\{2,1,3\}}\left(\frac{1}{2}; n\right) = \sum_{i=1}^{n} \frac{\left(\frac{1}{2}\right)^i}{(2i+1)^3}$$

$$\mathrm{S}_{\{1,2,3\},\{2,2,2\}}(7, 5; n) = \sum_{i=1}^{n} \frac{7^i}{(i+2)^3} \sum_{j=1}^{i} \frac{5^j}{(2j+2)^2}.$$

Because we have defined several classes of sums that have a hierarchy among them, we summarize their relations in Diagram 2.1, where with CS we denote the cyclotomic sums and HS the harmonic sums. The labels of the arrows illustrate what terms are transformed or simplified when we go from one class to the other.

2.1.1. Basic properties of the Mellin transform

The Mellin transform is an integral transform that can be obtained from the Laplace transform[4] that allows us to go from the function world to the sequence world or vice-versa (using the inverse transform). This tool will be crucial for us to study the binomial sums.

Definition 2.1.8 (Mellin transform)**.** *If f is an integrable function defined at least on the interval $(0; 1)$, we define the* ***Mellin transform*** *of f as:*

$$\mathbf{M}\left[f(x)\right](n) := \int_0^1 \mathrm{d}x\, x^n f(x). \tag{2.1.22}$$

It might happen that the function f is mildly non-integrable around $x = 1$, more precisely $f(x) \underset{x\to 1}{\sim} \alpha(1-x)^{-1}$, $\alpha \in \mathbb{K}^\star$. Then we need to extend the definition of the Mellin transform, for that we need to define the following distribution.

[4] Though historically this is not what happened, it had been studied only in specific cases by Mellin and was seemingly named "Mellin transform" by Hardy himself [80].

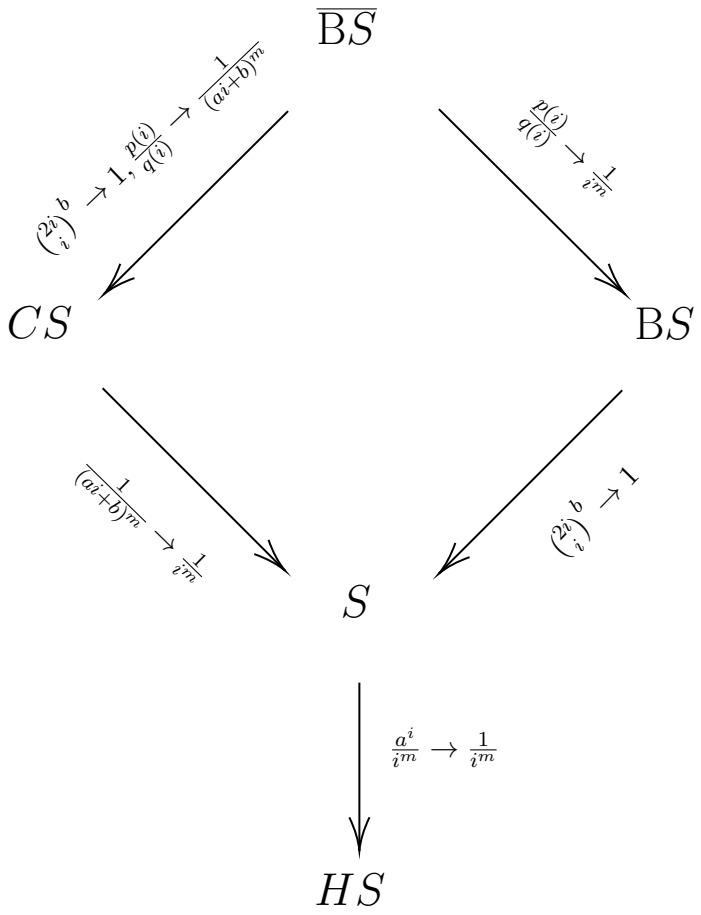


Figure 2.1.: Hierachy of nested sums

Definition 2.1.9 (+-distribution)**.** *Let f be a function that is defined at least on $(0;1)$. We can associate to it a distribution T_f^+ that acts on a test function g, provided that it is smooth enough on $(0;1)$ and defined at $x=1$, in the following way:*

$$\left\langle T_f^+, g \right\rangle = \int_0^1 \mathrm{d}x \, [f(x)]_+ \, g(x) := \int_0^1 \mathrm{d}x \, [g(x) - g(1)] \, f(x). \tag{2.1.23}$$

Definition 2.1.10 (Mellin transform for the +-prescription)**.** *Let f be a function that is defined in $(0;1)$ but not integrable at $x=1$. We define then the **Mellin transform** as a distribution and call this **+-prescription**:*

$$\mathbf{M}\left[[f(x)]_+\right](n) := \int_0^1 \mathrm{d}x \, (x^n - 1) f(x). \tag{2.1.24}$$

Definition 2.1.11 (Extended Mellin transform)**.** *If f is an integrable function defined at least on the interval $(0;1)$, $a \in \mathbb{K}$, we define the **extended Mellin transform** of f as:*

$$\tilde{\mathbf{M}}_a \left[f(x)\right](n) := \int_0^1 \mathrm{d}x \, \left[(ax)^n - 1\right] f(x). \tag{2.1.25}$$

This transform obeys in particular three classical relations, which will be crucial for us when it will come to rewriting, or more precisely Mellin inverting, them as Mellin transforms of certain classes of special functions that will be defined in the next Subsection 2.1.2.

Property 2.1.1 (Shift relation)**.** *For all $n, k \in \mathbb{N}$, the shift relation is:*

$$\mathbf{M}\left[f(x)\right](n+k) = \mathbf{M}\left[x^k f(x)\right](n). \tag{2.1.26}$$

Proof. This relation simply follows from the definition of the Mellin transform (2.1.8) and the fact that $x^{n+k} f(x) = x^n (x^k f(x))$. □

Property 2.1.2 (Resummation relation)**.** *For all $c \in \mathbb{R}$, the resummation relation is:*

$$\sum_{i=1}^{n} c^i \mathbf{M}\left[f(x)\right](i) = c^n \mathbf{M}\left[\frac{x}{x-\frac{1}{c}} f(x)\right](n) - \mathbf{M}\left[\frac{x}{x-\frac{1}{c}} f(x)\right](0). \tag{2.1.27}$$

Proof. One can easily obtain the resummation relation starting from the right handside:

$$c^n \mathbf{M}\left[\frac{x}{x-\frac{1}{c}} f(x)\right](n) - \mathbf{M}\left[\frac{x}{x-\frac{1}{c}} f(x)\right](0) = \int_0^1 \mathrm{d}x \, (cx)^n \frac{x}{x-\frac{1}{c}} f(x)$$

$$
\begin{aligned}
&- \int_0^1 \mathrm{d}x \, \frac{x}{x - \frac{1}{c}} f(x) \\
&= \int_0^1 \mathrm{d}x \, \frac{[(cx)^n - 1]\, x}{x - \frac{1}{c}} f(x) \\
&= \int_0^1 \mathrm{d}x \, \frac{[(cx)^{n+1} - (cx)]}{(cx) - 1} f(x) \\
\text{geometric sum} \rightarrow &= \sum_{i=1}^{n} c^i \mathbf{M}\left[f(x)\right](i)
\end{aligned}
$$

which concludes the proof. □

The last property relies on the fact that the Mellin transform preserves the structure of the function and sequence spaces — in particular the multiplication in sequence spaces is transposed to a convolution operation in the function space. The Mellin convolution, introduced next, will be crucial to introduce Property 2.1.4 below.

Definition 2.1.12 (Mellin convolution)**.** *Let f, g be two functions with support (at least) $[0;1]$, g integrable on $[0;1]$ and f integrable on $[0;1]$, or on $[0;1)$ and furthermore $f(x) \underset{x\to 1}{\sim} \alpha(1-x)^{-1}$, $\alpha \in \mathbb{R}^\star$. Then the* ***Mellin convolution*** *is defined as*

$$
(f \star g)(x) = f(x) \star g(x) = \int_0^1 \mathrm{d}x_1 \int_0^1 \mathrm{d}x_2 \, \delta(x - x_1 x_2) f(x_1) f(x_2). \tag{2.1.28}
$$

Property 2.1.3. *This Mellin convolution can be rewritten in a more useable form, depending on the regularity of f, g as well as whether we use the +-prescription [81].*

- *If f and g are both integrable at $x = 1$:*

$$
f(x) \star g(x) = \int_x^1 \mathrm{d}y \, \frac{1}{y} f(y) g\left(\frac{x}{y}\right) = \int_x^1 \mathrm{d}y \, \frac{1}{y} g(y) f\left(\frac{x}{y}\right). \tag{2.1.29}
$$

- *If $f \underset{x\to 1}{\sim} \alpha(1-x)^{-1}$, $\alpha \in \mathbb{K}^\star$ and g is integrable at $x = 1$:*

$$
\begin{aligned}
[f(x)]_+ \star g(x) &= \int_x^1 \mathrm{d}y \, f\left(\frac{x}{y}\right) \left[\frac{g(y)}{y} - \frac{x g(x)}{y^2}\right] - g(x) \int_0^x \mathrm{d}y \, f(y) \\
&= \int_x^1 \mathrm{d}y \, f\left(y\right) \left[\frac{1}{y} g\left(\frac{x}{y}\right) - g(x)\right] - g(x) \int_0^x \mathrm{d}y \, f(y).
\end{aligned} \tag{2.1.30}
$$

Proof. One can obtain (2.1.29) by performing explicitely the integral and using the standard formula for the Dirac δ-function:

$$
\delta(f(x)) = \sum_i \frac{\delta(x - x_i)}{|f'(x_i)|} \text{ with } x_i \text{ zeroes of } f. \tag{2.1.31}
$$

More precisely:

$$\begin{aligned}
(f \star g)(x) &= \int_0^1 \mathrm{d}x_1 \int_0^1 \mathrm{d}x_2\, \delta(x - x_1 x_2) f(x_1) g(x_2) \\
&= \int_0^1 \mathrm{d}x_1 \int_0^1 \mathrm{d}x_2\, \frac{1}{x_1} \delta\left(x_2 - \frac{x}{x_1}\right) f(x_1) g(x_2) \\
&= \int_0^1 \mathrm{d}x_1 \int_{-\infty}^{+\infty} \mathrm{d}x_2\, \frac{1}{x_1} \delta\left(x_2 - \frac{x}{x_1}\right) f(x_1) g(x_2) \Theta(1 - x_2) \Theta(x_2) \\
&= \int_0^1 \mathrm{d}x_1\, \frac{1}{x_1} f(x_1) g\left(\frac{x}{x_1}\right) \Theta\left(1 - \frac{x}{x_1}\right) \Theta\left(\frac{x}{x_1}\right) \\
&= \int_x^1 \mathrm{d}y\, \frac{1}{y} f(y) g\left(\frac{x}{y}\right).
\end{aligned}$$

In the case when f is defined in the +-prescription we have to perform the integral more carefully:

$$\begin{aligned}
(f_+ \star g)(x) &= \int_0^1 \mathrm{d}x_1 \int_0^1 \mathrm{d}x_2\, \delta(x - x_1 x_2) \left[f(x_1)\right]_+ g(x_2) \\
&= \int_0^1 \mathrm{d}x_1\, \left[f(x_1)\right]_+ \underbrace{\int_0^1 \mathrm{d}x_2\, \delta(x - x_1 x_2) g(x_2)}_{=: \bar{g}_x(x_1)} \\
(2.1.23) \rightarrow &= \int_0^1 \mathrm{d}x_1\, \left[\bar{g}_x(x_1) - \bar{g}_x(1)\right] f(x_1) \\
&= \int_0^1 \mathrm{d}x_1 \left[\frac{1}{x_1} \int_0^1 \mathrm{d}x_2\, \delta\left(x_2 - \frac{x}{x_1}\right) g(x_2) - \int_0^1 \mathrm{d}x_2\, \delta(x_2 - x) g(x_2)\right] f(x_1) \\
&= \int_0^1 \mathrm{d}x_1 \left[\frac{1}{x_1} g\left(\frac{x}{x_1}\right) \Theta\left(1 - \frac{x}{x_1}\right) \Theta\left(\frac{x}{x_1}\right) - g(x) \Theta(1 - x) \Theta(x)\right] f(x_1).
\end{aligned}$$

Now we use the fact that $0 < x < 1$ holds to directly simplify $\Theta(1 - x)\Theta(x) = 1$. Splitting the integral yields:

$$\begin{aligned}
(f_+ \star g)(x) = &\int_0^x \mathrm{d}x_1 \left[\frac{1}{x_1} g\left(\frac{x}{x_1}\right) \Theta\left(1 - \frac{x}{x_1}\right) \Theta\left(\frac{x}{x_1}\right) - g(x)\right] f(x_1) \\
&+ \int_x^1 \mathrm{d}x_1 \left[\frac{1}{x_1} g\left(\frac{x}{x_1}\right) \Theta\left(1 - \frac{x}{x_1}\right) \Theta\left(\frac{x}{x_1}\right) - g(x)\right] f(x_1).
\end{aligned}$$

Because $0 < x_1 < x \Rightarrow 1 - \frac{x}{x_1} < 0$ and $x < x_1 < 1 \Rightarrow 1 - \frac{x}{x_1} > 0$ we can simplify the first and second integral further which leads to the expected result:

$$(f_+ \star g)(x) = \int_x^1 \mathrm{d}y\, f(y) \left[\frac{1}{y} g\left(\frac{x}{y}\right) - g(x)\right] - g(x) \int_0^x \mathrm{d}y\, f(y).$$

This concludes the proof. □

A central property of the Mellin transform that we will use is:

Property 2.1.4 (Multiplicative relation)**.** *For f and g two functions with support (at least) $[0;1]$, g integrable on $[0;1]$ and f integrable on $[0;1]$, or on $[0;1)$ and $f(x) \underset{x\to 1}{\sim} \alpha(1-x)^{-1}$, $\alpha \in \mathbb{K}^\star$, we have the multiplicative relation:*

$$\mathbf{M}\left[f(x)\right](n) \cdot \mathbf{M}\left[g(x)\right](n) = \mathbf{M}\left[f(x) \star g(x)\right](n). \tag{2.1.32}$$

Lemma 2.1.1. *Let f and g be two functions with the same assumptions as in Property 2.1.4 above. Their convolution can be rewritten as:*

$$(f \star g)(x) = \int_x^1 \mathrm{d}y\, \frac{1}{y} f(y) g\left(\frac{x}{y}\right) = \int_0^{+\infty} \mathrm{d}z\, \bar{f}(z)\bar{g}(xz) \tag{2.1.33}$$

where

$$\bar{f}(z) := \Theta(z-1)\frac{1}{z} f\left(\frac{1}{z}\right), \quad \bar{g}(z) := \Theta(1-z)g(z).$$

Proof. To prove the lemma, it suffices to plug in the expressions for $\bar{f}$ and $\bar{g}$ in (2.1.33) and use the definition of the Heaviside Θ function:

$$\begin{aligned}
\int_0^{+\infty} \mathrm{d}z\, \bar{f}(z)\bar{g}(xz) &= \int_0^{+\infty} \mathrm{d}z\, \Theta(z-1)\Theta(1-xz)\frac{1}{z} f\left(\frac{1}{z}\right) g(xz) \\
&= \int_{\frac{1}{x}}^1 \mathrm{d}z\, \frac{1}{z} f\left(\frac{1}{z}\right) g(xz) \\
\text{change of variable} \to &= \int_x^1 \mathrm{d}y\, \frac{1}{y} f(y) g\left(\frac{x}{y}\right);
\end{aligned}$$

which concludes the proof. □

Property 2.1.5. *The Mellin convolution is associative, i.e., for three functions f, g, h we have:*

$$[(f \star g) \star h](x) = [f \star (g \star h)](x).$$

Proof. By applying the definition of the Mellin convolution (2.1.28) twice on the left-hand side we get:

$$\begin{aligned}
[(f \star g) \star h)](x) &= \underbrace{\left(\int_0^1 \mathrm{d}y \int_0^1 \mathrm{d}z\, \delta(x-yz) f(y) g(z)\right)}_{(f\star g)(x)} \star h(x) \\
&= \int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}w\, \delta(x-uw)(f \star g)(u) h(w)
\end{aligned}$$

$$= \int_0^1 \mathrm{d}y \int_0^1 \mathrm{d}z \int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}w\, \delta(x-uw)\delta(u-yz) f(y)g(z)h(w). \tag{2.1.34}$$

In a similar way, we get for the right hand side

$$[f \star (g \star h)](x) = \int_0^1 \mathrm{d}y \int_0^1 \mathrm{d}y \int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}w\, \delta(x-uw)\delta(w-yz) f(u)g(y)h(z). \tag{2.1.35}$$

Let us rename the variables $y \to w$, $w \to z$, $z \to y$ and then exchange $u \leftrightarrow w$ in (2.1.34):

$$[(f \star g) \star h)](x) = \int_0^1 \mathrm{d}y \int_0^1 \mathrm{d}z \int_0^1 \mathrm{d}u \int_0^1 \mathrm{d}w\, \delta(x-wz)\delta(w-uy) f(u)g(y)h(z).$$

We see that the functions f,g and h completely decouple from the w variable in both cases. So we integrate w out[5] and get:

$$\int_0^1 \mathrm{d}w\, \delta(x-uw)\delta(w-yz) = \int_{-\infty}^{+\infty} \mathrm{d}w\, \delta(x-uw)\delta(w-yz)\Theta(1-w)\Theta(w) = \delta(x-uyz)$$

$$\int_0^1 \mathrm{d}w\, \delta(x-wz)\delta(w-uy) = \int_{-\infty}^{+\infty} \mathrm{d}w\, \delta(x-wz)\delta(w-uy)\Theta(1-w)\Theta(w) = \delta(x-uyz).$$

This proves $[(f \star g) \star h](x) = [f \star (g \star h)](x)$. □

2.1.2. Polylogarithms, harmonic polylogarithms and generalized polylogarithms

Following the convention of [16], we define generalized root-valued polylogarithms using the same convention for letters, alphabet and words as in Definition 2.1.1.

Definition 2.1.13 (Generalized root-valued polylogarithms)**.** *Let f_a be a letter and $\vec{g}$ be a word made of letters that are all integrable at $x = 1$. We define the* ***generalized root-valued polylogarithms*** *(GRPL) as iterated integrals of the form, see reference [16]*

$$\mathrm{H}^*_{\emptyset}(x) := 1 \tag{2.1.36}$$

$$\mathrm{H}^*_{\mathrm{f_a}(\lambda),\vec{g}(\lambda)}(x) = \mathrm{H}^*_{\mathrm{f_a},\vec{g}}(x) := \int_x^1 \mathrm{d}t\, f_a(t)\mathrm{H}^*_{\vec{g}}(t), \tag{2.1.37}$$

[5] We consider these integrals in the distribution sense, as a special case of a distribution convolution [82].

and the letters are those defined in the Remark 2.1.11. Here $f(\lambda)$ should be read as $\lambda \mapsto f(\lambda)$ and not as the function f depending explicitely on a parameter called λ (iterated integral $\rightarrow$ name of variable changes). The ***depth k*** *of a GRPL is the length of the word used as their index, and naturally corresponds with the number of nested integrals it consists of.*

Remark 2.1.9. *We can also unfold the recursive definition of the GRPLs and write them as*

$$\mathrm{H}^{*}_{\mathrm{f}_1,\ldots,\mathrm{f}_k}(x) = \int_x^1 \mathrm{d}t_1\, f_1(t_1) \int_{t_1}^1 \mathrm{d}t_2\, f_2(t_2) \cdots \int_{t_{k_1}}^1 \mathrm{d}t_k\, f_k(t_k). \tag{2.1.38}$$

Remark 2.1.10. *Let us note that it is also possible to define the GRPLs using integrals with lower bound 0 and not with upper bound 1, i.e., reusing the same conventions as in Definition 2.1.13, we can write*

$$\mathrm{H}_{\emptyset}(x) := 1 \tag{2.1.39}$$

$$\mathrm{H}_{\mathrm{f_a}(\lambda),\overrightarrow{g}(\lambda)}(x) = \mathrm{H}_{\mathrm{f_a},\overrightarrow{g}}(x) := \int_0^x \mathrm{d}t\, f_a(t)\mathrm{H}_{\overrightarrow{g}}(t). \tag{2.1.40}$$

If we make use of this convention, we will denote it with H *instead of* H^{*}.

Property 2.1.6. *If $f_1, \ldots, f_k$, $k \in \mathbb{N}^{\star}$ are letters that are all integrable at $x = 1$, then there exists the relation:*

$$\mathrm{H}^{*}_{\mathrm{f}_1(\lambda),\mathrm{f}_2(\lambda),\ldots,\mathrm{f}_k(\lambda)}(x) = \mathrm{H}_{\mathrm{f}_1(1-\lambda),\mathrm{f}_2(1-\lambda),\ldots,\mathrm{f}_k(1-\lambda)}(1-x). \tag{2.1.41}$$

Proof. One can prove this by iterated changes of variables $t_i \rightarrow 1 - t_i$:

$$\begin{aligned}
\mathrm{H}^{*}_{\mathrm{f}_1(\lambda),\mathrm{f}_2(\lambda),\ldots,\mathrm{f}_k(\lambda)}(x) &= \int_x^1 \mathrm{d}t_1\, f_1(t_1) \int_{t_1}^1 \mathrm{d}t_2\, f_2(t_2) \cdots \int_{t_{k_1}}^1 \mathrm{d}t_k\, f_k(t_k) \\
t_1 \rightarrow 1-t_1 &= \int_0^{1-x} \mathrm{d}t_1\, f_1(1-t_1) \int_{1-t_1}^1 \mathrm{d}t_2\, f_2(t_2) \cdots \int_{t_{k_1}}^1 \mathrm{d}t_k\, f_k(t_k) \\
t_2 \rightarrow 1-t_2 &= \int_0^{1-x} \mathrm{d}t_1\, f_1(1-t_1) \int_0^{t_1} \mathrm{d}t_2\, f_2(1-t_2) \cdots \int_{t_{k_1}}^1 \mathrm{d}t_k\, f_k(t_k) \\
&\vdots = \vdots \\
t_k \rightarrow 1-t_k &= \int_0^{1-x} \mathrm{d}t_1\, f_1(1-t_1) \int_0^{t_1} \mathrm{d}t_2\, f_2(1-t_2) \cdots \int_0^{t_{k_1}} \mathrm{d}t_k\, f_k(1-t_k) \\
&= \mathrm{H}_{\mathrm{f}_1(1-\lambda),\mathrm{f}_2(1-\lambda),\ldots,\mathrm{f}_k(1-\lambda)}(1-x)
\end{aligned}$$

which concludes the proof. □

Similarly, we can convert H functions into H^* functions:

Property 2.1.7. *If $f_1, \ldots, f_k$, $k \in \mathbb{N}^\star$ are letters that are all integrable at $x = 0$, then*

$$\mathrm{H}_{\mathrm{f}_1(\lambda),\mathrm{f}_2(\lambda),\ldots,\mathrm{f}_\mathrm{k}(\lambda)}(x) = \mathrm{H}^*_{\mathrm{f}_1(1-\lambda),\mathrm{f}_2(1-\lambda),\ldots,\mathrm{f}_\mathrm{k}(1-\lambda)}(1-x). \tag{2.1.42}$$

Proof. Identical to the one for Property 2.1.6 if one reverses the roles of H and H^*. □

Remark 2.1.11. *In the present thesis, we will work with the following alphabet, containing root-valued letters [16]:*

$$f_1(x) = \frac{1}{1-x} \tag{2.1.43}$$

$$f_{-1}(x) = \frac{1}{1+x} \tag{2.1.44}$$

$$f_2(x) = \frac{1}{2-x} \tag{2.1.45}$$

$$f_{-2}(x) = \frac{1}{2+x} \tag{2.1.46}$$

$$f_{\mathrm{w}_1}(x) = \frac{1}{\sqrt{x}\sqrt{1-x}} \tag{2.1.47}$$

$$f_{\mathrm{w}_2}(x) = \frac{1}{\sqrt{x}\sqrt{1+x}} \tag{2.1.48}$$

$$f_{\mathrm{w}_3}(x) = \frac{1}{x\sqrt{1-x}} \tag{2.1.49}$$

$$f_{\mathrm{w}_4}(x) = \frac{1}{x\sqrt{1+x}} \tag{2.1.50}$$

$$f_{\mathrm{w}_5}(x) = \frac{1}{\sqrt{1+x}\sqrt{2+x}} \tag{2.1.51}$$

$$f_{\mathrm{w}_6}(x) = \frac{1}{\sqrt{1-x}\sqrt{2-x}} \tag{2.1.52}$$

$$f_{\mathrm{w}_7}(x) = \frac{1}{\sqrt{1-x}\sqrt{2+x}} \tag{2.1.53}$$

$$f_{\mathrm{w}_8}(x) = \frac{1}{x\sqrt{x-\frac{1}{4}}} \tag{2.1.54}$$

$$f_{\mathrm{w}_9}(x) = \frac{1}{(1-x)\sqrt{x}} = f_{\mathrm{w}_3}(1-x) \tag{2.1.55}$$

$$f_{\mathrm{w}_{10}}(x) = \frac{1}{x\sqrt{2-x}} \tag{2.1.56}$$

$$f_{w_{11}}(x) = \frac{1}{x\sqrt{1-x}\sqrt{2-x}} \tag{2.1.57}$$

$$f_{w_{12}}(x) = \frac{1}{\sqrt{x}\sqrt{8-x}} \tag{2.1.58}$$

$$f_{w_{13}}(x) = \frac{1}{(2-x)\sqrt{x}\sqrt{8-x}} \tag{2.1.59}$$

$$f_{w_{14}}(x) = \frac{1}{x\sqrt{x+\frac{1}{4}}} \tag{2.1.60}$$

$$f_{w_{15}}(x) = \frac{1}{x\sqrt{1+x}\sqrt{2+x}} \tag{2.1.61}$$

$$f_{w_{16}}(x) = \frac{1}{x\sqrt{2+x}} \tag{2.1.62}$$

$$f_{w_{17}}(x) = \frac{1}{\sqrt{x}\sqrt{8+x}} \tag{2.1.63}$$

$$f_{w_{18}}(x) = \frac{1}{(x+2)\sqrt{x}\sqrt{8+x}} \tag{2.1.64}$$

$$f_{w_{19}}(x) = \frac{1}{\sqrt{x}\sqrt{4-x}} \tag{2.1.65}$$

$$f_{w_{20}}(x) = \frac{1}{\sqrt{x}\sqrt{4+x}} \tag{2.1.66}$$

$$f_{w_{21}}(x) = \frac{1}{\sqrt{4+x}\sqrt{8+x}} \tag{2.1.67}$$

$$f_{w_{22}}(x) = \frac{1}{(2+x)\sqrt{x-\frac{1}{4}}} \tag{2.1.68}$$

$$f_{w_{23}}(x) = \frac{1}{(1+x)\sqrt{x}\sqrt{4+x}} \tag{2.1.69}$$

$$f_{w_{24}}(x) = \frac{1}{(2-x)\sqrt{x+\frac{1}{4}}} \tag{2.1.70}$$

$$f_{w_{25}}(x) = \frac{1}{\sqrt{4-x}\sqrt{8-x}} \tag{2.1.71}$$

$$f_{w_{26}}(x) = \frac{1}{(1-x)\sqrt{x}\sqrt{4-x}} \tag{2.1.72}$$

$$f_{w_{27}}(x) = \frac{1}{\left(x+\frac{1}{2}\right)\sqrt{x}\sqrt{4-x}} \tag{2.1.73}$$

$$f_{w_{28}}(x) = \frac{1}{x\sqrt{x+\frac{1}{8}}} \tag{2.1.74}$$

$$f_{\mathrm{w}_{29}}(x) = \frac{1}{(1-x)\sqrt{x-\frac{1}{4}}} \tag{2.1.75}$$

$$f_{\mathrm{w}_{30}}(x) = \frac{1}{(1-x)\sqrt{x}\sqrt{8-x}} \tag{2.1.76}$$

$$f_{\mathrm{w}_{31}}(x) = \frac{1}{(1+x)\sqrt{x+\frac{1}{4}}} \tag{2.1.77}$$

$$f_{\mathrm{w}_{32}}(x) = \frac{1}{(1-x)\sqrt{x}\sqrt{8+x}}. \tag{2.1.78}$$

Remark 2.1.12. *If one or several letters are non integrable at $x = 1$ and are of the form $f(x) \underset{x\to 1}{\sim} (1-x)^{-1}$, then we define the GRPLs as follows:*

$$\mathrm{H}^*_{\mathrm{v},\tilde{\mathrm{w}}}(x) := \int_x^1 \mathrm{d}t \left(f_v(t)\mathrm{H}^*_{\tilde{\mathrm{w}}}(t) - \sum_{i=0}^{k} c_i \frac{\ln(1-t)^i}{1-t} \right) + \sum_{i=0}^{k} \frac{c_i}{i+1} \ln(1-x)^{i+1} \tag{2.1.79}$$

where $k, c_0, \ldots, c_k$ are chosen so that all the non-integrable parts are cancelled out[6]*. In particular, for one unique non-integrable letter we have:*

$$\mathrm{H}^*_{\mathrm{w}}(x) := \int_x^1 \mathrm{d}t \left(\mathrm{H}^*_{\mathrm{w}}(t) - \frac{c}{1-t} \right) + c\ln(1-x). \tag{2.1.80}$$

Example 2.1.6. *Some examples of GRPL functions are:*

$$\begin{aligned}
\mathrm{H}^*_{\mathrm{f}_1}(x) &= \int_x^1 \mathrm{d}t\, \frac{1}{1-t} \\
\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3}}(x) &= \int_x^1 \mathrm{d}t_1\, \frac{1}{t_1} \int_{t_1}^1 \mathrm{d}t_2\, \frac{1}{t_2\sqrt{1-t_2}} \\
\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_{11}},\mathrm{f}_2,\mathrm{f}_{\mathrm{w}_8}}(x) &= \int_x^1 \mathrm{d}t_1\, \frac{1}{t_1\sqrt{1-t_1}\sqrt{2-t_1}} \int_{t_1}^1 \mathrm{d}t_2\, \frac{1}{2-t_2} \int_{t_2}^1 \mathrm{d}t_3\, \frac{1}{t_3\sqrt{t_3-\frac{1}{4}}}.
\end{aligned}$$

Remark 2.1.13. *If $f \in \mathcal{A} = \{f_{\alpha_1}, \ldots, f_{\alpha_n}\}$ (where the α_i are indices that can be for example numbers, number-indexed letters, etc.) it can be directly identified by its index $\alpha_i \in \{\alpha_1, \ldots, \alpha_n\}$, so we will write in those cases as a shorthand notation:*

$$\mathrm{H}^*_{\mathrm{f}_{\alpha_i}}(x) = \mathrm{H}^*_{\alpha_i}(x).$$

Remark 2.1.14. *Different to the usual definition of the generalized polylogarithm, for example in [8, 10, 12], we integrate in the $[x;1]$ region and not the $[0;x]$ region.*

[6] This definition is related to Hadamard regularization [83, 84].

Remark 2.1.15. *This definition for GRPLs that have a non-integrable singularity at $x = 1$ is unique and respects the Property 2.1.8.*

The choice of those particular letters in Remark 2.1.11 is motivated by the following facts [16]:

- The alphabet is such that there are no relations among the general polylogarithms (GPLs) modulo the shuffle relations.
- Some root letters were already found in the physical literature, for example in 2-loop integrals with massive propagators:

$$\frac{1}{\sqrt{x(4 \pm x)}} \quad \frac{1}{(1 \pm x)\sqrt{x(4 \pm x)}}.$$

- More specifically, we will show in Section 2.1.2.1 that the binomial coefficient is directly related to the following letters:

$$\binom{2n}{n} \sim \frac{1}{\sqrt{x(1-x)}} \quad \frac{1}{n\binom{2n}{n}} \sim \frac{1}{x\sqrt{1-x}} \quad p, n \in \mathbb{N}.$$

For the letters that we have defined, the corresponding polylogarithms at depth 1 were given in reference [16], equations (3.25)—(3.61).

In the case when the letter is properly defined, or at least integrable at $x = 1$, the computation is explicit, but as we have pointed out before, we have to use a Hadamard regularization (2.1.79) when this is not the case.

Example 2.1.7. *Let us explicitely perform the computation for (2.1.55). Because we have*

$$f_{w_9}(x) \underset{x\to 1}{\sim} \frac{1}{1-x},$$

we use (2.1.80) (special case of (2.1.79)) and get

$$\mathrm{H}^*_{w_9}(x) = \int_x^1 \mathrm{d}t \underbrace{\left(\frac{1}{(1-t)\sqrt{t}} - \frac{c}{1-t}\right)}_{=:d(t)} + c\ln(1-x).$$

We note that

$$d(t) \underset{t\to 1}{=} \frac{c-1}{t-1} + \frac{1}{2} + \mathcal{O}(t-1).$$

In order for the function to be integrable at $x = 1$, we have to fix $c = 1$. Therefore

the integral becomes:

$$\begin{aligned}\mathrm{H}^*_{\mathrm{w}_9}(x) &= \int_x^1 \mathrm{d}t\, \frac{1-\sqrt{t}}{(1-t)\sqrt{t}} + \ln(1-x) \\ &\underset{u=\sqrt{t}}{=} 2\int_{\sqrt{x}}^1 \mathrm{d}u\, \frac{1}{1+u} + \ln(1-x) \\ &= 2\ln 2 - \ln\left(\frac{1+\sqrt{x}}{1-\sqrt{x}}\right)\end{aligned}$$

which we can rewrite using the definition of arctanh *as* (3.76) *in* [16].

By construction of the GRPLs, we have the following

Property 2.1.8. *Let w be a letter and $\vec{w}$ a word, then we have*

$$\frac{\mathrm{d}}{\mathrm{d}x}\mathrm{H}^*_{\mathrm{w},\tilde{\mathrm{w}}}(x) = -f_w(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x). \tag{2.1.81}$$

Definition 2.1.14 (Shuffle relation)**.** *We define the* ***shuffle relation*** *between GPLs as follows:*

$$\mathrm{H}^*_{\tilde{\mathrm{a}}}(x) \sqcup\!\sqcup \mathrm{H}^*_{\tilde{\mathrm{b}}}(x) = \sum_k \mathrm{H}^*_{\tilde{c}_k}(x) \tag{2.1.82}$$

where the $\vec{c}_k$ correspond to merge of the words $\vec{a}$ and $\vec{b}$ such that the relative orders of the elements of $\vec{a}$ and $\vec{b}$ are preserved.[7].

The following lemma tells us how to use the shuffle relation to linearise GPLs [12, 85]:

Lemma 2.1.2. *For $x \in (0;1)$,*

$$\mathrm{H}^*_{\tilde{\mathrm{a}}}(x) \cdot \mathrm{H}^*_{\tilde{\mathrm{b}}}(x) = \mathrm{H}^*_{\tilde{\mathrm{a}}}(x) \sqcup\!\sqcup \mathrm{H}^*_{\tilde{\mathrm{b}}}(x).$$

Example 2.1.8. *For example, for $x \in (0;1)$ we have:*

$$\mathrm{H}^*_{\mathrm{f}_1}(x)\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{-1}}(x) = \mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_0,\mathrm{f}_{-1}}(x) + \mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{-1},\mathrm{f}_1}(x) + \mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_1,\mathrm{f}_{-1}}(x).$$

Exactly as in the case with the (extended) binomial sums, GRPLs contain various subclasses that have been explored in depth. As we will see later in Section 2.2, these integrals can be related, e.g., to certain classes of sums by means of the Mellin transform.

[7] For more details, see for instance [8, 9].

Definition 2.1.15 (Harmonic polylogarithms)**.** *Let $n \in \mathbb{N}^\star$. We define the* ***harmonic polylogarithms*** *(HPL) as [8]:*

$$\mathrm{H}^*_{\emptyset}(x) = 1 \tag{2.1.83}$$

$$\mathrm{H}^*_{\mathrm{b}}(x) = \int_x^1 \mathrm{d}t\, \frac{1}{t-b} \tag{2.1.84}$$

$$\mathrm{H}^*_{\mathrm{b}_1,\ldots,\mathrm{b}_n}(x) = \int_x^1 \mathrm{d}t_1\, \frac{1}{t_1-b} \mathrm{H}^*_{\mathrm{b}_2,\ldots,\mathrm{b}_n}(x) \tag{2.1.85}$$

$$\mathrm{H}^*_{\mathbf{1}_n}(x) = \frac{\log(1-x)^n}{n!} \tag{2.1.86}$$

where $b, b_i \in \{0, -1, 1\}$. We call n the ***depth*** *of the HPL.*

Example 2.1.9. *Typical examples of harmonic polylogarithms are:*

$$\mathrm{H}^*_1(x) = \int_x^1 \mathrm{d}t\, \frac{1}{1-t} = \log(1-x)$$

$$\mathrm{H}^*_{1,0,1}(x) = \int_x^1 \mathrm{d}y\, \frac{1}{1-y} \int_y^1 \mathrm{d}t\, \frac{1}{t} \int_t^1 \mathrm{d}z\, \frac{1}{1-z}.$$

One extends the definition from harmonic polylogarithms to generalized polylogarithms by allowing values of b different from $-1, 0, 1$. More precisely, let us define the auxiliary function f_a for $a \in \mathbb{K}$ with

$$f_a(x) := \begin{cases} \frac{1}{x} & \text{if } a = 0 \\ \frac{1}{|a| - \mathrm{sign}(a) x} & \text{if } a \neq 0 \end{cases}. \tag{2.1.87}$$

Definition 2.1.16 (Generalized harmonic polylogarithms)**.** *Let $n \in \mathbb{N}^\star$. We define the* ***generalized harmonic polylogarithms*** *(GPL) as [12, 14]:*

$$\mathrm{H}^*_{\mathrm{b}}(x) = \int_x^1 \mathrm{d}t\, f_b(t) \tag{2.1.88}$$

$$\mathrm{H}^*_{\mathrm{b}_1,\ldots,\mathrm{b}_n}(x) = \int_x^1 \mathrm{d}t_1\, f_{b_1}(t_1) \mathrm{H}^*_{\mathrm{b}_2,\ldots,\mathrm{b}_n}(x) \tag{2.1.89}$$

$$\mathrm{H}^*_{\mathbf{1}_n}(x) = \frac{\log(1-x)^n}{n!} \tag{2.1.90}$$

where $b, b_i \in \mathbb{Z}$. We call n the ***depth*** *of the GPL.*

Example 2.1.10. *Examples of GPLs are:*

$$\mathrm{H}^*_{-3}(x) = \int_x^1 \mathrm{d}t\, \frac{1}{t+3} = \log(4) - \log(3+x)$$

$$\mathrm{H}^*_{7,0,-\frac{3}{2}}(x) = \int_x^1 \mathrm{d}t\, \frac{1}{7-t} \int_t^1 \mathrm{d}y\, \frac{1}{y} \int_y^1 \mathrm{d}z\, \frac{1}{z+\frac{3}{2}}.$$

Finally, associated to the cyclotomic sums, we can define the cyclotomic polylogarithms [10], that give this class of functions and sums its name. Indeed, if we define the cyclotomic polynomials as

$$\Phi_n(x) = \frac{x^n - 1}{\prod_{d|n, d<n} \Phi_d(x)}, \quad k, n \in \mathbb{N}^\star \tag{2.1.91}$$

we can introduce the alphabet $\mathfrak{U}$ defined as

$$\mathfrak{U} = \left\{ f_0^0(x) = \frac{1}{x} \right\} \cup \left\{ f_k^l(x) = \frac{x^l}{\Phi_k(x)} \middle| \, k \in \mathbb{N}^\star, 0 \leq l \leq \varphi(k) \right\} \tag{2.1.92}$$

where $\varphi(k)$ denotes the Euler totient function that counts the positive integers up to k that are relatively prime to k.

Definition 2.1.17 (Cyclotomic polylogarithms)**.** *Let $n, m \in \mathbb{N}$, let the set of letters $f_{k_i}^{l_i}$, $i \in \{1, \ldots, m\}$ be defined as above. We define the* ***cyclotomic polylogarithms*** *(CPL) as [10]:*

$$\begin{aligned}
\mathrm{C}_k^l(x) &= \int_x^1 \mathrm{d}t \, f_k^l(t) \\
\mathrm{C}_{k_1,\ldots,k_m}^{l_1,\ldots,l_m}(x) &= \int_x^1 \mathrm{d}t_1 \, f_{k_1}^{l_1}(t_1) \mathrm{C}_{k_2,\ldots,k_m}^{l_2,\ldots,l_m}(t_1) \\
\mathrm{C}_{\mathbf{1}_m}^{\mathbf{1}_m}(x) &= \frac{\log(1-x)^m}{m!}
\end{aligned}$$

where $l_i, k_i \in \mathbb{N}$. We call m the ***depth*** *of the CPL.*

Example 2.1.11. *Taking the cyclotomic polynomials*

$$\Phi_1(x) = x - 1, \; \Phi_2(x) = x + 1, \; \Phi_3(x) = x^2 + x + 1, \; \Phi_4(x) = x^2 + 1, \text{etc.},$$

we get the cyclotomic polylogarithms

$$\begin{aligned}
C_2^1(x) &= \int_x^1 \mathrm{d}t \, \frac{t}{1+t} \\
C_{0,12}^{0,2}(x) &= \int_x^1 \mathrm{d}t \int_t^1 \mathrm{d}y \, \frac{y^2}{1 - y^2 + y^4} \\
C_{4,5,6}^{1,2,3}(x) &= \int_x^1 \mathrm{d}t \, \frac{t}{1+t^2} \int_t^1 \mathrm{d}y \, \frac{y^2}{1 + y + y^2 + y^3 + y^4} \int_y^1 \mathrm{d}z \, \frac{z^3}{1 - z + z^2}.
\end{aligned}$$

2.1.2.1. A note on the integral representation of the central binomial coefficient

We will elaborate for which reason the central binomial coefficient gives such a nice Mellin representation, and why no reasonable representations for more complicated binomial coefficients are found. Let us start with the more general binomial coefficient:

$$\binom{pn}{n} = \frac{(pn)!}{n!((p-1)n)!}. \tag{2.1.93}$$

In the following we will use the identity

$$\forall p, n \in \mathbb{N} \quad \binom{n}{p} = \frac{1}{(n+1)\mathrm{B}(p+1, n-p+1)}$$

where B is the Beta function:

$$\mathrm{B}(\alpha, \beta) := \int_0^1 \mathrm{d}x\, x^{\alpha-1}(1-x)^{\beta-1}. \tag{2.1.94}$$

Also we will use the identity

$$\int_0^{\pi/2} \mathrm{d}t\, (\cos t)^{a-1} \cos(bt) = \frac{\pi}{2^a} \frac{1}{a\mathrm{B}\left(\frac{a+b+1}{2}, \frac{a-b+1}{2}\right)} \qquad \mathrm{Re}(a) > 0 \tag{2.1.95}$$

to represent the inverse Beta function in integral form. Now the idea is to combine both identities (2.1.94) and (2.1.95) in order to obtain an integral representation for the binomial coefficient. First we apply (2.1.94):

$$\forall p \geq 2, \quad \binom{pn}{n} = \frac{1}{(pn+1)\mathrm{B}(n+1, pn-n+1)}.$$

If we want to match the expression in (2.1.95), we have to solve:

$$\left\{ \begin{array}{rcl} n+1 & = & \frac{a+b+1}{2} \\ (p-1)n+1 & = & \frac{a-b+1}{2} \end{array} \right. \Rightarrow \left\{ \begin{array}{rcl} a & = & 1+np \\ b & = & (2-p)n \end{array} \right. .$$

We see that $a = np+1$ matches indeed the a term that appears in the denominator of (2.1.95). Therefore, multiplying by 2^{np+1}, we can write:

$$\forall p \geq 2, \quad \binom{pn}{n} = \frac{2^{pn+1}}{\pi} \int_0^{\frac{\pi}{2}} \mathrm{d}t\ (\cos t)^{pn} \cos\left((p-2)nt\right).$$

Let us make a trigonometric change of variables $x = \cos t$ to gain

$$\forall p \geq 2, \quad \binom{pn}{n} = \frac{2^{pn+1}}{\pi} \int_0^1 \mathrm{d}x\, \frac{x^{pn}}{\sqrt{1-x^2}} T_{(p-2)n}(x),$$

where $T_k(x)$ are the Chebychev polynomials [86] of the first kind, defined by

$$\forall x \in \mathbb{R} \quad T_n(x) = \frac{1}{2}\left[\left(x - \sqrt{x^2-1}\right)^n + \left(x + \sqrt{x^2-1}\right)^n\right]. \tag{2.1.96}$$

We make now a last change of variable $u = x^p$ to try to get as close as possible to an expression of the form of a Mellin transform:

$$\forall p \geq 2, \quad \binom{pn}{n} = \frac{2^{pn+1}}{p\pi}\int_0^1 \mathrm{d}u\, u^n u^{\frac{1-p}{p}}\left(1-u^{\frac{2}{p}}\right)^{-\frac{1}{2}} T_{(p-2)n}(u^{\frac{1}{p}}). \tag{2.1.97}$$

If we set $p = 2$, the dependence on n completely factors out and we see that the formula simplifies to

$$\binom{2n}{n} = \frac{4^n}{\pi}\int_0^1 \mathrm{d}x\, x^n \frac{1}{\sqrt{x(1-x)}} = \frac{4^n}{\pi}\mathbf{M}\left[f_{w_1}(x)\right](n) \tag{2.1.98}$$

which is the expression we find in equation (3.4) of [16]. The point is that whenever $p > 2$, we cannot completely factor out the n-dependence, since it occurs in the index of the Chebychev polynomial. Therefore, there seems no simple way to uncouple the $T_{(p-2)n}(u^{\frac{1}{p}})$ into two parts where one has a simple dependence on n and the other does not. Given that the idea of the Mellin inverse is to get rid of any kind of sum over n, this representation is therefore not helpful for the task at hand.

In order to obtain the integral representation for $\frac{1}{n\binom{pn}{n}}$, we have to use the following relation between the Beta function and the binomial coefficient:

$$\forall m, n \in \mathbb{N}, \quad \mathrm{B}(m,n) = \frac{m+n}{mn\binom{m+n}{m}}.$$

By identification, we get:

$$\frac{p}{p-1}\frac{1}{\binom{pn}{n}} = \mathrm{B}(n,(p-1)n) \Rightarrow \frac{1}{n\binom{pn}{n}} = \frac{p-1}{p}\mathrm{B}(n,(p-1)n).$$

Once again, the interesting case happens when $p = 2$, as then we get the following Laplace multiplication relation using the duplication formula [87] to get a formula relating integer and half-integer values for the Beta function [88]:

$$\forall n \in \mathbb{N}, \quad \mathrm{B}(n,n)\mathrm{B}\left(n+\frac{1}{2}, n+\frac{1}{2}\right) = \frac{\pi}{2^{4n-1}n}.$$

Applying this formula as well as (2.1.95) we get:

$$\mathrm{B}(n,n) = \frac{1}{4^{n-1}}\int_0^{\frac{\pi}{2}} \mathrm{d}t\, (\cos t)^{2n-1}.$$

We make two changes of variables

$$\int_0^{\frac{\pi}{2}} \mathrm{d}t\ (\cos t)^{2n-1} \underset{x=\cos t}{=} \int_0^1 \mathrm{d}x\ \frac{x^{2n-1}}{\sqrt{1-x^2}} \underset{u=x^2}{=} \frac{1}{2}\int_0^1 \mathrm{d}u\ \frac{u^n}{u\sqrt{1-u}}$$

and by combining everything together and simplifying the constants in front we get [16]:

$$\frac{1}{n\binom{2n}{n}} = \frac{1}{4^n}\int_0^1 \mathrm{d}x\ \frac{x^n}{x\sqrt{1-x}} = \frac{1}{4^n}\mathbf{M}\left[f_{w_3}(x)\right](n). \tag{2.1.99}$$

Remark 2.1.16. *The appearance of binomial coefficients on the sums side, and equivalently of square valued letters, lead to a complexification of the integral representations whatever the directions we try to extend the problems. Indeed, as considered in [11], when we consider coefficients of the form* $\binom{2n}{n}^2$ *on the sum side we have ellipitic coefficients of simple arguments, such as* $(1-x)$*, that appear on the integral side, which are quantities that are more difficult to deal with in principle. Extending this construction will open up the class of sums involving, e.g., also the letters* $\binom{2n}{n}^2$ *in Section 2.5.4. This case is not the modular case yet, as treated e.g. in reference [89]*[8].

2.2. Methods to compute Mellin convolutions

In the previous section we have introduced the binomial sums and generalized root-valued polylogarithms as a specific class of special functions (together with several subclasses). Our general aim is to relate functions to sums by computing Mellin transforms, and relating sums to integral by computing inverse Mellin transforms. In this section we study indepth the Mellin transform, which is the core ingredient that will help us to "build" inverse Mellin transforms in an algorithmic way, as we will describe in detail in the next section.

2.2.1. General convolution properties of Mellin transforms

One of the key properties of the Mellin transform is the multiplicative relation in Property 2.1.4, that we remind below:

$$\mathbf{M}\left[f(x)\right](n)\cdot\mathbf{M}\left[g(x)\right](n) = \mathbf{M}\left[(f\star g)(x)\right](n).$$

We follow the reasoning of [16]. Provided we can compute the convolution of the functions f and g, it is therefore possible to simplify a product of Mellin

[8] See also the references given in [71].

representations to a unique Mellin representation. Let us collect in this section several general formulas that we can derive for convolutions involving the GPLs. For convenience, we introduce, given a function $f(x)$, the notation

$$f(x)^{\star p} := \underbrace{f(x) \star \cdots \star f(x)}_{p \text{ times}}, \; f(x)^{\star 1} := f(x), \; f(x)^{\star 0} := 1. \tag{2.2.1}$$

Lemma 2.2.1. *Let f_a an integrable function on $[0;1]$, $p \in \mathbb{N}_{\geq 1}$, $\vec{w}$ a word. We have*

$$\frac{1}{x^{\star p}} \star [f_a(x) \mathrm{H}^*_{\tilde{\mathrm{w}}}(x)] = \frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}$$

where $\mathbf{a}_p := \underbrace{a, \cdots, a}_{p \text{ times}}$, $\mathbf{a}_0 := \emptyset$.

Proof. We prove the property by induction on p. We start with the case $p = 1$. Let $f(x) := f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)$. We need to check that f is integrable at $x = 1$. Given that f_a is integrable, it means that

$$f_a(x) \underset{x\to 1}{\sim} \frac{\alpha}{(1-x)^\beta}, \quad \alpha \in \mathbb{R}, \, \beta < 1.$$

In a similar way, due to (2.1.79), the divergence of $\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)$ will be at most of the form

$$\mathrm{H}^*_{\tilde{\mathrm{w}}}(x) \underset{x\to 1}{\sim} \gamma \log(1-x)^q, \quad \gamma \in \mathbb{R}, \, q \in \mathbb{N}.$$

Because $\log(1-x)^q (1-x)^{-\beta}$ for the given q, β is integrable at $x = 1$, f is integrable at $x = 1$. We can therefore compute the convolution

$$\begin{aligned} \frac{1}{x} \star f(x) &\underset{(2.1.28)}{=} \int_x^1 \mathrm{d}y \, \frac{1}{y} \frac{1}{\frac{x}{y}} f(y) = \frac{1}{x} \int_x^1 \mathrm{d}y \, f(y) \\ &= \frac{1}{x} \int_x^1 \mathrm{d}y \, f_a(x) \mathrm{H}^*_{\tilde{\mathrm{w}}}(y) \underset{(2.1.36)}{=} \frac{\mathrm{H}^*_{\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}. \end{aligned} \tag{2.2.2}$$

Suppose the property is true[9] for p. We have, by associativity of the Mellin convolution

$$\frac{1}{x^{\star(p+1)}} \star [f_a(x) \mathrm{H}^*_{\tilde{\mathrm{w}}}(x)] = \frac{1}{x} \star \left(\frac{1}{x^{\star p}} \star [f_a(x) \mathrm{H}^*_{\tilde{\mathrm{w}}}(x)] \right) \underset{(\mathrm{IH})}{=} \frac{1}{x} \star \frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}.$$

Because $\frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}$ is integrable at $x = 1$, we have reduced the problem down to the $p = 1$ case above, in (2.2.2), so identifying $f(x) = \frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}$ we get

$$\frac{1}{x} \star \frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x} = \frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}$$

[9] We denote the induction hypothesis with (IH).

which concludes the proof. □

Combining with the Mellin multiplicative formula we immediately get the following

Theorem 2.2.1. *If f_a is a function integrable on $[0;1]$, $p \in \mathbb{N}_{\geq 1}$, $\vec{w}$ a word then we have*

$$\frac{1}{n^p}\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right](n) = \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{a},\tilde{\mathrm{w}}}(x)}{x}\right](n). \tag{2.2.3}$$

Proof. We use the fact that

$$\frac{1}{n} = \mathbf{M}\left[\frac{1}{x}\right](n)$$

to write

$$\begin{aligned}\frac{1}{n^p}\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}(\mathrm{x})}\right](n) &= \left(\mathbf{M}\left[\frac{1}{x}\right](n)\right)^p \cdot \mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}(\mathrm{x})}\right](n)\\ &\underset{\mathrm{P}2.1.4}{=} \mathbf{M}\left[\frac{1}{x^{\star p}} \star \left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}(\mathrm{x})}\right]\right](n)\end{aligned}$$

and now using the Lemma 2.2.1 we can conclude the proof. □

Using this theorem as well as the Mellin inverse of the binomial coefficient given in (2.1.98) and the inverse binomial coefficient given in (2.1.99), we get immediately the following Mellin inverses

$$\frac{1}{n^p} = \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1}}(x)}{x}\right](n) \quad (p \geq 1) \tag{2.2.4}$$

$$\frac{1}{n^p}\binom{2n}{n} = \frac{4^n}{\pi}\mathbf{M}\left[\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{w}_1}(x)\right](n) \quad (p \geq 1) \tag{2.2.5}$$

$$\frac{1}{n^{p+1}\binom{2n}{n}} = \frac{1}{4^n}\mathbf{M}\left[\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\mathrm{w}_3}(x)\right](n) \quad (p \geq 1). \tag{2.2.6}$$

We can now extend those convolution relations to the case of certain rational functions, i.e., for cases when we have

$$\frac{p(n)}{q(n)} \cdot \mathbf{M}\left[f(x)\right](n)$$

where f is a function integrable[10] on $[0;1]$, $p, q \in \mathbb{C}[X]$ such that $\deg p \leq \deg q$ and for all roots $\alpha \in \mathbb{C}$ of q we have $\mathrm{Re}\,\alpha \leq 0$. This last property is essential to use the following observation. We know that for $\mathrm{Re}\,\beta \leq 1$, we have convergence of the

[10] Or integrable in the +-prescription sense at minima.

following integral:

$$\int_0^1 \mathrm{d}x \, \frac{1}{x^\beta} = \frac{1}{1-\beta}.$$

Therefore, by setting $\beta = \alpha+1-n$, for $n \in \mathbb{N}$, the integral stays always convergent if $\mathrm{Re}\,\alpha \leq 0$:

$$\mathbf{M}\left[\frac{1}{x^{\alpha+1}}\right](n) = \frac{1}{n-\alpha}. \tag{2.2.7}$$

Given a letter f_a and $\alpha \in \mathbb{C}$ we introduce the function

$$f_{\bar{\alpha}^a}(x) := x^\alpha f_a(x) \tag{2.2.8}$$

We can now extend Lemma 2.2.1.

Lemma 2.2.2. *Let f_a be an integrable function on $[0;1]$, $p \in \mathbb{N}_{\geq 1}$, $\vec{w}$ a word and $\alpha \in \mathbb{C}$ such that $\mathrm{Re}\,\alpha \leq 0$. Then we have*

$$\left(\frac{1}{x^{\alpha+1}}\right)^{\star p} \star [f_a(x)\mathrm{H}^*_{\vec{w}}(x)] = \frac{\mathrm{H}^*_{\mathbf{0}_{p-1},\bar{\alpha}^a,\vec{w}}(x)}{x^{\alpha+1}}.$$

Proof. We prove the lemma by induction on p. Similarly to Lemma 2.2.1 we know that $f_a(x)\mathrm{H}^*_{\vec{w}}(x)$ is integrable at $x=1$, so we can apply (2.1.28):

$$\begin{aligned}
\frac{1}{x^{\alpha+1}} \star [f_a(x)\mathrm{H}^*_{\vec{w}}(x)] &= \int_x^1 \mathrm{d}y \, \frac{1}{y}\left(\frac{y}{x}\right)^{\alpha+1} f(y)\mathrm{H}^*_{\vec{w}}(y) \\
&= \frac{1}{x^{\alpha+1}} \int_x^1 \mathrm{d}y \, \underbrace{y^\alpha f(y)}_{f_{\bar{\alpha}^a}} \mathrm{H}^*_{\vec{w}}(y) \\
&\underset{\text{Def. 2.1.13}}{=} \frac{\mathrm{H}^*_{\bar{\alpha}^a,\vec{w}}(x)}{x^{\alpha+1}}.
\end{aligned} \tag{2.2.9}$$

Now suppose the formula is true for $p \geq 1$. We have for $p+1$:

$$\begin{aligned}
\left(\frac{1}{x^{\alpha+1}}\right)^{\star(p+1)} \star [f_a(x)\mathrm{H}^*_{\vec{w}}(x)] &= \frac{1}{x^{\alpha+1}} \star \left[\left(\frac{1}{x^{\alpha+1}}\right)^{\star p} \star [f_a(x)\mathrm{H}^*_{\vec{w}}(x)]\right] \\
&\underset{\text{(IH)}}{=} \frac{1}{x^{\alpha+1}} \star \frac{\mathrm{H}^*_{\mathbf{0}_{p-1},\bar{\alpha}^a,\vec{w}}(x)}{x^{\alpha+1}}.
\end{aligned}$$

We are back to the case $p=1$ in (2.2.9) with $f_a(x) = \frac{1}{x^{\alpha+1}}$, and because of $f_{\bar{\alpha}^a}(x) = \frac{x^\alpha}{x^{\alpha+1}} = \frac{1}{x}$ we get

$$\frac{1}{x^{\alpha+1}} \star \frac{\mathrm{H}^*_{\mathbf{0}_{p-1},\bar{\alpha}^a,\vec{w}}(x)}{x^{\alpha+1}} = \frac{\mathrm{H}^*_{\mathbf{0}_{p},\bar{\alpha}^a,\vec{w}}(x)}{x^{\alpha+1}}$$

which concludes the proof. □

Theorem 2.2.2. *Let f_a be an integrable function on $[0;1]$, $p \in \mathbb{N}_{\geq 1}$, $\vec{w}$ a word and $\alpha \in \mathbb{C}$ such that $\operatorname{Re}\alpha \leq 0$. Then we have:*

$$\frac{1}{(n-\alpha)^p}\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right](n) = \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\bar{\alpha}^{\mathrm{a}},\tilde{\mathrm{w}}}(x)}{x^{\alpha+1}}\right](n). \tag{2.2.10}$$

Proof. We use the Mellin inverse

$$\frac{1}{n-\alpha} = \mathbf{M}\left[\frac{1}{x^{\alpha+1}}\right](n)$$

as well as the multiplicative formula in Property 2.1.4 to get

$$\begin{aligned}\frac{1}{(n-\alpha)^p}\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right](n) &= \mathbf{M}\left[\left(\frac{1}{x^{\alpha+1}}\right)^{\star p}\right](n)\cdot\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right](n)\\ &= \mathbf{M}\left[\left(\frac{1}{x^{\alpha+1}}\right)^{\star p}\star\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right]\right](n).\end{aligned}$$

Using the Lemma 2.2.2 this completes the proof. □

One important application is summarised in the following corollary.

Corollary 2.2.2.1. *Let f_a be an integrable function on $[0;1]$, $\vec{w}$ a word, and let $p, q \in \mathbb{C}[X]$ with $\deg p \leq \deg q$. Let $\alpha_1, \ldots, \alpha_r$ be the roots of q, with k_i the multiplicity of α_i and $\operatorname{Re}\alpha_i \leq 0$. Then*

$$\frac{p(n)}{q(n)}\mathbf{M}\left[f_a(x)\mathrm{H}^*_{\tilde{\mathrm{w}}}(x)\right](n) = \mathbf{M}\left[\sum_{i=1}^{r}\sum_{s=1}^{k_i}A_{i,s}\frac{\mathrm{H}^*_{\mathbf{0}_{\mathrm{p}-1},\bar{\alpha_{\mathrm{i}}}^{\mathrm{a}},\tilde{\mathrm{w}}}(x)}{x^{\alpha_i+1}}\right](n).$$

Proof. Applying partial fraction decomposition to p/q over $\mathbb{C}$, it yields

$$\frac{p(n)}{q(n)} = \sum_{i=1}^{r}\sum_{s=1}^{k_i}\frac{A_{i,s}}{(n-\alpha_i)^p}$$

where we have $A_{i,s} \in \mathbb{C}$. Thus the corollary follows by Theorem 2.2.2. □

2.2.2. Harmonic sums and plus-prescriptions

Some Mellin transforms have to be defined differently, using the **+-prescription**, which changes the way we can manipulate and convolve them. In particular, this happens when on the sum side we consider harmonic sums.

Example 2.2.1. *The simplest example we can take is that of the harmonic numbers*

of the first kind [7]:

$$S_1(n) = \sum_{i=1}^{n} \frac{1}{i} = \sum_{i=1}^{n} \mathbf{M}\left[\frac{1}{x}\right](i) \underset{(2.1.27)}{=} \int_0^1 \mathrm{d}x \, \frac{x^n - 1}{x - 1} \underset{(2.1.24)}{=} \mathbf{M}\left[\left(\frac{1}{x-1}\right)_+\right](n). \tag{2.2.11}$$

Here, we had to apply the Property 2.1.2, i.e., the resummation formula, by combining both integrals

$$\text{"}\mathbf{M}\left[\frac{x}{x-1}\frac{1}{x}\right](n)\text{"} \text{ and } \text{"}\mathbf{M}\left[\frac{x}{x-1}\frac{1}{x}\right](0)\text{"}$$

appearing in (2.1.27), since none of them is convergent individually. The form of the integral then forces us to use the Mellin +-prescription (2.1.24) which concludes this Mellin inversion process.

Next, using the result above, we try to deal with

$$S_{1,1}(n) = \sum_{i=1}^{n} \frac{1}{i} S_1(i) = \sum_{i=1}^{n} \mathbf{M}\left[\frac{1}{x}\right](i) \cdot \mathbf{M}\left[\left(\frac{1}{x-1}\right)_+\right](i) \tag{2.2.12}$$

and it turns out that to continue we need to perform the convolution that involves the +-prescription term. Remember that the initial definition of the Mellin convolution is

$$f(x) \star g(x) = \int_0^1 \mathrm{d}x_1 \int_0^1 \mathrm{d}x_2 \, \delta(x - x_1 x_2) f(x_1) g(x_2).$$

In the case when f is defined in the +-prescription, i.e., is more precisely a +-distribution, we integrate out one of the variables and Property 2.1.3 tells us that

$$\begin{aligned}[f(x)]_+ \star g(x) &= \int_x^1 \mathrm{d}y \, f\left(\frac{x}{y}\right)\left[\frac{1}{y} g(y) - \frac{x}{y^2} g(x)\right] - g(x) \int_0^x \mathrm{d}y \, f(y) \\ &= \int_x^1 \mathrm{d}y \, f(y)\left[\frac{1}{y} g\left(\frac{x}{y}\right) - g(x)\right] - g(x) \int_0^x \mathrm{d}y \, f(y).\end{aligned} \tag{2.2.13}$$

Thus we can perform the convolution from (2.2.12) to obtain

$$\begin{aligned}\left(\frac{1}{x-1}\right)_+ \star \frac{1}{x} &= \int_x^1 \mathrm{d}y \, \frac{1}{y-1}\left[\frac{1}{y}\frac{1}{\frac{x}{y}} - \frac{1}{x}\right] - \frac{1}{x}\int_0^x \mathrm{d}y \, \frac{1}{y-1} \\ &= -\frac{1}{x}\int_0^x \mathrm{d}y \, \frac{1}{y-1}.\end{aligned}$$

We would like now to express the result using a GPL, but the integral we have is from 0 to x and not x to 1 as in our convention. However, remember from Remark

2.1.12 that GPLs with non-integrable singularities around $x = 1$ are defined in a slightly different manner by

$$\mathrm{H}^*_{\mathrm{w}}(x) := \int_x^1 \mathrm{d}t \left(f_w(t) - \frac{c}{1-t} \right) + c \ln(1-x)$$
$$\mathrm{H}^*_{\mathrm{w},\tilde{\mathrm{w}}}(x) := \int_x^1 \mathrm{d}t \left(f_w(t) \mathrm{H}^*_{\tilde{\mathrm{w}}}(t) - \sum_{i=0}^{k} c_i \frac{\ln(1-t)^i}{1-t} \right) + \sum_{i=0}^{k} \frac{c_i}{i+1} \ln(1-x)^{i+1}$$

where $c, c_i \in \mathbb{R}$ are chosen uniquely such that the integrand on the RHS is integrable at $t = 1$. Using this result, we can relate the integral appearing in (2.2.12) to a known harmonic polylogarithm:

$$\mathrm{H}^*_1(x) = \int_x^1 \mathrm{d}t \left(\frac{1}{t-1} - \frac{c}{1-t} \right) + c \ln(1-x)$$
$$\underset{c=1}{=} \ln(1-x) = -\int_0^x \mathrm{d}t \, \frac{1}{1-y} = -\mathrm{H}_1(x). \quad (2.2.14)$$

More generally we have the following property.

Property 2.2.1. *For $p \in \mathbb{N}^\star$, we have [12]:*

$$\mathrm{H}^*_{1_\mathrm{p}}(x) = \frac{\ln(1-x)^p}{p!} = (-1)^p \mathrm{H}_{1_\mathrm{p}}(x) \quad (2.2.15)$$

$$\mathrm{H}^*_{0_\mathrm{p}}(x) = \frac{(-\ln x)^p}{p!} = (-1)^p \mathrm{H}_{0_\mathrm{p}}(x). \quad (2.2.16)$$

Proof. We show (2.2.15) by induction on p. The initialisation has been done in (2.2.14). We need to show that $\mathrm{H}^*_{\mathbf{1}_{n+1}}(x) = (-1)^{n+1} \mathrm{H}_{\mathbf{1}_{n+1}}(x)$. We write

$$\frac{\mathrm{d}}{\mathrm{d}x} \mathrm{H}^*_{\mathbf{1}_{n+1}}(x) = -\frac{\mathrm{H}^*_{\mathbf{1}_n}(x)}{1-x}$$
$$\text{induction hypothesis} \longrightarrow = (-1)^{n+1} \frac{\mathrm{H}_{\mathbf{1}_n}(x)}{1-x}$$
$$= (-1)^{n+1} \frac{\mathrm{d}}{\mathrm{d}x} \mathrm{H}_{\mathbf{1}_{n+1}}(x).$$

Therefore by direct integration $\mathrm{H}^*_{\mathbf{1}_{n+1}}(x) = (-1)^{n+1} \mathrm{H}_{\mathbf{1}_{n+1}}(x) + C$ with $C = \mathrm{H}^*_{\mathbf{1}_{n+1}}(0)$ as $\mathrm{H}_{\mathbf{1}_{n+1}}(0) = 0$. By the regularization given in (2.1.79), we also have that the constant $\mathrm{H}^*_{\mathbf{1}_{n+1}}(0) = \mathrm{H}_{\mathbf{1}_{n+1}}(0) = 0$, so $C = 0$ and this concludes the proof of (2.2.15). The proof for (2.2.16) follows in a similar way. □

We use this result to state the next property.

Property 2.2.2. *For $p \in \mathbb{N}$, we have*

$$\left(\frac{\mathrm{H}^*_{1_p}(x)}{x-1}\right)_+ \star \frac{1}{x} = -\frac{\mathrm{H}^*_{1_{p+1}}(x)}{x}. \tag{2.2.17}$$

Proof. The proof relies on (2.1.30) and (2.2.15):

$$\begin{aligned}
\left(\frac{\mathrm{H}^*_{1_p}(x)}{x-1}\right)_+ \star \frac{1}{x} &\underset{(2.1.30)}{=} \frac{1}{x}\int_0^x \mathrm{d}y\, \frac{\mathrm{H}^*_{\mathbf{1}_p}(y)}{1-y} \\
(2.2.15) \longrightarrow &= \frac{(-1)^p}{x}\int_0^x \mathrm{d}y\, \frac{\mathrm{H}_{\mathbf{1}_p}(y)}{1-y} \\
&= (-1)^p \frac{\mathrm{H}_{\mathbf{1}_{p+1}}(x)}{x} \\
(2.2.15) \longrightarrow &= (-1)^{p+p+1}\frac{\mathrm{H}^*_{\mathbf{1}_{p+1}}(x)}{x} = -\frac{\mathrm{H}^*_{\mathbf{1}_{p+1}}(x)}{x}
\end{aligned}$$

which concludes the proof. □

In particular, this means we can define immediately the following special convolution rules [9].

Property 2.2.3. *Let $p, q \geq 1$. We have*

$$S_{\mathbf{1}_p}(n) = (-1)^{p+1}\mathbf{M}\left[\left(\frac{\mathrm{H}^*_{\mathbf{1}_{p-1}}(x)}{x-1}\right)_+\right](n) \tag{2.2.18}$$

$$\frac{1}{n^q}S_{\mathbf{1}_p}(n) = (-1)^p\mathbf{M}\left[\frac{\mathrm{H}^*_{\mathbf{0}_{q-1},\mathbf{1}_p}(x)}{x}\right](n). \tag{2.2.19}$$

Proof. We prove the formula (2.2.18) by induction on p. The base case is done in (2.2.11). For $p > 1$ we proceed as follows:

$$\begin{aligned}
\mathrm{S}_{\mathbf{1}_{p+1}}(n) &= \sum_{i=1}^{n}\frac{1}{i}\mathrm{S}_{\mathbf{1}_p}(i) \\
\text{induction hypothesis} \longrightarrow &= (-1)^{p+1}\sum_{i=1}^{n}\mathbf{M}\left[\frac{1}{x}\right](n)\cdot\mathbf{M}\left[\left(\frac{\mathrm{H}^*_{\mathbf{1}_{p-1}}(x)}{x-1}\right)_+\right](n) \\
(2.2.17) \longrightarrow &= (-1)^{p+1}\mathbf{M}\left[-\frac{\mathrm{H}^*_{\mathbf{1}_{p+1}}(x)}{x}\right](n) \\
(2.1.2) \longrightarrow &= (-1)^{p+2}\mathbf{M}\left[\left(\frac{\mathrm{H}^*_{\mathbf{1}_p}(x)}{x-1}\right)_+\right](n).
\end{aligned}$$

The proof of (2.2.19) follows immediately from (2.2.18) combined together with Theorem 2.2.1. □

Example 2.2.2. *In particular, coming back to identity (2.2.11) we can finally conclude that*

$$S_{1,1}(n) = -\sum_{i=1}^{n} \mathbf{M}\left[\left(\frac{\mathrm{H}_1^*(x)}{x-1}\right)_+\right](i) = -\int_0^1 \mathrm{d}x\, \frac{x^n-1}{x-1}\mathrm{H}_1^*(x).$$

If we were dealing only with generalized S-sums, having this result would already be quite satisfactory, but because we also consider binomial coefficients, we have to consider two additional classes of sums which lead to +-prescription convolutions.

Theorem 2.2.3. *For $n, p \in \mathbb{N}^\star$, we have*

$$\binom{2n}{n} S_{1_p}(n) = (-1)^p \frac{4^n}{\pi}\mathbf{M}\left[I_{1,p-1}(x) - \frac{\mathrm{H}_{1_\mathrm{p}}^*(x)}{\sqrt{x(1-x)}}\right](n) \tag{2.2.20}$$

$$\frac{1}{n\binom{2n}{n}} S_{1_p}(n) = (-1)^p \frac{1}{4^n}\mathbf{M}\left[I_{2,p-1}(x) - \frac{\mathrm{H}_{1_\mathrm{p}}^*(x)}{x\sqrt{1-x}}\right](n) \tag{2.2.21}$$

where

$$I_{1,p}(x) := \frac{1}{p!}\int_x^1 \mathrm{d}y\, \frac{\ln(1-y)^p}{y-1}\left(\frac{1}{\sqrt{x(y-x)}} - \frac{1}{\sqrt{x(1-x)}}\right) \tag{2.2.22}$$

$$I_{2,p}(x) := \frac{1}{p!}\int_x^1 \mathrm{d}y\, \frac{\ln(1-y)^p}{y-1}\left(\frac{1}{x}\sqrt{\frac{y}{y-x}} - \frac{1}{x\sqrt{1-x}}\right). \tag{2.2.23}$$

Proof. Using the Property 2.2.3 as well as the Mellin inverses of (inverse) binomial coefficients obtained in (2.1.98) and (2.1.99), we write

$$\binom{2n}{n} S_{1_p}(n) = (-1)^{p+1}\frac{4^n}{\pi}\mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](n) \cdot \mathbf{M}\left[\left(\frac{\mathrm{H}_{1_{\mathrm{p}-1}}^*(x)}{x-1}\right)_+\right](n) \tag{2.2.24}$$

$$\frac{1}{n\binom{2n}{n}} S_{1_p}(n) = (-1)^{p+1}\frac{1}{4^n}\mathbf{M}\left[\frac{1}{x\sqrt{1-x}}\right](n) \cdot \mathbf{M}\left[\left(\frac{\mathrm{H}_{1_{\mathrm{p}-1}}^*(x)}{x-1}\right)_+\right](n). \tag{2.2.25}$$

We perform the +-prescription convolution explicitely for (2.2.24):

$$\left(\frac{\mathrm{H}_{1_{\mathrm{p}-1}}^*(x)}{x-1}\right)_+ \star \frac{1}{\sqrt{x(1-x)}} = \frac{1}{(p-1)!}\int_x^1 \mathrm{d}y\, \frac{\mathrm{H}_{1_{\mathrm{p}-1}}^*(y)}{y-1}\left[\frac{1}{y}\frac{1}{\sqrt{\frac{x}{y}\left(1-\frac{x}{y}\right)}} - \frac{1}{\sqrt{x(1-x)}}\right]$$
$$- \frac{1}{\sqrt{x(1-x)}}\int_0^x \mathrm{d}y\, \frac{\mathrm{H}_{1_{\mathrm{p}-1}}^*(y)}{y-1}.$$

Now using that $\mathrm{H}^*_{1_p}(x) = \ln(1-x)^p/p!$, we recognize the regularizing integral (2.2.22) and using the relation $\mathrm{H}^*_{1_p}(x) = (-1)^p \mathrm{H}_{1_p}(x)$, we can simplify the convolution to:

$$\binom{2n}{n} S_{1_p}(n) = (-1)^p \frac{4^n}{\pi} \mathbf{M}\left[I_{1,p-1}(x) - \frac{\mathrm{H}^*_{1_p}(x)}{\sqrt{x(1-x)}} \right](n).$$

In the same way, we perform the +-prescription convolution explicitely for (2.2.25):

$$\left(\frac{\mathrm{H}^*_{1_{p-1}}(x)}{x-1} \right)_+ \star \frac{1}{x\sqrt{1-x}} = \frac{1}{(p-1)!} \int_x^1 \mathrm{d}y \, \frac{\ln(1-y)^{p-1}}{y-1} \left[\frac{1}{y} \frac{1}{\frac{x}{y}\sqrt{1-\frac{x}{y}}} - \frac{1}{x\sqrt{1-x}} \right] - \frac{1}{x\sqrt{1-x}} \int_0^x \mathrm{d}y \, \frac{\mathrm{H}^*_{1_{p-1}}(y)}{y-1}.$$

Once again we recongize the regularisation integral (2.2.23) and using the relation $\mathrm{H}^*_{1_p}(x) = (-1)^p \mathrm{H}_{1_p}(x)$, we can simplify the convolution to:

$$\frac{1}{n\binom{2n}{n}} S_{1_p}(n) = (-1)^p \frac{1}{4^n} \mathbf{M}\left[I_{2,p-1}(x) - \frac{\mathrm{H}^*_{1_p}(x)}{x\sqrt{1-x}} \right](n)$$

which concludes the proof. □

Therefore, the problem now boils down to the computation of the two integral classes $I_{1,p}(x)$ and $I_{2,p}(x)$ in Theorem 2.2.3. It turns out that while very similar expression-wise, one of them turns out to be significantly more complicated to compute than the other. More precisely, to compute $I_{p,1}(n)$, we can simply plug the integral into Mathematica and it manages to compute it up to relatively high $p \in \mathbb{N}$. The values of $I_{1,p}(x)$ have been computed for test purpose up to $p = 10$. Below are given the explicit values for $p \leq 5$:

$$I_{0,1}(x) = -\frac{\log(4)}{\sqrt{(1-x)x}}$$

$$I_{1,1}(x) = \frac{\pi^2 - 3\log^2(4)}{6\sqrt{(1-x)x}} - \frac{\log(4)\log(1-x)}{\sqrt{(1-x)x}}$$

$$I_{2,1}(x) = \frac{-12\zeta_3 - \log^3(4) + \pi^2\log(4)}{6\sqrt{(1-x)x}} - \frac{\log(64)\log^2(1-x)}{6\sqrt{(1-x)x}} + \frac{\left(\pi^2 - 3\log^2(4)\right)\log(1-x)}{6\sqrt{(1-x)x}}$$

$$I_{3,1}(x) = \frac{-240\zeta_3\log(4-4x) + 3\pi^4 - 5\log^4(4) + 10\pi^2\log^2(4)}{120\sqrt{(1-x)x}} - \frac{\log(4)\log^3(1-x)}{6\sqrt{(1-x)x}} + \frac{\left(\pi^2 - 3\log^2(4)\right)\log^2(1-x)}{12\sqrt{(1-x)x}} + \frac{(\pi - \log(4))\log(4)(\pi + \log(4))\log(1-x)}{6\sqrt{(1-x)x}}$$

$$I_{4,1}(x) = \frac{\log(1-x)\left(-240\zeta_3\log(4) + 3\pi^4 - 5\log^4(4) + 10\pi^2\log^2(4)\right)}{120\sqrt{(1-x)x}}$$

$$
\begin{aligned}
&+\frac{\log^2(1-x)\left(-12\zeta_3-\log^3(4)+\pi^2\log(4)\right)}{12\sqrt{(1-x)x}}\\
&+\frac{120\pi^2\zeta_3-2160\zeta_5-360\zeta_3\log^2(4)-3\log^5(4)+10\pi^2\log^3(4)+9\pi^4\log(4)}{360\sqrt{(1-x)x}}\\
&-\frac{\log(4)\log^4(1-x)}{24\sqrt{(1-x)x}}+\frac{\left(\pi^2-3\log^2(4)\right)\log^3(1-x)}{36\sqrt{(1-x)x}}\\
I_{5,1}(x)=&\frac{\log^3(1-x)\left(-12\zeta_3-\log^3(4)+\pi^2\log(4)\right)}{36\sqrt{(1-x)x}}\\
&+\frac{\log^2(1-x)\left(-240\zeta_3\log(4)+3\pi^4-5\log^4(4)+10\pi^2\log^2(4)\right)}{240\sqrt{(1-x)x}}\\
&+\frac{\log(1-x)\left(120\pi^2\zeta_3-2160\zeta_5-360\zeta_3\log^2(4)-3\log^5(4)+10\pi^2\log^3(4)+9\pi^4\log(4)\right)}{360\sqrt{(1-x)x}}\\
&+\frac{-30240\zeta_3^2-5040\zeta_3\log^3(4)+5040\pi^2\zeta_3\log(4)-90720\zeta_5\log(4)+}{15120\sqrt{(1-x)x}}\\
&+\frac{79\pi^6-21\log^6(4)+105\pi^2\log^4(4)+189\pi^4\log^2(4)}{15120\sqrt{(1-x)x}}\\
&-\frac{\log(4)\log^5(1-x)}{120\sqrt{(1-x)x}}+\frac{\left(\pi^2-3\log^2(4)\right)\log^4(1-x)}{144\sqrt{(1-x)x}}.
\end{aligned}
$$

However, when we try to compute explicitely $I_{2,p}(x)$, Mathematica only manages to perform the calculation for $p=0$, which gives us

$$
I_{0,2}(x)=\frac{\frac{\log\left(\frac{x}{4}\right)}{\sqrt{1-x}}+2\tanh^{-1}\left(\sqrt{1-x}\right)}{x} \tag{2.2.26}
$$

and for the rest, it fails. We describe here the following method, that allowed us to obtain explicity the cases $p=1,2$ and can be in principle extended for the cases $p\geq 3$, modulo technical limitations.

1. First we make a change of variable to get rid of the x in the bound, i.e., turn the bounds to $[0;1]$ instead of $[x;1]$. We get

$$
I_{1,2}(x)=\int_0^1 \mathrm{d}y\,\mathcal{I}_{p,2}(x,y)
$$

with

$$
\mathcal{I}_{p,2}(x,y)=-\frac{x\log^p\left(1-\frac{x}{(x-1)y+1}\right)}{\left(\sqrt{y}+1\right)\left((x-1)y+1\right)\sqrt{y-xy}}\quad\text{with } y\to\frac{x}{1+(x-1)y}
$$

The special cases $p=1,2$ read as follows:

- Case $p = 1$:

$$I_{1,2}(x) = \int_0^1 \mathrm{d}y \; - \frac{x \log\left(1 - \frac{x}{(x-1)y+1}\right)}{\left(\sqrt{y}+1\right)((x-1)y+1)\sqrt{y-xy}} \quad \text{with } y \to \frac{x}{1+(x-1)y} \tag{2.2.27}$$

- Case $p = 2$:

$$I_{2,2}(x) = \int_0^1 \mathrm{d}y \; - \frac{x \log^2\left(1 - \frac{x}{(x-1)y+1}\right)}{\left(\sqrt{y}+1\right)((x-1)y+1)\sqrt{y-xy}} \quad \text{with } y \to \frac{x}{1+(x-1)y} \tag{2.2.28}$$

2. This allows us to apply Christoph Koutschan's `HolonomicFunctions` package [90, 91], in particularly using the `CreativeTelescoping` function in order to find the (inhomegenous) differential equation associated to the integrals.
   - For the case $p = 1$, we get the differential equation

$$\left(4x^3 - 8x^2 + 4x\right) f^{(4)}(x) + \left(32x^2 - 40x + 8\right) f^{(3)}(x) + (55x - 30) f''(x) + 15 f'(x) = -\frac{-3x + x\log(1-x) + 3}{(1-x)^{3/2}x}$$

     of order 4.
   - For the case $p = 2$, we get the differential equation

$$\begin{aligned}&\left(8x^5 - 24x^4 + 24x^3 - 8x^2\right) f^{(6)}(x) \\ &\quad + \left(156x^4 - 360x^3 + 252x^2 - 48x\right) f^{(5)}(x) \\ &\quad + \left(926x^3 - 1514x^2 + 636x - 48\right) f^{(4)}(x) \\ &\quad + \left(1911x^2 - 1904x + 336\right) f^{(3)}(x) \\ &\quad + (1155x - 490) f''(x) + 105 f'(x) \\ &= \frac{2x(33 - 7x) - x\log(1-x)(-18x + (2x+1)\log(1-x) + 20) - 30}{(1-x)^{5/2}x}\end{aligned}$$

     of order 6.
   - We have also managed to obtain the differential equations up to $p = 5$, with $p = 5$ taking a few days to be produced. Below we summarize the orders of the differential equations:

p	1	2	3	4	5
order	4	6	8	10	12

3. Once we get this equation, suppose of order d, we can simply feed it to the function `SolveDE` from Jakob Ablinger's `HarmonicSums` [9,10,12,72–74], and get the solution

$$f(x) = \sum_{i=1}^{d} \alpha_i h_i(x) + p(x)$$

where $h_i(x)$ are homogeneous solutions, $p(x)$ is the particular solution and $\alpha_i \in \mathbb{K}$ are the constants we search for.

- For[11] $p = 1$, we get the following homogenous solutions

$$h_1(x) = 1, \quad h_2(x) = \frac{2i}{\sqrt{1-x}}, \quad h_3(x) = -\frac{2\left(\mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda}}(x) - \sqrt{1-x}\mathrm{H}_{\frac{1}{\lambda}}(x)\right)}{\sqrt{1-x}}$$

$$h_4(x) = -\frac{4i\left(-\mathrm{H}_{\frac{1}{1-\lambda}}(x) - \mathrm{H}_{\frac{1}{\lambda}}(x) + \sqrt{1-x}\mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda}}(x) + 2x\right)}{3\sqrt{1-x}}$$

and the following particular solution

$$\begin{aligned} p(x) = &-\mathrm{H}_{\frac{1}{\lambda},\frac{\sqrt{1-\lambda}}{\lambda}}(x) + \mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda},\frac{1}{1-\lambda}}(x) + \mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda},\frac{1}{\lambda}}(x) + \frac{\mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda},\frac{\sqrt{1-\lambda}}{\lambda}}(x)}{\sqrt{1-x}} \\ &- \frac{\mathrm{H}_{\frac{1}{1-\lambda},\frac{1}{\lambda}}(x)}{\sqrt{1-x}} - \frac{\mathrm{H}_{\frac{1}{\lambda},\frac{1}{1-\lambda}}(x)}{\sqrt{1-x}} - \frac{\mathrm{H}_{\frac{1}{\lambda},\frac{1}{\lambda}}(x)}{\sqrt{1-x}} + \frac{2(3x-1)\mathrm{H}_{\frac{1}{1-\lambda}}(x)}{3\sqrt{1-x}} \\ &+ \frac{2\left(75x + 99\sqrt{1-x} - 25\right)\mathrm{H}_{\frac{1}{\lambda}}(x)}{75\sqrt{1-x}} \\ &+ \frac{2\left(25\sqrt{1-x} - 99\right)\mathrm{H}_{\frac{\sqrt{1-\lambda}}{\lambda}}(x)}{75\sqrt{1-x}} + \frac{2\left(6\sqrt{1-x} - x\right)}{3\sqrt{1-x}}. \end{aligned}$$

4. In addition we have to compute a certain number of initial conditions. We expand up to the required order k around some particular point x, and then integrate this expansion. For example, we can expand around $x = 1/2$:

$$\mathcal{I}_{p,2}(x,y) \underset{x\to\frac{1}{2}}{=} \sum_{i=0}^{k} \mathcal{I}_{i,p,2}(y)\left(x-\frac{1}{2}\right)^i + \mathcal{O}\left(\left(x-\frac{1}{2}\right)^{k+1}\right).$$

[11] Due to the size of the solutions and further coefficients, we restrict the example we show to $p = 1$ from now on.

Note that the more preferable points $x = 0$ or $x = 1$ are not suitable, because there are either Fubini theorem application issues, i.e., the integral of the expansions contains factors that cannot match those we obtain from the expansion of the general solution $f(x)$ (such as logarithmic factors), or altogether the expansion is non-integrable. The particular point $x = 1/2$ is chosen as the third "simplest" point after $x = 0, 1$.

5. We can now also expand the solution up to the same order around $x = 1/2$:

$$f(x) = \sum_{i=0}^{k} \left[\sum_{j=1}^{d} \frac{\alpha_j}{i!} h_j^{(i)}\left(\frac{1}{2}\right) + \frac{p^{(i)}\left(\frac{1}{2}\right)}{i!} \right].$$

Combining these two together, i.e., by matching the corresponding coefficients, we can build a linear system in the α_i that we can then solve to obtain the full solution $f(x)$

Remark 2.2.1. *This different processes (computing the integrals of the expansion of the original integral and computing the expansion of the solution) give rise to many intermediate constants, which turn out to be the bottleneck for the computation of convolution of higher order expressions. We have managed to compute the two cases $p = 1$ and $p = 2$. For larger p the expressions got too cumbersome, due to the appearance of intermediate constants, to be able to be dealt with easily in Mathematica.*

Remark 2.2.2. *In order to simplify the many constants that pop up during the steps 4 and 5, after all possible standard simplifications, for example using identities between special functions, we used the PSLQ algorithm [92] to try to find integer relations between the constants. Because the list of constants was usually quite large, we used a custom function called* `FindPSLQRelations` *which is implemented in* `RICA`[12].

For $p = 0$, the solution obtained through direct integration in Mathematica and rewritten using the GRPLs is:

$$I_{2,0}(x) = -\frac{\mathrm{H}^*_{\mathrm{f}_0}(x)}{\sqrt{1-x}x} + \frac{\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3}}(x)}{x} - \frac{2\log(2)}{\sqrt{1-x}x}. \tag{2.2.29}$$

Here below we write down the solutions obtained using the method above for $p = 1$:

$$I_{2,1}(x) = -\frac{\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_1}(x)}{\sqrt{1-x}x} - \frac{\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3}}(x)}{x} - \frac{2\mathrm{H}^*_{\mathrm{f}_0}(x)\log(2)}{\sqrt{1-x}x}$$

[12] See Appendix A.4.1.

$$
\begin{aligned}
&- \frac{\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{2\mathrm{H}^*_{\mathrm{f}_1}(x)\log(2)}{\sqrt{1-x}x} + \frac{\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0}(x)}{x} + \frac{\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1}(x)}{x} \\
&+ \frac{\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3}}(x)}{\sqrt{1-x}x} + \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3}}(x)\log(2)}{x} + \frac{\pi^2 - 12\log^2(2)}{6\sqrt{1-x}x}
\end{aligned} \tag{2.2.30}
$$

and finally for $p = 2$:

$$
\begin{aligned}
I_{2,2}(x) = &- \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1}(x)}{x} + \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0,\mathrm{f}_1}(x)}{x} + \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1,\mathrm{f}_0}(x)}{x} - \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_0,\mathrm{f}_1}(x)}{\sqrt{1-x}x} \\
&- \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_1,\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_1,\mathrm{f}_1}(x)}{\sqrt{1-x}x} - \frac{2\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_0,\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{2\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_0,\mathrm{f}_1}(x)}{\sqrt{1-x}x} \\
&- \frac{2\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_1,\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_1}(x)}{\sqrt{1-x}x} - \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_0}(x)}{\sqrt{1-x}x} + \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3}}(x)}{x} \\
&- \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0}(x)}{x} + \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3}}(x)}{\sqrt{1-x}x} + \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0,\mathrm{f}_0}(x)}{x} - \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3}}(x)}{\sqrt{1-x}x} \\
&+ \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0}(x)}{\sqrt{1-x}x} - \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_{\mathrm{w}_3}}(x)}{x} + \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_0}(x)}{x} - \frac{2\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_0,\mathrm{f}_0}(x)}{\sqrt{1-x}x} \\
&+ \frac{\left(\pi^2 - 12\log^2(2)\right)\mathrm{H}^*_{\mathrm{f}_0}(x)}{3\sqrt{1-x}x} - \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_0,\mathrm{f}_0}(x)}{\sqrt{1-x}x} + \frac{2\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3}}(x)}{\sqrt{1-x}x} \\
&+ \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1,\mathrm{f}_1}(x)}{x} + \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1}(x)}{\sqrt{1-x}x} + \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_1}(x)}{x} \\
&+ \frac{\left(\pi^2 - 12\log^2(2)\right)\mathrm{H}^*_{\mathrm{f}_1}(x)}{3\sqrt{1-x}x} - \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_1,\mathrm{f}_1}(x)}{\sqrt{1-x}x} - \frac{2\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3}}(x)}{x} \\
&+ \frac{\left(12\log^2(2) - \pi^2\right)\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3}}(x)}{3x} + \frac{4\log(2)\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_3},\mathrm{f}_{\mathrm{w}_3}}(x)}{\sqrt{1-x}x} \\
&+ \frac{2\left(-6\zeta(3) - 4\log^3(2) + \pi^2\log(2)\right)}{3\sqrt{1-x}x}.
\end{aligned} \tag{2.2.31}
$$

2.3. A simple recursive algorithm for Mellin inversion of iterated sums

In Section 2.1 we introduced on the one hand different classes of sums and on the other hand different classes of iterated integrals, and in Section 2.2, we studied special cases of Mellin convolutions and succeeded in performing explicitely a Mellin inversion by means of integral representations. Using the previously obtained results, we can now summarise everything in a simple algorithm that performs Mellin

inversions fully automatically. We have implemented this algorithm in our `RICA` package.

2.3.1. The classical method versus our recursive method

Before delving into our recursive method, we present the classical method that uses holonomic functions and sequences and that is developed in [17] and [18].

Definition 2.3.1 (Holonomic function). *A function $f(x)$ is called **holonomic** (also **D-finite**) if there exist polynomials $p_0(x), p_1(x), \dots, p_d(x) \in \mathbb{K}[x]$ (not all p_i being 0) such that the following holonomic differential equation holds:*

$$p_d(x)\frac{\mathrm{d}^d f(x)}{\mathrm{d}x^d} + p_{d-1}(x)\frac{\mathrm{d}^{d-1} f(x)}{\mathrm{d}x^{d-1}} + \cdots + p_0(x) f(x) = 0. \tag{2.3.1}$$

Definition 2.3.2 (Holonomic sequence). *A sequence $(f_n)_{n\geq 0} \in \mathbb{K}^{\mathbb{N}}$ is called **holonomic** (or **P-finite**) if there exist polynomials $p_0(n), p_1(n), \dots, p_d(n) \in \mathbb{K}[n]$ (not all p_i being 0) such that the following holonomic recurrence holds:*

$$p_d(n) f_{d+n} + p_{d-1}(n) f_{d+n-1} + \cdots + p_0(n) f_n = 0. \tag{2.3.2}$$

In order to compute the Mellin inverse of a certain sum $S(n)$ this classical algorithm[13] proceeds in the following way:

1. Compute an holonomic recurrence for $S(n)$, seen as the Mellin transform of an holonomic function $f(x)$, i.e., we identify $S(n) = \mathbf{M}[f(x)](n)$.
2. Obtain the associated holonomic differential equation for $f(x)$ using identities between Mellin transform of derivatives of a functions and shifted Mellin transforms [17].
3. Compute a linear independent set of solutions of the differential equation.
4. Compute initial values of $\mathbf{M}[f(x)](n)$.
5. Combine all of this to get a closed form for $f(x)$, concluding the Mellin inversion.

This method is very general, because one can obtain rather easily the (holonomic) differential equation that the inverse Mellin transform obeys. In particular, Ablinger's `HarmonicSums` package can find all d'Alembertian solutions[14] and also solve differential equations that factorizes to linear factors up to one second order

[13] An implementation by Jakob Ablinger exists in `HarmonicSums` [17, 18].
[14] See Definition 3.1.2.

factor[15]. However, if we cannot solve the differential equation we are stuck. The second shortcoming comes from the fact that even provided we can solve the differential equation, it might be challenging to obtain explicit initial values to combine the homogeneous solutions accordingly.

Remark 2.3.1. *Actually, the method to compute the Mellin transform of a function $f(x)$, i.e., the reverse operation of what we just describe, follows a very similar method [17]:*

- *Compute a holonomic differential equation for $f(x)$.*
- *Obtain the associated holonomic recurence for $\mathbf{M}[f(x)](n)$ using Mellin identities relating derivates and shifts [17].*
- *Compute initial values for the recurrence.*
- *Solve the recurrence using* `Sigma` *[94] in terms of d'Alembertian solutions[16] to get a closed form representation for $\mathbf{M}[f(x)](n)$.*

An alternative method, that we properly put in algorithmic form and implemented in `RICA`, follows a very different path as we will see below, since it computes the Mellin inverse recursively from the binomial sums and uses fundamental properties of the Mellin transform studied in the sections above. It is therefore much more straightforward and relies on very simple principles, but the main shortcoming is that different cases have to be identified and implemented individually. However, inspired by this method we will derive tools in Section 2.4.2 that enable us to compute in Section 2.5.4 asymptotic expansions of sums that cannot be derived from the differential equation approach.

Remark 2.3.2. *We note that there exists a more recent method to compute Mellin inverses described in [13]. This method relies on resummation of local operators on the physical side, and mathematically involves using computation of generating functions $\tilde{F}(t) = \sum_{n=0}^{\infty} t^n \mathbf{M}[f(x)](n)$, called t-representations, for a given sum $S(n) = \mathbf{M}[f(x)](n)$. In order to obtain $f(x)$, i.e., compute the Mellin inverse of $S(n)$, one then relies on complex analysis relations on the one hand — in particular computing the discontinuity of given functions across the branch cut induced by some poles — and on the other hand solving differential equations. While the path is different, because we still solve differential equations, we get similar advantages and disadvantages to the method proposed before.*

[15] Solving second order linear homogeneous differential equations uses the so called "Kovacic algorithm" described in [93] and then studied further and applied to Mellin inversion in [18].
[16] See [16, 95–98].

2.3.2. The general method

Consider as usual the binomial sum

$$\mathcal{BS}_{(a_1,\ldots,a_k)}(n) = \sum_{i_1=1}^{n} a_1(i_1) \sum_{i_2=1}^{i_1} a_2(i_2) \cdots \sum_{i_k=1}^{i_{k-1}} a_k(i_k).$$

In order to compute its Mellin inverse, we proceed in the following way:

Base case: We need to represent the innermost summand $a_k(i_k)$ as a Mellin integral

The initial step consists to look individually at the innermost summand, $a_k(i_k)$. Because of the definition of a binomial sum, it takes the form

$$a_p(i; b, c, m) = \binom{2i}{i}^b \frac{c^i}{i^m}, \ b \in \{-1, 0, 1\}, \ c \in \mathbb{K}^\star, \ m \in \mathbb{N};$$

given in (2.1.4). So to be able to obtain its Mellin inverse, it suffices to know some "basic building blocks" and then use the rules from Section 2.2.1. By "basic building blocks", we mean Mellin inverses of the central binomial coefficient and some associated quantities [16]:

$$\frac{1}{n} = \mathbf{M}\left[\frac{1}{x}\right](n) = \mathbf{M}\left[f_0(x)\right](n) \tag{2.3.3}$$

$$\binom{2n}{n} = \frac{4^n}{\pi}\mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](n) = \frac{4^n}{\pi}\mathbf{M}\left[f_{w_1}(x)\right](n) \tag{2.3.4}$$

$$\frac{1}{n\binom{2n}{n}} = \frac{1}{4^n}\mathbf{M}\left[\frac{1}{x\sqrt{1-x}}\right](n) = \frac{1}{4^n}\mathbf{M}\left[f_{w_3}(x)\right](n) \tag{2.3.5}$$

$$\frac{1}{(2n+1)\binom{2n}{n}} = \frac{1}{2}\frac{1}{4^n}\mathbf{M}\left[\frac{1}{\sqrt{1-x}}\right](n) \tag{2.3.6}$$

$$\frac{1}{(n+1)\binom{2n}{n}} = \frac{1}{4^n}\mathbf{M}\left[\frac{1}{\sqrt{1-x}} - \frac{\mathrm{H}^*_{w_3}(x)}{2}\right](n) \tag{2.3.7}$$

Proof. The proof for (2.3.3) proceeds as follows:

$$\mathbf{M}\left[\frac{1}{x}\right](n) \underset{(2.1.8)}{=} \int_0^1 \mathrm{d}x\, x^n \frac{1}{x} = \left[\frac{x^n}{n}\right]_0^1 = \frac{1}{n} \tag{2.3.8}$$

which concludes the proof. □

Proof. The proof for (2.3.4) is shown in Section 2.1.2.1. □

Proof. The proof for (2.3.5) is shown in Section 2.1.2.1. □

Proof. To prove (2.3.6), we use (2.3.5) where we shift $n \to n+1$, so that

$$\begin{aligned}\frac{1}{4^{n+1}}\mathbf{M}\left[\frac{1}{x\sqrt{1-x}}\right](n+1) &\underset{(2.1.1)}{=} \frac{1}{4^{n+1}}\mathbf{M}\left[\frac{1}{\sqrt{1-x}}\right](n)\\ &= \frac{1}{(n+1)\binom{2(n+1)}{n+1}}\\ &= \frac{1}{(2n+1)\binom{2n}{n}}.\end{aligned}$$

Multiplying both sides by 2 we obtain (2.3.7). □

Proof. Finally, to prove (2.3.7), we notice that

$$(n+1)^2\binom{2(n+1)}{n+1} = 2(n+1)(2n+1)\binom{2n}{n}$$

and

$$\frac{2}{(2n+1)\binom{2n}{n}} - \frac{1}{(n+1)(2n+1)\binom{2n}{n}} = \frac{\cancel{2n+2-1}}{\cancel{(2n+1)}(n+1)\binom{2n}{n}} = \frac{1}{(n+1)\binom{2n}{n}}.$$

Using the shifted version of (2.3.5) (like in the proof for (2.3.6)) as well as Theorem 2.2.1 we get

$$\begin{aligned}\frac{1}{(n+1)\binom{2n}{n}} &= \frac{4}{(n+1)\binom{2(n+1)}{n+1}} - \frac{2}{(n+1)^2\binom{2(n+1)}{n+1}}\\ (2.2.3) \longrightarrow &= 4\cdot\frac{1}{4^{n+1}}\mathbf{M}\left[\frac{1}{\sqrt{1-x}}\right](n) - 2\cdot\frac{1}{4^{n+1}}\mathbf{M}\left[\frac{\mathrm{H}^*_{\mathrm{w}_3}(x)}{x}\right](n+1)\\ (2.1.1) \longrightarrow &= \frac{1}{4^n}\mathbf{M}\left[\frac{1}{\sqrt{1-x}}\right](n) - \frac{1}{2}\cdot\frac{1}{4^n}\mathbf{M}\left[\mathrm{H}^*_{\mathrm{w}_3}(x)\right](n)\\ &= \frac{1}{4^n}\mathbf{M}\left[\frac{1}{\sqrt{1-x}} - \frac{\mathrm{H}^*_{\mathrm{w}_3}(x)}{2}\right](n)\end{aligned}$$

which concludes the proof. □

Using those individual Mellin inverses that we computed manually together with the Theorems from Section 2.2, we can build all the possible cases that we consider

for an individual summand. Those form a subset $\mathcal{S}_{\text{comp}}$ of the extended binomial sums that is comprised of elements of the form:

$$\mathcal{S}_{\text{comp}} = \left\{ \frac{c^i}{i^m}\binom{2i}{i}^b, \frac{p(i)}{q(i)}c^i\binom{2i}{i}, \frac{p'(i)}{q'(i)}\frac{c^i}{i\binom{2i}{i}}, \frac{p'(i)}{q'(i)}\frac{c^i}{(i+1)\binom{2i}{i}}, \frac{p'(i)}{q'(i)}\frac{c^i}{(2i+1)\binom{2i}{i}} \right\} \tag{2.3.9}$$

where

$$c \in \mathbb{K}, b \in \{-1, 0, 1\}, m \in \mathbb{N},$$
$$p, q, p', q' \in \mathbb{K}[X], \deg p \leq \deg q, \deg p' \leq \deg q', V(q), V(q') \subset \mathbb{R}_-.$$

For practical applications, i.e., for what is implemented in the `RICA` package, we require that the roots of q, q' are in $\mathbb{R}_-$ due to (2.2.7).

Summation step: Once we obtain the Mellin representation of the summand a_k, we sum over it using the (2.1.27) resummation formula to get rid of the sum.

$$\begin{aligned}
&\sum_{i_k=1}^{i_{k-1}} a_k(i_k) = \sum_{i_k=1}^{i_{k-1}} (c_k)^{i_k}\, \mathbf{M}\left[f_{a_k}(x)\right](n) \\
&(2.1.27) \to = (c_k)^{i_{k-1}}\, \mathbf{M}\left[\frac{x}{x-\frac{1}{c_k}} f_{a_k}(x)\right](i_{k-1}) - \mathbf{M}\left[\frac{x}{x-\frac{1}{c_k}} f_{a_k}(x)\right](0).
\end{aligned}$$

Remark 2.3.3. *We notice that by applying the resummation formula, we obtain one term that carries the i_{k-1} index dependence and another one that is a constant that can at best be computed analytically and represented in terms of known constants, for instance, $\log n, \zeta_n, \pi, \ldots, n \in \mathbb{N}_{\geq 2}$, and at worst still has a symbolic representation that allows numerical computations up to any number of digits.*

Convolution step: We compute the convolution between the previously obtained Mellin integrals and the new summand.

The initialisation step has been done. Now suppose that at the step $1 \leq j < k$, we managed to represent the following sub-sum as a Mellin integral:

$$\begin{aligned}
\mathcal{BS}_{(a_{j+1},\ldots,a_k)}(i_j) &= \sum_{i_{j+1}=1}^{i_j} a_{j+1}(i_{j+1}) \cdots \sum_{i_k=1}^{i_{k-1}} a_k(i_k) \\
&= c_{0,j+1} + \sum_{p=1}^{m_j} c_{p,j+1}^{i_j} M[f_{p,j+1}(x)](i_j),\ c_{0,j+1}, c_{p,j+1} \in \mathbb{K}.
\end{aligned} \tag{2.3.10}$$

In particular the constant $c_{0,j+1}$ could be a linear combination of Mellin transforms evaluated at 0, obtained from the application of (2.1.27). We now need to represent the expression

$$a_j(i_j)\mathcal{BS}_{(a_{j+1},\ldots,a_k)}(i_j),$$

where once again we suppose that $a_j(i_j)$ is an element from the set $\mathcal{S}_{\text{comp}}$, i.e.,

$$a_j(i_j) = c_j^{i_j}\mathbf{M}\left[f_j(x)\right](i_j).$$

Because of that, the relation above can be rewritten as

$$a_j(i_j)\mathcal{BS}_{(a_{j+1},\ldots,a_k)}(i_j) = c_{0,j+1}c_j^{i_j}\mathbf{M}\left[f_j(x)\right](i_j) + \sum_{p=1}^{m_j} c_{p,j+1}^{i_j}c_j^{i_j}\boxed{\mathbf{M}\left[f_j(x)\right](i_j)\mathbf{M}\left[f_{p,j+1}(x)\right](i_j)}. \quad (2.3.11)$$

Therefore, we have to compute the boxed quantity, i.e., the Mellin convolutions, in order to continue.

Remark 2.3.4. *This step is the most delicate one and is the tradeoff for avoiding to have to deal with differential equations in the classical method. Indeed, we shifted the difficulty from searching solutions for holonomic differential equations to computations of convolution integrals.*

There exist different possibilties to compute those convolutions:

- **General convolution rules**: In [16], several theorems are proven that allow one to immediately convolve certain general classes of binomial sums. In the following we recalculate a series of results given in [16], which correspond to classes of sums for which general convolution formulas are proven[17]:

$$\binom{2n}{n}\mathrm{S}_{k_1,\ldots,k_m}(x_1,\ldots,x_m)(n)$$

Theorem 2.3.1. *Let* $a_0,\ldots,a_k < 0$ *and define* $f_{\mathrm{b_i}}(x) := \frac{1}{\sqrt{(x-a_i)(x-a_{i+1})}}$ *for* $i \in \{0,\ldots,k-1\}$ *and* $f_{\mathrm{b_k}}(x) := \frac{1}{\sqrt{(1-x)(x-a_k)}}$. *Then we have*

$$\int_x^1 dt \frac{\mathrm{H}^*_{\mathrm{a_1},\ldots,\mathrm{a_k}}(t)}{(t-a_0)\sqrt{t-x}} = \frac{\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(x)}{\sqrt{x-a_0}}$$

and

$$\binom{2n}{n}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{a_1},\ldots,\mathrm{a_k}}(x)}{x-a_0}\right](n) = \frac{4^n}{\pi}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(x)}{\sqrt{x\,(x-a_0)}}\right](n).$$

[17] The proofs are in the original article [16].

$$\frac{1}{(2n+1)\binom{2n}{n}}\sum_{i=1}^{n} x_0^i \binom{2i}{i} \mathrm{S}_{k_1,\ldots,k_m}(x_1,\ldots,x_m)(i)$$

Theorem 2.3.2. *Let $a_0,\ldots,a_k,c<0$ and define $f_{\mathrm{b_i}}(x) := \frac{1}{\sqrt{(x-a_i)(x-a_{i+1})}}$ for $i\in\{0,\ldots,k-1\}$, $f_{\mathrm{b_k}}(x) := \frac{1}{\sqrt{(1-x)(x-a_k)}}$, $f_{\mathrm{v}}(x) := \frac{1}{x\sqrt{x-c}}$, and $f_{\mathrm{w}}(x) := \frac{1}{(x-a_0)\sqrt{x-c}}$. Then we have*

$$\int_x^1 dt \frac{\sqrt{t-a_0} f_{\mathrm{b_0}}(t)\mathrm{H}^*_{\mathrm{b_1},\ldots,\mathrm{b_k}}(t)}{\sqrt{t-x}} = \pi \mathrm{H}^*_{\mathrm{a_1},\ldots,\mathrm{a_k}}(x)$$
$$\int_x^1 dt \frac{\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(t)}{(t-c)\sqrt{(t-a_0)\,(t-x)}} = \frac{\pi}{\sqrt{x-c}}\int_x^1 dt \frac{\mathrm{H}^*_{\mathrm{a_1},\ldots,\mathrm{a_k}}(t)}{(t-a_0)\,\sqrt{t-c}}$$

and

$$\frac{1}{(2n+1)\binom{2n}{n}}\mathrm{M}\left[\frac{x\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(x)}{(x-c)\sqrt{x\,(x-a_0)}}\right](n) = \frac{\pi}{2\cdot 4^n}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{w},\mathrm{a_1},\ldots,\mathrm{a_k}}(x)}{\sqrt{x-c}}\right](n)$$
$$\frac{1}{(n+1)\binom{2n}{n}}\mathrm{M}\left[\frac{x\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(x)}{(x-c)\sqrt{x\,(x-a_0)}}\right](n) = \frac{\pi}{4^n}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{w},\mathrm{a_1},\ldots,\mathrm{a_k}}(x)}{\sqrt{x-c}} - \frac{1}{2}\mathrm{H}^*_{\mathrm{v},\mathrm{w},\mathrm{a_1},\ldots,\mathrm{a_k}}(x)\right](n)$$
$$\frac{1}{n\binom{2n}{n}}\mathrm{M}\left[\frac{x\mathrm{H}^*_{\mathrm{b_0},\ldots,\mathrm{b_k}}(x)}{(x-c)\sqrt{x\,(x-a_0)}}\right](n) = \frac{\pi}{4^n}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{a_0},\ldots,\mathrm{a_k}}(x)}{x} + c\frac{\mathrm{H}^*_{\mathrm{w},\mathrm{a_1},\ldots,\mathrm{a_k}}(x)}{x\sqrt{x-c}}\right](n).$$

$$\frac{1}{n\binom{2n}{n}}\mathrm{S}_{k_1,\ldots,k_m}(x_1,\ldots,x_m)(n)$$

Theorem 2.3.3. *Let $a_0,\ldots,a_k<0$ and with $e_0,e_1\in\{0,1\}$ define $f_{b_i^{e_0},e_1}(x) := \frac{1}{x\sqrt{(x-a_i)^{e_0}(x-a_{i+1})^{e_1}}}$ for $i\in\{0,\ldots,k-1\}$ and $f_{\mathrm{b_k}^{e_0}}(x) := \frac{1}{x\sqrt{(1-x)(x-a_k)^{e_0}}}$. Then we have*

$$\int_x^1 dt \frac{\sqrt{t}\mathrm{H}^*_{\mathrm{a_1},\ldots,\mathrm{a_k}}(t)}{(t-a_0)\,\sqrt{t-x}} = \sum_{e_0,\ldots,e_k=0}^{1} \frac{\prod_{i=0}^k a_i^{e_i}}{\sqrt{(x-a_0)^{e_0}}}\mathrm{H}^*_{\mathrm{b_0}^{e_0,e_1},b_1^{e_1,e_2},\ldots,b_k^{e_k}}(x)$$

and

$$\frac{1}{n\binom{2n}{n}}\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathbf{a}_1,\ldots,a_k}(x)}{x-a_0}\right](n)=\frac{1}{4^n}\mathrm{M}\left[\frac{1}{x}\sum_{e_0,\ldots,e_k=0}^{1}\frac{\prod_{i=0}^{k}a_i^{e_i}}{\sqrt{(x-a_0)^{e_0}}}\mathrm{H}^*_{\mathrm{b}_0^{e_0,e_1},b_1^{e_1,e_2},\ldots,b_k^{e_k}}(x)\right](n).$$

$$\binom{2n}{n}\sum_{i=1}^{n}\frac{x_0^i}{i\binom{2i}{i}}\mathrm{S}_{k_1,\ldots,k_m}(x_1,\ldots,x_m)(i)$$

Theorem 2.3.4. *Let $a_0,\ldots,a_k,c<0$ and with $e_0,e_1\in\{0,1\}$ define $f_{\mathrm{b}_\mathrm{i}^{e_0,e_1}}(x):=\frac{1}{x\sqrt{(x-a_i)^{e_0}(x-a_{i+1})^{e_1}}}$ for $i\in\{0,\ldots,k-1\}$ as well as $f_{b_k^{e_0}}(x):=\frac{1}{x\sqrt{(1-x)(x-a_k)^{e_0}}}$ and $f_{\mathrm{w}_0 e_0}(x):=\frac{1}{(x-a_0)^{e_0}\sqrt{x(x-c)}}$. Then with $e_1,\ldots,e_k\in\{0,1\}$ we have*

$$\binom{2n}{n}\mathrm{M}\left[\sum_{e_0,\ldots,e_k=0}^{1}\frac{\left(\prod_{i=0}^{k}a_i^{e_i}\right)\mathrm{H}^*_{\mathrm{b}_0^{e_0,e_1},b_1^{e_1,e_2},\ldots,b_k^{e_k}}(x)}{(x-c)\sqrt{(x-a_0)^{e_0}}}\right](n)=4^n\mathrm{M}\left[\frac{\mathrm{H}^*_{\mathrm{w}_0^0,\mathrm{a}_1,\ldots,\mathrm{a}_\mathrm{k}}(x)}{\sqrt{x(x-c)}}+a_0\frac{\mathrm{H}^*_{\mathrm{w}_0^1,\mathrm{a}_1,\ldots,\mathrm{a}_\mathrm{k}}(x)}{\sqrt{x(x-c)}}\right](n).$$

Remark 2.3.5. *The theorems require that the parameters satisfy $a_0,\ldots,a_k,c<0$, but in most if not all the application cases we consider, this hypothesis is not always fulfilled. Fortunately this is a weak hypothesis and in the spirit of Example 1 p.22 and following pages of [16], by allowing analytic continuation of the integrands of the theorems, we can consider positive values for the a_i and c parameters.*

Remark 2.3.6. *We also consider cases where the S-sums do not reduce to harmonic sums $\mathrm{S}_{\mathbf{1}_p}$. How this can be tackled will be explained below.*

These theorems can therefore be considered as a set of "rules", which we will call in the following also "theorem-rules". Using Mathematica's pattern matching, if the left-hand side can be identified in a particular expression, it can be immediately replaced. We have translated them[18] to Mathematica syntax and implemented in our `RICA` package.

- **Plus prescription cases**: When we have p^{th}-order harmonic sums $\mathrm{S}_{\mathbf{1}_p}(n)$, we use results from Subsection 2.2.2, in particular Property 2.2.3 and Theorem 2.2.3, which can be applied in the first case at least for $p\leq 10$ and in the second case for $p=0,1,2$.

[18] Technically speaking, these are not directly translatable to simple rules because the formulas are rather general and have many parameters. So in most cases those will be simple Mathematica functions acting like rules.

- **Case by case convolutions**: Given that summation gives rise to constant parts[19], we have to deal with Mellin representations using GRPLs of different depth. It gives rise to additional convolutions which can simplify as follows:
  - No binomial coefficient, i.e., $a_j(i_j) = \frac{p_j(i_j)}{q_j(i_j)} \frac{c_j^{i_j}}{i_j^{k_j}}$: in this case we can simply use Theorems (2.2.1), (2.2.2).
  - Simple cases with letters other than the 32 already defined, e.g., $\frac{\sqrt{x}}{\sqrt{1-x}(x+b)}$ $(b > 0)$: we encoutered some letters that are rather simple but were not part of the theorem-rule settings. In this case, we try to explicitely compute the Mellin convolution using `ExtendedMellinConvolve`[20], our own function implemented in `RICA`, or adapting the function used by Mathematica, `MellinConvolve`, to our usage[21].We identified in particular two classes of convolutions this way, which were added as a set of rules to `RICA`:

$$\left(\frac{1}{x\sqrt{1-x}}\right) \star \left(\frac{1}{x+b}\sqrt{\frac{x}{1-x}}\right) = \begin{cases} \frac{\pi\left(1-\frac{b}{\sqrt{(b+1)(b+x)}}\right)}{x} & \text{if } b \in (-\infty,-1] \cup \mathbb{R}_+ \backslash \{1\} \\ \frac{\pi\left(1-\frac{b}{\sqrt{|(b+1)(b+x)|}}\right)}{x} & \text{if } b \in (-1;0) \end{cases} \tag{2.3.12}$$

$$\left(\frac{1}{\sqrt{x(1-x)}}\right) \star \left(\frac{1}{(x+b)\sqrt{1-x}}\right) = \frac{\pi}{\sqrt{(b+1)x(b+x)}} \text{ if } b \in (-\infty;-1) \cup \mathbb{R}_+^\star \backslash \{1\}. \tag{2.3.13}$$

- **Generalized convolution polylogarithms**: In the case all the methods above fail, we can define a new set of functions and keep on doing the computations. This case is discussed in depth in the next Section 2.3.4.

Induction step: Repeat the summation and convolution step until we reach the most outward sum.

As we can see, the procedure is rather straightfoward, and basically consists in three ingredients: computing Mellin inverses of the summands, summing over them and then computing convolutions to be able to write the next summand as a sum of Mellin transforms — instead of a product — in order to be able to sum over them

[19] See in Remark 2.3.3.

[20] See Appendix A.2.1.

[21] Mathematica's function uses a different definition for the Mellin transform, more precisely for a function $f = f(x)$ and a parameter s, the Mellin transform of f is

$$\mathbf{M}\left[f(x)\right](s) = \int_0^{+\infty} \mathrm{d}x\, x^{s-1} f(x).$$

again. The summation formula (2.1.27) does not require particular hypotheses. Furthermore, we proved the different formulas for the inversion of basic blocks obtained from the specification of the binomial sums, and we rely either on theorems or explicit computations for the convolution steps. Moreover, by construction, the procedure finishes in at most p iteration steps for a binomial sum of depth p, so the method is actually an algorithm. The downside is that we did not manage to represent the *full* class of (extended) binomial sums due to the complexity of the convolution step — what we manage to represent is the class of sums of physical interest, such as those computed in the article [16], and we extended those to some extra classes that were not described there, for example by extending to coefficients with rational functions, or to the plus-prescription cases described in Subsection 2.2.2 and not mentioned precisely in [16]. Therefore, to summarise, our algorithm will apply to the following subset of the extended binomial sums from Definition 2.1.4:

Specification: The algorithm above applies when the input is a binomial sum of the form

$$\overline{\mathcal{BS}}_{(\alpha_1,\ldots,\alpha_j,\ldots,\alpha_k)}(n) \tag{2.3.14}$$

where, using the conventions $\alpha_j = \alpha_j(i_j; b_j, c_j, p_j, q_j)$ of Definition 2.1.4 the following holds. There exists a $\lambda \in \{1, \ldots, k\}$ such that

$$(b_\lambda, b_{\lambda+1}) \in \{(0,1), (-1,1), (0,-1), (1,-1)\} \tag{2.3.15}$$

and

$$\forall i \in \{\lambda, \lambda+1\}, b_i = -1 \Rightarrow x | q_i(x). \tag{2.3.16}$$

Furthermore we require that for all the indices $i \in \{1, \ldots, k\}, i \neq \lambda, \lambda+1$ we have

$$b_i = 0,\ \deg p_i \leq \deg q_i,\ V(q_i) \in \mathbb{R}_-. \tag{2.3.17}$$

Example 2.3.1. *Let us compute explicitely using this algorithm the Mellin inverse of the following sum:*

$$\mathcal{BS}_{((1,1,0),(0,1,2))}(n) = \sum_{k=1}^{n} \binom{2i}{i} S_2(i) = \sum_{k=1}^{n} \binom{2i}{i} \sum_{j=1}^{i} \frac{1}{j^2}.$$

- *First we compute the Mellin representation of* $\frac{1}{i^2}$ *by applying Theorem 2.2.1. We get:*

$$\frac{1}{i^2} = \mathbf{M}\left[\frac{\mathrm{H}_0^*(x)}{x}\right](i).$$

- *Using the summation formula (2.1.27), we can then obtain:*

$$\begin{aligned} S_2(i) &= \sum_{k=1}^{i} \mathbf{M}\left[\frac{\mathrm{H}_0^*(x)}{x}\right](k) \\ &= \mathbf{M}\left[\frac{x}{x-1}\frac{\mathrm{H}_0^*(x)}{x}\right](i) - \mathbf{M}\left[\frac{x}{x-1}\frac{\mathrm{H}_0^*(x)}{x}\right](0) \\ &= \int_0^1 \mathrm{d}x\, x^i \frac{\mathrm{H}_0^*(x)}{x-1} - \int_0^1 \mathrm{d}x\, \frac{\mathrm{H}_0^*(x)}{x-1} \\ &= \int_0^1 \mathrm{d}x\, x^i \frac{\mathrm{H}_0^*(x)}{x-1} + \zeta_2. \end{aligned}$$

- *Now that the innermost sum has an integral representation, we shift to the next and last level. We have:*

$$\sum_{i=1}^{k} \binom{2i}{i} S_2(i) = \sum_{i=1}^{n} \binom{2i}{i} \mathbf{M}\left[\frac{\mathrm{H}_0^*(x)}{x-1}\right](i) + \zeta_2 \sum_{i=1}^{n} \binom{2i}{i}. \tag{2.3.18}$$

- *We see that the first part of (2.3.18) is of the form:*

$$\binom{2i}{i} \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathrm{a}_1,\ldots,\mathrm{a}_\mathrm{k}}(\mathrm{x})}{x-a_0}\right](i)$$

which is related to the Mellin representation of cases of the form

$$\binom{2i}{i} S_2(i) \quad \to \quad \binom{2i}{i} S_{k_1,\ldots,k_m}(x_1,\ldots,x_m)(i).$$

The Theorem-rule 2.3.1 yields:

$$\binom{2i}{i} \mathbf{M}\left[\frac{\mathrm{H}_0^*(x)}{x-1}\right](i) = \frac{4^i}{\pi} \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathrm{b},\mathrm{w}_1}(x)}{\sqrt{x(x-1)}}\right](i) \text{ with } b(x) = \frac{1}{\sqrt{x(x-1)}}$$

Using analytic continuation and shuffle relations of H^* *[8, 9] we get:*

$$\binom{2i}{i} \mathbf{M}\left[\frac{\mathrm{H}_0^*(x)}{x-1}\right](i) = -\mathbf{M}\left[\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2\sqrt{x(1-x)}}\right](i).$$

- *For the second part, we use the base block representation of binomial coefficients:*

$$\binom{2i}{i} = \frac{4^i}{\pi} \mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](i).$$

- *Finally, we make use of the summation formula (2.1.27) for both parts:*

$$\frac{1}{\pi}\sum_{i=1}^{n} 4^i \mathbf{M}\left[\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2\sqrt{x(1-x)}}\right](i) = \frac{1}{\pi}\int_0^1 \mathrm{d}x\, \frac{(4x)^n-1}{x-\frac{1}{4}}\sqrt{\frac{x}{1-x}}\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2}$$

$$\frac{\pi}{6}\sum_{i=1}^{k} 4^i \mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](i) = \frac{\pi}{6}\int_0^1 \mathrm{d}x\, \frac{(4x)^n-1}{x-\frac{1}{4}}\sqrt{\frac{x}{1-x}}.$$

- *Resumming everything, we get:*

$$\sum_{i=1}^{n}\binom{2i}{i}S_2(i) = -\frac{1}{\pi}\int_0^1 \mathrm{d}x\, \frac{(4x)^n-1}{x-\frac{1}{4}}\sqrt{\frac{x}{1-x}}\left(\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2}-\zeta_2\right). \tag{2.3.19}$$

Later we will use this representation to compute its asymptotic expansion in Example 2.5.4.

2.3.3. An example with RICA

We show how we can deal with the examples described in Section 2.3.2 in the `RICA` package. When we start a computing session, first we preload `Sigma` [1, 94] and `HarmonicSums` [9, 10, 12, 72–74]:

```
In[1]:= << Sigma.m;
```
```
Sigma - A summation package by Carsten Schneider – © RISC
```
```
In[2]:= << HarmonicSums.m;
```
```
HarmonicSums by Jakob Ablinger – © RISC
```

and then our package[22]:

```
In[3]:= << RICA.m;
```
```
Rule Induced Convolutions for Asymptotics (RICA) package by Nikolai Fadeev © RISC-JKU
```

We define the sum that we want to study using `HarmonicSums`' `GS`, which basically takes as first argument the list of summands from left to right expressed in the `VarGL` variable[23] and the external bound n of the sum:

```
In[4]:= sum1 = GS[{Binomial[2 VarGL, VarGL], 1/VarGL^2}, n];
```

We can use the `RICA` function `GSToHSS` to obtain the explicit form of the `GS` object and check its explicit form:

[22] This will hold true for all further RICA examples, i.e., we will suppose these preloading steps have already been done.

[23] This variable acts in the same way as the λ in the GRPLs introduced in Definition 2.1.13.

In[5]:= **GSToHSS[sum1]**

Out[5]= $$\sum_{\tau_1=1}^{\mathtt{n}} \binom{2\tau_1}{\tau_1} \left(\sum_{\tau_2=1}^{\tau_1} \frac{1}{\tau_2^2} \right)$$

We can now do the steps above to compute the Mellin inverse using a single function of `RICA` called `SumToMellin`:

In[6]:= **mel1 = SumToMellin[sum1, C, x]**

Out[6]= $$\left\{ -\frac{2\text{Mellin}\left[4^{\mathtt{n}}\mathtt{x}^{\mathtt{n}} - 1, \frac{\sqrt{\mathtt{x}}\left(\text{Hwb}[\{\text{fw1}\},\mathtt{x}]^2 - 2\text{z2}\right)}{\sqrt{1-\mathtt{x}}(4\mathtt{x}-1)}\right]}{\pi}, \{\} \right\}$$

The function takes three compulsory arguments, and several optional ones in the form of rules[24]:

- The first argument is the sum, or more generally the linear combination of sums, for which we want to compute the Mellin inverse.
- The second argument is the letter over which we should index the constants appearing in the expression — we will explain further what it means below.
- The last argument is the x variable over which we integrate in the Mellin representation.

Except if the function fails, in which case it should return back the original sum, the output is in the form of a tuple.

The **first element** is the Mellin inverse of the sum, or more precisely, a linear combination of (extended) Mellin transforms

$$\sum_{i=1}^{k} \alpha_i \text{Mellin}\left[(a_i x)^{\,n} - 1, f_i(x)\right] \quad \text{with } \alpha_i, a_i \in \mathbb{K}^* \tag{2.3.20}$$

where the notation for the Mellin transforms should be understood as

$$\text{Mellin}[(a_i x)^n - 1, f_i(x)] = \int_0^1 \mathrm{d}x \, [(a_i x)^n - 1] \, f_i(x).$$

The $f_i(x)$ can be given as GRPLs using the following notations:

- With a `GLb` from Ablinger's `HarmonicSums` package — the first argument is a list of explicit letters in the `VarGL` variable and the second is the outward lower bound variable of the integral, x. For example

$$\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}, \frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}}\right\}, x\right] = \int_x^1 \mathrm{d}t_1 \frac{1}{t_1} \int_{t_1}^1 \mathrm{d}t_2 \frac{1}{\sqrt{t_2(1-t_2)}}.$$

[24] Those are described at length in Appendix A.2.2.

- An `Hwb` function that we have encoded in `RICA` which does a similar role to `GLb` except that the list of letters is implicit[25] — in particular, if we take the letter $f_0(x) = \frac{1}{x}$, in `GLb` we will need to write $\frac{1}{\mathrm{VarGL}}$ whereas in `Hwb f0` will be enough. For example

$$\mathrm{Hwb}\left[\{\mathrm{f0}, \mathrm{fw1}\}, x\right] = \mathrm{GLb}\left[\left\{\frac{1}{\mathrm{VarGL}}, \frac{1}{\sqrt{\mathrm{VarGL}(1-\mathrm{VarGL})}}\right\}, x\right]$$

where we used the definition of f_{w_1} in (2.1.47). In order to be able to go from one convention to the other, we have implemented in `RICA` two functions, `ConvertHwb` (`Hwb` → `GLb`) and `GLbToHwb` (`GLb` → `Hwb`) — for more details about those, see Appendix A.

The **second element** of the tuple is a list of constants specifications in the form of

$$\{C_1 \to \mathrm{expr}_1, \cdots, C_p \to \mathrm{expr}_p\} \tag{2.3.21}$$

where $p \in \mathbb{N}^\star$ the C_i is the generic name for the constant given in the input, and expr_i are expressions involving specific points of GRPL/GPL/HPL functions (at 0 or 1) or sometimes usual constants ($\zeta_p, \log p$, etc.) that define the constant C_i. This means that we should replace all occurences of C_i by the corresponding expr_i in the Mellin transform. In the example above we did not obtain such constants: the expression can be expressed without them. To see this second feature, we look at another sum:

```
In[7]:= sum2 = GS[{(1/3)^VarGL/(VarGL+2), 2^VarGL/(VarGL+1), 1/VarGL^2}, n];

In[8]:= GSToHSS[sum2];

Out[8]= Sum_{τ1=1}^{n} (3^-τ1 (Sum_{τ2=1}^{τ1} 2^τ2 (Sum_{τ3=1}^{τ2} 1/τ3^2))/(τ2+1))/(τ1+2)
```

where its Mellin inverse is:

```
In[9]:= mel2 = SumToMellin[sum2, C, x]

Out[9]= {2Mellin[{(2/3)^n x^n - 1, x^2 GLb[{1/((VarGL-1/2)VarGL), 1/((VarGL-1)VarGL), f0[VarGL]}, x]/(2x-3)}, n, x]}
```

[25] For more details, see Appendix A.

$$+\frac{\mathtt{x}^2\mathtt{z2GLb}\left[\left\{\frac{1}{(\mathrm{VarGL}-\frac{1}{2})\mathrm{VarGL}}\right\},\mathtt{x}\right]}{2\mathtt{x}-3}\Bigg] + (\mathtt{C}_1(-\mathtt{z2})-\mathtt{C}_2)\,\mathrm{Mellin}\left[3^{-\mathtt{n}}\mathtt{x}^{\mathtt{n}}-1,\frac{\mathtt{x}^2}{\mathtt{x}-3}\right],$$
$$\left\{\mathtt{C}_1\to \mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}}\right\},0\right],\right.$$
$$\left.\mathtt{C}_2\to \mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}},\frac{1}{(\mathrm{VarGL}-1)\mathrm{VarGL}},\mathrm{f0[VarGL]}\right\},0\right]\right\}\Bigg\}$$

where we have

$$\mathrm{GLb}\left[\left\{\frac{1}{\left(\mathrm{VarGL}-\frac{1}{2}\right)\mathrm{VarGL}},\frac{1}{(\mathrm{VarGL}-1)\mathrm{VarGL}},\mathrm{f0[VarGL]}\right\},x\right]$$
$$=\int_x^1 \mathrm{d}\tau_1\int_{\tau_1}^1 \mathrm{d}\tau_2\int_{\tau_2}^1 \mathrm{d}\tau_3\,\frac{1}{\left(-\frac{1}{2}+\tau_1\right)\tau_1\left(-1+\tau_2\right)\tau_2\tau_3} \tag{2.3.22}$$

$$\mathrm{GLb}\left[\left\{\frac{1}{\left(\mathrm{VarGL}-\frac{1}{2}\right)\mathrm{VarGL}}\right\},x\right]=\int_x^1 \mathrm{d}\tau_1\,\frac{1}{\left(\tau_1-\frac{1}{2}\right)\tau_1}. \tag{2.3.23}$$

We see here that we have two constants specified by the second term of the output tuple:

$$\left\{C_1\to \mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}}\right\},0\right],\right.$$
$$\left.C_2\to \mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}},\frac{1}{(\mathrm{VarGL}-1)\mathrm{VarGL}},\mathrm{f0[VarGL]}\right\},0\right]\right\},$$

which means that every occurence of C_1 and C_2 in the Mellin inverse should be replaced by those values. In particular, here we have:

$$C_1=\mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}}\right\},0\right]=\oint_0^1 \mathrm{d}\tau_1\,\frac{\tau_1}{\tau_1-\frac{1}{2}}=1 \tag{2.3.24}$$

$$C_2=\mathrm{GLb}\left[\left\{\frac{\mathrm{VarGL}}{\mathrm{VarGL}-\frac{1}{2}},\frac{1}{(\mathrm{VarGL}-1)\mathrm{VarGL}},\mathrm{f0[VarGL]}\right\},0\right]$$
$$=\int_0^1 \mathrm{d}\tau_1\int_{\tau_1}^1 \mathrm{d}\tau_2\int_{\tau_2}^1 \mathrm{d}\tau_3\,\frac{\tau_1}{\left(-\frac{1}{2}+\tau_1\right)\left(-1+\tau_2\right)\tau_2\tau_3}=\frac{\pi^2}{6}-\frac{7\zeta(3)}{4}. \tag{2.3.25}$$

We also see that when `RICA` manages to identify a known letter among the 32 defined in Remark 2.1.11, it will explicitely make them appear even in `GLb` expressions

as unactivated functions of `VarGL` for clarity sake[26]. In order to use those constants one can execute Mathematica's `Activate` function at any time on those constants, which will then activate and therefore evaluate the unactivated expressions.

2.3.4. Generalized convolution polylogarithms

When one performs the Mellin convolution step in the previous section, it happens that we get some Mellin tranforms that cannot be convolved according to any of our available steps. In this case, the algorithm may stop with the message `$Failed`. But we decided to take a different approach that will be essential in Section 2.5.4.

The idea is to tackle situations where we cannot compute a Mellin transform using the methods described in the Subsection 2.3.2. Then we can keep the convolution integral — which is defined as a function on the interval $(0;1)$ — and keep on doing the computations. However, this raises the following issue: if the abovementioned computations were intermediate, we might need to apply the resummation formula (2.1.27) and then convolve again when we reach the next sum. The idea is therefore to define a proper notation for these new objects and therefore define them as their own proper class of functions.

Definition 2.3.3 (Generalized convolution polylogarithms)**.** *Let $x \in (0;1)$, $\mathcal{A}$ an alphabet, $f : x \mapsto f(x)$ a function defined at least on $(0;1)$ that can be or not be a letter of $\mathcal{A}$, $\vec{w}$ a word of $\mathcal{A}$. We define a* ***w-letter*** *as a tuple $\underline{\mathrm{w}} = [f;\vec{w}]$, and a* ***generalized word*** *$\underline{\mathrm{V}}$ as any combinations of letters f_i and w-letters $\underline{\mathrm{w}_i}$ of the form $(f_i, \underline{\mathrm{x}_j})$ and $(\underline{\mathrm{w}_i}, \underline{\mathrm{w}_j})$, and iterated combinations with the obtained generalized words $\underline{W_i}$ of the form $(f_i, \underline{W_j})$, $(\underline{w_i}, \underline{W_j})$, $(\underline{W_i}, \underline{w_j})$ and $(\underline{W_i}, \underline{W_j})$. We say that $\underline{V}$ is a* ***fundamental generalized word*** *if it is of the form $\underline{V} = (\underline{\mathrm{w}_i}, \underline{\mathrm{w}_j})$ where $\underline{\mathrm{w}_i}, \underline{\mathrm{w}_j}$ are two w-letters. We define the* ***generalized convolution polylogarithms*** *(GCPL), denoted with $\mathcal{H}^*$, in the following way:*

- *The base case for a single w-letter:*

$$\mathcal{H}^*_{[1;\emptyset]}(x) := \mathrm{H}^*_{\emptyset}(x) = 1 \tag{2.3.26}$$

$$\mathcal{H}^*_{[\mathrm{f};\overrightarrow{w}]}(x) := f(x)\,\mathrm{H}^*_{\overrightarrow{w}}(x). \tag{2.3.27}$$

- *If $\underline{\mathrm{V}}, \underline{\mathrm{W}}$ are two generalized words, we define recursively:*

$$\mathcal{H}^*_{(\mathrm{f},\underline{\mathrm{V}})}(x) := f(x)\mathcal{H}^*_{\underline{\mathrm{V}}}(x) \tag{2.3.28}$$

[26] Some functions were not part of the 32 letters but appeared several times in certain examples. So we added them to the list of functions that `RICA` can identify and make appear in `GLb`, the `HarmonicSums` [9] representation of GRPLs, to make constants more legible.

$$\mathcal{H}^*_{(\underline{V},\underline{W})}(x) := \mathcal{H}^*_{\underline{V}}(x) \star \mathcal{H}^*_{\underline{W}}(x) \tag{2.3.29}$$

where as usual $\star$ denotes the Mellin convolution[27].

Example 2.3.2. *Here are some examples of generalized words:*

$$[f_1; \emptyset]$$

$$[f_0; (f_{w_1}, f_{w_9}, f_2)]$$

$$([1; (f_1, f_2)], [f_0; f_0])$$

$$((f_{w_3}, (f_1, [f_{w_{28}}; (f_{w_1}, f_{w_9}, f_2)])), ([1; (f_1, f_2)], [f_0; f_0]))$$

Example 2.3.3. *Simple examples of GCPLs are:*

$$\mathcal{H}^*_{[\mathrm{f}_0;\emptyset]}(x) = f_0(x) = \frac{1}{x}$$

$$\mathcal{H}^*_{[\mathrm{f}_1;(\mathrm{f}_{\mathrm{w}_1},\mathrm{f}_{\mathrm{w}_3})]}(x) = f_1(x)\mathrm{H}^*_{\mathrm{f}_{\mathrm{w}_1},\mathrm{f}_{\mathrm{w}_3}}(x) = \frac{1}{1-x}\int_x^1 \mathrm{d}t_1 \frac{1}{\sqrt{t_1(1-t_1)}}\int_{t_1}^1 \mathrm{d}t_2 \frac{1}{t_2\sqrt{1-t_2}}$$

Example 2.3.4. *The GCPL allows us to rewrite convolutions of GRPL/GPL/HPL in a very compact way that is stable under convolutions. Let f, g be letters, and $\vec{v}, \vec{w}$ words. Then:*

$$[f(x)\,\mathrm{H}^*_{\vec{v}}(x)] * [g(x)\,\mathrm{H}^*_{\vec{w}}(x)] = \mathcal{H}^*_{[\mathrm{f};\vec{v}]}(x) \star \mathcal{H}^*_{[\mathrm{g};\vec{w}]}(x) = \mathcal{H}^*_{([\mathrm{f};\vec{v}],[\mathrm{g};\vec{w}])}(x).$$

If we take for instance $f(x) = \frac{1}{x}$, $g(x) = \frac{1}{2+x}$, $\vec{v} = \frac{1}{1+x}$ and $\vec{w} = \left(\frac{1}{x\sqrt{1-x}}, \frac{1}{\sqrt{x(1-x)}}\right)$ we get

$$\mathcal{H}^*_{\left(\left[\frac{1}{\lambda};\frac{1}{1+\lambda}\right],\left[\frac{1}{2+\lambda};\left(\frac{1}{\lambda\sqrt{1-\lambda}},\frac{1}{\sqrt{\lambda(1-\lambda)}}\right)\right]\right)}(x) = \frac{1}{x}\int_x^1 \mathrm{d}x_1 \frac{\mathrm{H}^*_{\frac{1}{\lambda+1}}\left(\frac{x}{x_1}\right)\mathrm{H}^*_{\frac{1}{\lambda\sqrt{1-\lambda}},\frac{1}{\sqrt{\lambda(1-\lambda)}}}(x_1)}{x_1+2}. \tag{2.3.30}$$

Example 2.3.5. *Some GCPLs can be computed explicitely using well-known classical functions, for example the one below which contains a double convolution:*

$$\begin{aligned}\mathcal{H}^*_{([\lambda;\emptyset],([\frac{1}{\lambda};\emptyset],[\frac{1}{1+\lambda};\emptyset]))}(x) &= x\int_x^1 \mathrm{d}y \frac{1}{y^3}\int_y^1 \mathrm{d}z \frac{1}{z+1} \\ &= \frac{-x^2\log(2x) + (x^2-1)\log(x+1) + (x-1)x + \log(2)}{2x}.\end{aligned} \tag{2.3.31}$$

[27] It is implicitely understood that we will use convolution in the +-prescription if one of the GCPL requires it.

We also define a particular subclass of GCPLs.

Definition 2.3.4 (Pure generalized convolution polylogarithm)**.** *Let $x \in (0;1)$, $\mathcal{A}$ an alphabet, $f : x \mapsto f(x)$ a function defined at least on $(0;1)$ that can be or not a letter of $\mathcal{A}$ and $\vec{w}$ a word of $\mathcal{A}$. We say that a $\underline{w_i}$ is a **pure w-letter** if it is of the form $[f_i;\emptyset]$, a generalized word $\underline{V}$ is a **pure generalized word** if it contains only pure w-letters, i.e., if it is built only from convolutions of such pure w-letters $\underline{w_i}$ and letters f_i. Consequently, we call $\mathcal{H}^*_{\underline{V}}(x)$ a **pure generalized convolution polylogarithm**.*

Example 2.3.6. *The GCPL (2.3.31) is a pure GCPL while (2.3.30) is not.*

One can deduce some properties of those GCPLs based on their definition.

Property 2.3.1. *Let f, g be two functions defined on $(0;1)$, $\underline{U}, \underline{V}, \underline{W}$ generalized words. We have the following identities:*

$$\mathcal{H}^*_{(\mathrm{f},(\mathrm{g},\underline{V}))}(x) = \mathcal{H}^*_{(\mathrm{fg},\underline{V})}(x) \tag{2.3.32}$$

$$\mathcal{H}^*_{(\underline{U},(\underline{V},\underline{W}))}(x) = \mathcal{H}^*_{((\underline{U},\underline{V}),\underline{W})}(x) \tag{2.3.33}$$

$$\mathcal{H}^*_{(\lambda,([\frac{1}{\lambda};\emptyset],\underline{V}))}(x) = \int_x^1 \mathrm{d}y\, \mathcal{H}^*_{\underline{V}}(y) \tag{2.3.34}$$

where the last identity translates (iterated) integration in the GCPL language.

Proof. We obtain the proof for (2.3.32) simply by applying (2.3.28):

$$\mathcal{H}^*_{(\mathrm{f},(\mathrm{g},\underline{V}))}(x) \underset{(2.3.28)}{=} f(x)\mathcal{H}^*_{(\mathrm{g},\underline{V})}(x) \underset{(2.3.28)}{=} f(x)g(x)\mathcal{H}^*_{\underline{V}}(x) \underset{(2.3.28)}{=} \mathcal{H}^*_{\{\mathrm{fg},\underline{V}\}}(x).$$

The identity (2.3.33) is proven using the associativity of the Mellin convolution proven in Property 2.1.5:

$$\begin{aligned}
\mathcal{H}^*_{(\underline{U},(\underline{V},\underline{W}))}(x) &\underset{(2.3.29)}{=} \mathcal{H}^*_{\underline{U}}(x) \star \mathcal{H}^*_{(\underline{V},\underline{W})}(x) \\
&\underset{(2.3.29)}{=} \mathcal{H}^*_{\underline{U}}(x) \star \left[\mathcal{H}^*_{\underline{V}}(x) \star \mathcal{H}^*_{\underline{W}}(x)\right] \\
\text{Property 2.1.5} \longrightarrow &= \left[\mathcal{H}^*_{\underline{U}}(x) \star \mathcal{H}^*_{\underline{V}}(x)\right] \star \mathcal{H}^*_{\underline{W}}(x) \\
&\underset{2\times(2.3.29)}{=} \mathcal{H}^*_{((\underline{U},\underline{V}),\underline{W})}(x).
\end{aligned}$$

Finally, the last property is proved as follows:

$$\mathcal{H}^*_{(\lambda,([\frac{1}{\lambda};\emptyset],\underline{V}))}(x) \underset{(2.3.28)}{=} x\mathcal{H}^*_{([\frac{1}{\lambda};\emptyset],\underline{V})}(x)$$

$$\underset{(2.3.29)}{=} \not{x} \int_x^1 \mathrm{d}y \, \frac{1}{\not{y}} \frac{1}{\frac{\not{x}}{\not{y}}} \mathcal{H}^*_{\underline{V}}(y)$$
$$= \int_x^1 \mathrm{d}y \, \mathcal{H}^*_{\underline{V}}(y)$$

which concludes all the proofs. □

Remark 2.3.7. *Due to (2.3.34), the GCPL with the 4^{th} generalized word of Example 2.3.2 will simplify to:*

$$\mathcal{H}^*_{\left(\left(\mathrm{f}_{\mathrm{w}_3},\left(\mathrm{f}_1,\left[\mathrm{f}_{\mathrm{w}_{28}};\left(\mathrm{f}_{\mathrm{w}_1},\mathrm{f}_{\mathrm{w}_9},\mathrm{f}_2\right)\right]\right)\right),\left([1;(\mathrm{f}_1,\mathrm{f}_2)],[\mathrm{f}_0;\mathrm{f}_0]\right)\right)} = \mathcal{H}^*_{\left(\left[\mathrm{f}_1\mathrm{f}_{\mathrm{w}_3}\mathrm{f}_{\mathrm{w}_{28}};\left(\mathrm{f}_{\mathrm{w}_1},\mathrm{f}_{\mathrm{w}_9},\mathrm{f}_2\right)\right],\left([1;(\mathrm{f}_1,\mathrm{f}_2)],[\mathrm{f}_0;\mathrm{f}_0]\right)\right)}.$$

The symbolic representation of the GRPLs using GCPLs is not unique. As it turns out in Property 2.3.3 below, one gets at least a representation in terms of pure GCPLs. Here we rely on the following property:

Property 2.3.2. *Let $\mathrm{H}^*_{\alpha_1,\ldots,\alpha_k}(x)$ be a GRPL with α_i being some letters. Then there exists a pure generalized word $\underline{V}$ such that*

$$\mathrm{H}^*_{\alpha_1,\ldots,\alpha_k}(x) = \mathcal{H}^*_{\underline{V}}(x).$$

Proof. Taking the identity (2.3.34) and replacing $\underline{V}$ there by the pure w-letter $\{\alpha_k;\emptyset\}$, we obtain:

$$\mathcal{H}^*_{\left(\lambda,\left(\left[\frac{1}{\lambda};\emptyset\right],[\alpha_k;\emptyset]\right)\right)}(x) = \int_x^1 \mathrm{d}y \, \mathcal{H}^*_{(\alpha_k;\emptyset)}(y) \underset{(2.3.27)}{=} \int_x^1 \mathrm{d}y \, \alpha_k(y) = \mathrm{H}^*_{\alpha_k}(x).$$

It suffices therefore to iterate the procedure to build the pure generalized word associated to the word $\vec{\alpha} = (\alpha_1, \ldots, \alpha_k)$, i.e., we define

$$\underline{V_{\alpha,1}} := \left(\lambda \cdot \alpha_{k-1}(\lambda), \left(\left[\frac{1}{\lambda};\emptyset\right], [\alpha_k;\emptyset]\right)\right)$$

and then build the words $\underline{V_{\alpha,2}}, \ldots, \underline{V_{\alpha,k}}$ where for all $i \in \{2, \ldots, k-2\}$ we have

$$\underline{V_{\alpha,i+1}} := \left(\lambda \cdot \alpha_{k-i-1}(\lambda), \left[\left(\frac{1}{\lambda};\emptyset\right], \underline{V_{\alpha,i}}\right)\right)$$

and

$$\underline{V_\alpha} = \underline{V_{\alpha,k}} := \left(\lambda, \left(\left(\frac{1}{\lambda};\emptyset\right), \underline{V_{\alpha,k-1}}\right)\right).$$

By construction we get

$$\mathrm{H}^*_{\alpha_1,\ldots,\alpha_k}(x) = \mathcal{H}^*_{\underline{V_\alpha}}(x)$$

which concludes the proof. □

Example 2.3.7. *We rewrite the following GRPLs as pure GCPLs:*

$$\mathrm{H}^*_{\frac{1}{\sqrt{\lambda(1-\lambda)}}}(x) = \mathcal{H}^*_{\left(\lambda,\left(\left[\frac{1}{\lambda};\emptyset\right],\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right]\right)\right)}(x)$$

$$\mathrm{H}^*_{\frac{1}{1+\lambda},\frac{1}{2+\lambda}}(x) = \mathcal{H}^*_{\left(\lambda,\left(\left[\frac{1}{\lambda};\emptyset\right],\left(\frac{\lambda}{1+\lambda},\left(\left[\frac{1}{\lambda};\emptyset\right],\left[\frac{1}{2+\lambda};\emptyset\right]\right)\right)\right)\right)}(x).$$

One can immediately deduce from Property 2.3.2 the following corollary:

Property 2.3.3. *Any GCPL can be rewritten as a pure GCPL.*

Proof. Suppose we consider a GCPL $\mathcal{H}^*_{\underline{V}}(x)$. If $\underline{V}$ contains any non-pure w-letter $[f_i;\vec{w}]$, it corresponds by definition to the function $f_i(x)\mathrm{H}^*_{\vec{w}}(x)$. So by Property 2.3.2, there exists a pure generalized word $\underline{V_w}$ such that

$$\mathcal{H}^*_{[\mathrm{f_i};\vec{\mathrm{w}}]} = \mathcal{H}^*_{(\mathrm{f_i},\underline{\mathrm{V_w}})}.$$

Replacing all the non-pure w-letters in such a way, we achieve to get a pure GCPL representation. □

We have not only implemented GCPL functions in the `RICA` package but also wrote several functions that allow us to convert those to and from `GLb` functions when possible, or in general to their explicit integral representations[28].

First, the GCPL functions are called `GLCb` functions in `RICA`. To understand how they work we can write the GCPL defining relations in Mathematica formalism:

```
GLCb[{1,{}},x]=1
GLCb[{f[VarGL],{g1[VarGL],...,gn[VarGL]}},x]=
f[x]*GLb[{g1[VarGL],...gn[varGL]},x]]
GLCb[{f[VarGL],V},x]=f[x]*GLCb[V,x]
GLCb[{V,W},x]=MellinConvolve[GLCb[V,x],GLCb[W,x],x].
```

Here we wrote explicitely the word $\vec{w} = (g_1, \ldots, g_n)$, $\underline{V}$ and $\underline{W}$ are generalized words, and by `MellinConvolve` we mean the standard Mellin convolution or the one in the +-prescription. We see from the first and second equation that in order to represent a w-letter we simply use the following syntax: {letter, word}. Furthermore, an empty letter corresponds to an empty set {}. Because the GCPL have usually quite a heavy indexing, it is more practical to see explicitely which (nested) convolution integrals it represents, and for that, we implemented in `RICA` the function `GLCbToIntegral`. Suppose we have the following GCPL:

[28] For more details see Appendix A.

```
In[10]:= glcb1 = GLCb[{{{1/VarGL, {1/(Sqrt[VarGL(1 - VarGL)]), 1/(VarGL Sqrt[1 - VarGL])}}},
                      {1/VarGL, {1/(VarGL + 2)}}}, x];
```

We can now simply call `GLCbToIntegral` that takes as an argument a single GLCb function[29]:

```
In[11]:= GLCbToIntegral[glcb1]

Out[11]= (Integrate[(GLb[{1/(VarGL+2)}, x1] GLb[{{1/Sqrt[-((VarGL-1)VarGL)], 1/Sqrt[1-VarGL VarGL]}}, x/x1])/x1, {x1, x, 1}])/x
```

This function also identifies automatically if the generalized word relies on the +-prescription, i.e., it writes the proper convergent integral according to Property 2.1.3. For example, if we define the following GCPL:

```
In[12]:= glcb2 = GLCb[{{{1/(VarGL Sqrt[1 - VarGL]), {}}}, {1/(1 - VarGL) {1/(1 - VarGL)}}}, x];
```

we obtain the following explicit representation:

```
In[13]:= GLCbToIntegral[glcb2]

Out[13]= -(Integrate[(GLb[{1/(1-VarGL)}, x1])/(1-x1), {x1, 0, x}])/Sqrt[1 - x x]
         - Integrate[GLb[{1/(1-VarGL)}, x/x1] (1/Sqrt[1-x1] - 1/Sqrt[1-x])/((x - x1) x1), {x1, x, 1}]
```

In order to prevent Mathematica from evaluating these integrals, we put the output in inactive mode. In order to try to evaluate them explicitelly with Mathematica, one has to apply in addition the `Activate` function of Mathematica.

Example 2.3.8. *If we want to compute the Mellin inverse of the following binomial sum*

$$S(n) = \sum_{i=1}^{n} \binom{2i}{i} \sum_{j=1}^{i} \frac{1}{8^j} \binom{2j}{j}, \tag{2.3.35}$$

we immediately see that it does not fit any of the cases of the Theorems 2.3.1 – 2.3.4. More precisely, when we start to execute our recursive method in the spirit of Example 2.3.1, we enter in the following step:

$$\begin{aligned} S(n) &= \sum_{i=1}^{n} \binom{2i}{i} \sum_{j=1}^{i} \frac{1}{2^j \pi^2} \mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](j) \\ &= \frac{1}{\pi^2} \sum_{i=1}^{n} \frac{1}{2^i} \binom{2i}{i} \mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](i) - \frac{\mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](0)}{\pi^2} \sum_{i=1}^{n} \binom{2i}{i} \end{aligned}$$

[29] It has several options that are specified in Appendix A.1.8.

$$= \frac{1}{\pi^2}\sum_{i=1}^{n}\frac{1}{2^i}\boxed{\mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](i)\cdot\mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](i)}$$
$$-\frac{4\mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](0)}{\pi^2}\tilde{\mathbf{M}}_4\left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)}\right](n)$$

where we remind that

$$\tilde{\mathbf{M}}_4\left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)}\right](n) = \int_0^1 \mathrm{d}x\,((4x)^n-1)\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)};$$

compare Definition 2.1.11. The boxed quantity is the one for which we cannot manage to compute the convolution, neither by the theorems cited above — it does not match any pattern — nor by an explicit convolution. In this case, we use the GCPL formalism, i.e.,

$$\frac{1}{\sqrt{x(1-x)}}\star\frac{x}{(x-2)\sqrt{x(1-x)}} = \mathcal{H}^*_{\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{\lambda}{(\lambda-2)\sqrt{\lambda(1-\lambda)}};\emptyset\right]\right)}(x),$$

and then we can finish the computation by replacing the convolution by the GCPL. Summing it up we obtain the output:

$$S(n) = \frac{2}{\pi^2}\tilde{\mathbf{M}}_2\left[\frac{x}{2x-1}\mathcal{H}^*_{\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{\lambda}{(\lambda-2)\sqrt{\lambda(1-\lambda)}};\emptyset\right]\right)}(x)\right](n)$$
$$-\frac{4\mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](0)}{\pi^2}\tilde{\mathbf{M}}_4\left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)}\right](n).$$

with $\tilde{M}$ defined in Definition 2.1.11.

Remark 2.3.8. *There are two reasons why the GCPL functions are useful. First, we can always finish a computation and return some result as long as the input of* `SumToMellin` *is a sum from the extended binomial sum class, by construction of the GCPLs. This technical feature has been illustrated in the Example 2.3.8 above, where the sum was not looking particularly complicated but gave rise to a case that could not be dealt with in the classical approach. Secondly, we are interested in asymptotics of (extended) binomial sums, as we will see in the next sections. This result might be enough — either because the dominant term does not involve GCPL functions (which is why we need to be able to always finish the computations), or more generally because asymptotics of GCPL functions can be achieved around certain points. The first reason is once again visible in the example above: indeed,*

because of the 2^n factor in $\tilde{\mathbf{M}}_2$, it turns out once we finish the computations that the term containing the GCPL is subdominant, as opposed to the simpler term

$$-\frac{4\mathbf{M}\left[\frac{x}{(x-2)\sqrt{(1-x)x}}\right](0)}{\pi^2}\tilde{\mathbf{M}}_4\left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)}\right](n)$$

which contains a 4^n and is the only one we need to deal with if we want to get the (dominant) asymptotics.

2.3.5. The problem of extended convolution multiplication formulas

To finish the part concerning Mellin inversions of binomial sums, we want to discuss a particular issue that arose during the computation of certain convolutions. Suppose we are given a function $f : (0, 1) \to \mathbb{R}$ in the +-prescription, and a function $g : (0; 1) \to \mathbb{R}$ which is integrable around 1. In this case, refering to [16] for example, we have that the following holds:

$$\mathbf{M}^{+}\left[f(x)\right](n)\mathbf{M}\left[g(x)\right](n) = \mathbf{M}\left[(f_{+} \star g)(x)\right](n), \tag{2.3.36}$$

which is the multiplicative formula from Property 2.1.4 in the +-prescription. While this formula holds true if we assume that $f(x) \underset{x\to 1}{\sim} \alpha(1-x)$, $\alpha \in \mathbb{R}$, which is the case for all the letters used in Remark 2.1.11, it actually needs correction when we consider more special situations, in particular if

$$f(x) \underset{x\to 1}{\sim} \frac{A}{(x-1)(1-x)^{\alpha}} \quad g(x) \underset{x\to 1}{\sim} -B\frac{(1-x)^{\alpha}}{x-1}, \quad A, B \in \mathbb{R},\ 0 < \alpha < 1. \tag{2.3.37}$$

Such a situation can happen as a result of applying the Mellin summation formula (2.1.27) in certain binomial sums. For example consider the following sum:

$$\mathcal{BS}_{((1,1,0),(1,1/4,0))}(n) = \sum_{i=1}^{n}\binom{2i}{i}\sum_{j=1}^{i}\frac{1}{4^j}\binom{2j}{j}. \tag{2.3.38}$$

Using the algorithm described in Section 2.3 and implemented in `RICA`, we can obtain a Mellin representation using GCPL functions:

$$\text{In[14]:=}\ \text{SumToMellin}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}, \frac{\mathbf{4}^{\text{VarGL}}}{\text{VarGL}\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}}\right\}, \mathbf{n}\right], \mathbf{C}, \mathbf{x}, \text{UseGLCb} \to \text{True}\right]$$

Out[14]= $\left\{\frac{4\text{Mellin}\left[4^{\mathtt{n}}\mathtt{x}^{\mathtt{n}}-1,\frac{\text{xGLCb}\left[\left\{\left\{\frac{1}{(\text{VarGL}-1)\sqrt{1-\text{VarGL}}},\{\}\right\},\left\{\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}},\{\}\right\}\right\},\mathtt{x}\right]}{4\mathtt{x}-1}\right]}{\pi^2},\{\}\right\}$

We see here that the result involves the function

$$\mathcal{H}^*_{\left(\left[\frac{1}{(\lambda-1)\sqrt{1-\lambda}};\emptyset\right],\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right]\right)}(x)=\underbrace{\left(\frac{1}{(1-x)\sqrt{1-x}}\right)}_{=:f(x)}\star\underbrace{\left(\frac{1}{\sqrt{x(1-x)}}\right)}_{=:g(x)} \tag{2.3.39}$$

where

$$g(x)=\frac{1}{\sqrt{x(1-x)}}\underset{x\to 1}{\sim}\frac{1}{\sqrt{1-x}},$$

which means f and g match the form from (2.3.37). Let us verify whether the formula (2.3.37) holds. First, we can compute the left-hand side (LHS):

$$\mathbf{M}^+\left[f(x)\right](n)=\int_0^1 \mathrm{d}x\,(x^n-1)f(x)=\frac{2\sqrt{\pi}\Gamma(n+1)}{\Gamma\left(n+\frac{1}{2}\right)}-2, \tag{2.3.40}$$

$$\mathbf{M}\left[g(x)\right](n)=\int_0^1 \mathrm{d}x\,x^n g(x)=\frac{\sqrt{\pi}\Gamma\left(n+\frac{1}{2}\right)}{\Gamma(n+1)}. \tag{2.3.41}$$

Therefore, further manipulations of both expressions under the assumption that $n\in\mathbb{N}$ and multiplying them, the left hand side simplifies to:

$$\mathbf{M}^+\left[f(x)\right](n)\cdot\mathbf{M}\left[g(x)\right](n)=2\pi-\frac{2\sqrt{\pi}\Gamma\left(n+\frac{1}{2}\right)}{n!} \tag{2.3.42}$$

To get the right-hand side, we first need to compute the convolution in the $+$-prescription of f and g. We use for that Property 2.1.3 and write the convolution as definite integrals[30]:

$$\begin{aligned}(f_+\star g)(x)&=\int_x^1 \mathrm{d}y\,f(y)\left[\frac{1}{y}g\left(\frac{x}{y}\right)-g(x)\right]-g(x)\int_0^x \mathrm{d}y\,f(y)\\ \text{COV}\to&=\int_0^1 \mathrm{d}z\,\frac{(x-1)\left((1+(x-1)z)g(x)-g\left(\frac{x}{(x-1)z+1}\right)\right)f((x-1)z+1)}{(x-1)z+1}\\ &\quad-g(x)x\int_0^1 \mathrm{d}z\,f(xz)\\ &=\int_0^1 \mathrm{d}z\,\frac{(1-x)x^2}{((1-x)x(1-xz))^{3/2}}+\frac{\sqrt{(1-x)x(1-z)}-\sqrt{(1-x)x}}{xz\sqrt{(1-x)^3(1-z)z}}\end{aligned}$$

[30] COV stands for "change of variables".

$$= \frac{2}{\sqrt{x(1-x)}}. \tag{2.3.43}$$

As it turns out in this case, the +-prescription integral luckily turns out to yield a rather simple result. It remains to compute the RHS of (2.3.36), which yields:

$$\mathbf{M}\left[(f_+ \star g)(x)\right](n) = \int_0^1 \mathrm{d}x^n \, (f_+ \star g)(x) = \int_0^1 \mathrm{d}x \, \frac{2x^n}{\sqrt{x(1-x)}} = -\frac{2\sqrt{\pi}\Gamma\left(n+\frac{1}{2}\right)}{\Gamma(n+1)}.$$

This result shows that in the configuration we considered, we have

$$\mathbf{M}^+\left[f(x)\right](n) \cdot \mathbf{M}\left[g(x)\right](n) \neq \mathbf{M}\left[(f_+ \star g)(x)\right](n).$$

Nevertheless, if now we compute instead

$$\mathbf{M}^+\left[(f_+ \star g)(x)\right](n) = \int_0^1 \mathrm{d}x \, \frac{2(x^n-1)}{\sqrt{x(1-x)}} = 2\pi - \frac{2\sqrt{\pi}\Gamma\left(n+\frac{1}{2}\right)}{\Gamma(n+1)}, \tag{2.3.44}$$

the equality is restored! This result does appear to be quite surprising because while the +-prescription is defined for functions that are divergent[31] at $x = 1$, the convolution of f and g yields as expected an expression that is convergent at $x = 1$, since

$$\frac{2}{\sqrt{x(1-x)}} \underset{x\to 1}{\sim} \frac{2}{\sqrt{1-x}}.$$

Motivated by the example above, we want find other such cases when (2.3.37) holds where we should add an extra correction to the (2.1.4) formula, leading to:

$$\mathbf{M}^+\left[f(x)\right](n)\mathbf{M}\left[g(x)\right](n) = \mathbf{M}^+\left[(f_+ \star g)(x)\right](n). \tag{2.3.45}$$

Intuitively speaking, one can understand the need of a correction because

$$f(x)g(x) \underset{x\to 1}{\sim} -\frac{AB}{(x-1)^2}.$$

That is to say, when writing the Mellin transform in the +-prescription for the function f is enough for it to become *integrable*, the fact that the product $f(x)g(x)$ has a pole of order two is what makes the convolution "less integrable" than a function with a pole of order $\alpha \in [1;2)$. Therefore the Mellin transform of the convolution has to compensate for that also. Before continuing, we need first a preliminary definition of some classes of special functions; see for instance [99–103] and [79].

[31] With a pole of order strictly less than 2.

Definition 2.3.5 (Hypergeometric functions)**.** *Let $p, q \in \mathbb{N}$. We define the* ***generalized hypergeometric functions*** *(or* ***PFQs****) and* ***regularised generalized hypergeometric functions*** *(or* ***regularized PFQs****) using respectively the notations ${}_pF_q$ and ${}_p\tilde{F}_q$.*

$$ {}_pF_q(\alpha_1, \dots, \alpha_p; \beta_1, \dots, \beta_q; x) := \sum_{n=0}^{\infty} \frac{(\alpha_1)_n \cdots (\alpha_p)_n}{(\beta_1)_n \cdots (\beta_q)_n} \frac{x^n}{n!}, \tag{2.3.46} $$

$$ {}_p\tilde{F}_q(\alpha_1, \dots, \alpha_p; \beta_1, \dots, \beta_q; x) := \frac{{}_pF_q(\alpha_1, \dots, \alpha_p; \beta_1, \dots, \beta_q; x)}{\prod_{i=1}^{q} \Gamma(\beta_i)}. \tag{2.3.47} $$

Definition 2.3.6 (Appell functions)**.** *Let $a_1, b_1, b_2 \in \mathbb{K}$, x, y such that $|x|, |y| < 1$. We define the* ***Appell function of the first kind*** *as:*

$$ F_1(a; b_1, b_2; c; x, y) = \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \frac{(a)_{m+n} (b_1)_m (b_2)_n}{(c)_{m+n}} x^m y^n. \tag{2.3.48} $$

We manage to prove a theorem that shows (2.3.45) holds for certain classes of functions.

Theorem 2.3.5. *Define:*

$$ f_\alpha(x) = \frac{1}{(x-1)(1-x)^\alpha} \text{ and } g_\alpha(x) = -\frac{(1-x)^\alpha}{x-1}, \quad 0 < \alpha < 1. $$

Then we have

$$ \mathbf{M}^+ \left[(f_\alpha \star g_\alpha)_+(x) \right](n) = \mathbf{M}^+ \left[f_\alpha(x) \right](n) \cdot \mathbf{M} \left[g_\alpha(x) \right](n) \tag{2.3.49} $$

$$ = \frac{n!(\pi n! \csc(\pi\alpha) - \Gamma(\alpha)\Gamma(n-\alpha+1))}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}. \tag{2.3.50} $$

Proof. First we compute the left-hand side LHS(n) of (2.3.45). This can be done explicitely. We obtain:

$$ \mathbf{M}^+ \left[f(x) \right](n) = \int_0^1 \mathrm{d}x \, (x^n - 1) f(x) = \frac{\pi \csc(\pi\alpha)\Gamma(n+1)}{\Gamma(\alpha+1)\Gamma(n-\alpha+1)} - \frac{1}{\alpha}, $$

$$ \mathbf{M} \left[g(x) \right](n) = \int_0^1 \mathrm{d}x \, x^n \, g(x) = \frac{\Gamma(\alpha)\Gamma(n+1)}{\Gamma(n+\alpha+1)}. $$

Combining both and massaging a bit the expression we get:

$$ \mathrm{LHS}(n) = \mathbf{M}^+ \left[f(x) \right](n) \cdot \mathbf{M} \left[g(x) \right](n) = \frac{n!(\pi n! \csc(\pi\alpha) - \Gamma(\alpha)\Gamma(n-\alpha+1))}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}. $$

Now we have to compute the right hand side RHS(n) of (2.3.45). First, we compute the +-prescription convolution. Given the presence of the $(1-x)^\alpha$ in both f and g, the convolution will give rise to hypergeometric functions in the worst case. In order to be able to perform the computation of the convolution in Mathematica, we have to do the following:

- First, we compute the indefinite integral for the +-prescription function, i.e., $f_{\alpha\,+} \star g_\alpha$ given in (2.1.30), but where we make the change of variable $y \to \frac{x}{1+(x-1)y}$ and consider for the moment only the indefinite integral:

$$I(x,y) = \underbrace{\int \mathrm{d}y \, \frac{(x-1)f((x-1)y+1)\left(((x-1)y+1)g(x) - g\left(\frac{x}{(x-1)y+1}\right)\right)}{(x-1)y+1}}_{I_1(x,y)}$$
$$\underbrace{- g(x) \int \mathrm{d}y \, f(xy)}_{I_2(x,y)} + C_0.$$

  for $C_0 \in \mathbb{R}$. The two integrals can be simplified further to

$$\begin{aligned} I_1(x,y) &= -\int \mathrm{d}y \, \frac{y^{-\alpha-1}\left(\left(\frac{1-y}{(x-1)y+1}\right)^\alpha + y - 1\right)}{(x-1)(y-1)} \\ &= \frac{y^{-\alpha}}{\alpha(x-1)} \\ &\quad - \frac{y^{-\alpha}(\alpha y F_1(1-\alpha;1-\alpha,\alpha;2-\alpha;y,y-xy) + (\alpha-1)F_1(-\alpha;-\alpha,\alpha;1-\alpha;y,y-xy))}{(\alpha-1)\alpha(x-1)} \\ &\quad + C_1 \end{aligned}$$

  and

$$I_2(x,y) = g(x) \int \mathrm{d}y \, f(x\,y) = \frac{(1-x)^\alpha (1-xy)^{-\alpha}}{\alpha(x-1)} + C_2.$$

  where $C_1, C_2 \in \mathbb{R}$ are two integration constants.

- Now, we define the following quantity:

$$I(x,\epsilon) := I(x,1-\epsilon) - I(x,\epsilon)$$

  and obtain the excepted convolution by taking the limit of $\epsilon \to 0^+$:

$$h_\alpha(x) := (f_\alpha \star g_\alpha)_+(x) = \lim_{\epsilon \to 0^+} I(x,\epsilon).$$

Performing the computation[32], we obtain:

$$h_\alpha(x) = \frac{\pi\alpha \csc(\pi\alpha)({}_2F_1(1-\alpha,\alpha;1;1-x) - {}_2F_1(-\alpha,\alpha;1;1-x)) + (1-x)^\alpha}{\alpha(x-1)}. \tag{2.3.51}$$

In particular, if we take $\alpha = \frac{1}{2}$, which corresponds to the typical case we are dealing with our alphabet (2.1.43), containing square root letters, the expression simplifies to:

$$\left(f_{\frac{1}{2}} \star g_{\frac{1}{2}}\right)_+ (x) = \frac{2\left(K(1-x) + \sqrt{1-x} - E(1-x)\right)}{x-1} \tag{2.3.52}$$

involving the following elliptic functions:

– The incomplete elliptic integral of the second kind:

$$E(x|\phi) := \int_0^\phi \mathrm{d}\theta \sqrt{1 - x\sin^2\theta}, \quad E(x) = E\left(x\middle|\frac{\pi}{2}\right). \tag{2.3.53}$$

– The complete elliptic integral of the first kind:

$$K(x) = F\left(\frac{\pi}{2}\middle| x\right), \text{ where } F(\phi|x) = \int_0^\phi \mathrm{d}\theta \frac{1}{\sqrt{1 - x\sin^2(\theta)}}. \tag{2.3.54}$$

- Now we need to compute the actual Mellin transform of $h_\alpha(x)$, i.e.,

$$\mathbf{M}^+\left[h_\alpha(x)\right](n) = \int_0^1 \mathrm{d}x\,(x^n - 1)h_\alpha(x) = \sum_{i=0}^{n-1}\int_0^1 \mathrm{d}x\, x^i(x-1)h_\alpha(x).$$

It turns out that neither the full integral, nor the summand of the geometrically expanded Mellin transform can be easily computed. In particular, Mathematica returns us some Meijer G expressions which are close to impossible to manipulate. But it turns out that we manage to compute an antiderivative of the most complicated part of $h_\alpha(x)$ using Maple which gives

$$\int^x \mathrm{d}z\, \frac{{}_2F_1(1-\alpha,\alpha;1;1-z) - {}_2F_1(-\alpha,\alpha;1;1-z)}{(z-1)} = \\ \alpha^2(1-x)\,{}_4F_3(1,1,\alpha+1,1-\alpha;2,2,2;1-x)$$

[32] From a technical point of view, we obtain the result by peforming in Mathematica the series expansion of $I(x,\epsilon)$ and taking the constant term — the absence of poles in ϵ is then an additional indicator of the correctness of the procedure, as by construction the integrals making up $(f_\alpha \star g_\alpha)_+(x)$ should be convergent.

$$+ (1-\alpha)\alpha(1-x)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1-x) + C_3, \quad (2.3.55)$$

where $C_3 \in \mathbb{R}$. Thanks to that we can compute the full antiderivate of $h_\alpha(x)$:

$$\begin{aligned}\int^x \mathrm{d}z\, h_\alpha(z) = \pi\alpha(x-1)\csc(\pi\alpha)\,&((\alpha-1)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1-x)\\ &-\alpha\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,2;1-x))\\ &+\frac{(1-x)^\alpha}{\alpha^2} + C_4, \quad C_4 \in \mathbb{R}. \end{aligned} \quad (2.3.56)$$

- Finally, we aim at performing integration by parts to obtain the full Mellin convolution of $h_\alpha(x)$:

$$\int_0^1 \mathrm{d}x\, h_\alpha(x)\,(x^n-1) = \underbrace{\left[(x^n-1)\int^x \mathrm{d}z\, h(z)\right]_0^1}_{=:P_1} \underbrace{-n\int_0^1 \mathrm{d}x\, x^{n-1}\int^x \mathrm{d}z\, h_\alpha(z)}_{=:P_2}. \quad (2.3.57)$$

To carry this out, we verify that $h_\alpha(x)$ is integrable at $x=0$ and $x=1$ respectively (i.e., its antiverivative is defined at those points). This is the case for $\alpha \in (0;1)$ because:

$$\begin{aligned} h_\alpha(x) &\underset{x\to 0}{=} \frac{\pi\csc(\pi\alpha)\log(x)}{\Gamma(1-\alpha)\Gamma(\alpha)} + \mathcal{O}(1)\\ h_\alpha(x) &\underset{x\to 1}{=} -\frac{1}{\alpha(1-x)^{1-\alpha}} + \mathcal{O}\left((1-x)^\alpha\right). \end{aligned}$$

- The constant term P_1 simplifies to:

$$\begin{aligned} P_1 = \pi\alpha\csc(\pi\alpha)\,&(\alpha\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,2;1)\\ &-(\alpha-1)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1)) + \frac{1}{\alpha^2} + C_4. \end{aligned} \quad (2.3.58)$$

Remark 2.3.9. *Technically, Mathematica does not manage to directly compute the constant. Here we made the change of variables $x \to 1-x$ so that*

$$P_1 = -\left[((1-x)^n-1)\int^{1-x} \mathrm{d}z\, h_\alpha(z)\right]$$

and the bound terms at $x=0$ and $x=1$ are obtained by limit computations.

- Further we integrate the second part, P_2 in (2.3.57):

$$P_2 = \pi\alpha n!\csc(\pi\alpha)\Big((\alpha-1)\,{}_4\tilde{F}_3(1,1,2-\alpha,\alpha+1;2,2,n+2;1) \\ -\alpha\,{}_4\tilde{F}_3(1,1,1-\alpha,\alpha+1;2,2,n+2;1)\Big) - \frac{n!\Gamma(\alpha)}{\alpha\Gamma(n+\alpha+1)} - C_4 \quad (2.3.59)$$

and collecting both P_1 and P_2 the full expression for the Mellin transform of $h_\alpha(x)$ is:

$$\begin{aligned}\mathbf{M}^+\left[h_\alpha(x)\right](n) = \pi\alpha\csc(\pi\alpha)\Big[n!\Big(&(\alpha-1)\,{}_4\tilde{F}_3(1,1,2-\alpha,\alpha+1;2,2,n+2;1)\\ &-\alpha\,{}_4\tilde{F}_3(1,1,1-\alpha,\alpha+1;2,2,n+2;1)\Big)\\ &+\alpha\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,2;1)\\ &-(\alpha-1)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1)\Big]\\ &+\frac{1-\frac{\alpha n!\Gamma(\alpha)}{\Gamma(n+\alpha+1)}}{\alpha^2}.\end{aligned} \quad (2.3.60)$$

- Now we basically need to prove that:

$$d_\alpha(x) := \mathbf{M}^+\left[h_\alpha(x)\right](n) - \mathbf{M}^+\left[f_\alpha(x)\right](n)\cdot\mathbf{M}\left[g(x)\right](n) \stackrel{?}{=} 0$$

where the difference is:

$$\begin{aligned}d_\alpha(x) =&\frac{1}{\alpha^2} - \frac{\pi(n!)^2\csc(\pi\alpha)}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}\\ &+\pi\alpha\csc(\pi\alpha)\Big(n!\Big((\alpha-1)\,{}_4\tilde{F}_3(1,1,2-\alpha,\alpha+1;2,2,n+2;1)\\ &-\alpha\,{}_4\tilde{F}_3(1,1,1-\alpha,\alpha+1;2,2,n+2;1)\Big)\\ &+\alpha\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,2;1)\\ &-(\alpha-1)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1)\Big).\end{aligned} \quad (2.3.61)$$

- We expand the expression $d_\alpha(x)$ and use the definition (2.3.47) as well as the fact that $\Gamma(1+n)/\Gamma(n+2) = 1/(n+1)$ to obtain the following sum of standard hypergeometric functions:

$$\begin{aligned}d_\alpha(x) =&\frac{1}{\alpha^2} - \frac{\pi\csc(\pi\alpha)\Gamma(n+1)^2}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}\\ &\pi\alpha^2\csc(\pi\alpha)\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,2;1)\\ &-\pi\alpha^2\csc(\pi\alpha)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1)\\ &+\pi\alpha\csc(\pi\alpha)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,2;1)\end{aligned}$$

$$+\frac{\pi\alpha^2\csc(\pi\alpha)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,n+2;1)}{n+1}$$
$$-\frac{\pi\alpha^2\csc(\pi\alpha)\,{}_4F_3(1,1,1-\alpha,\alpha+1;2,2,n+2;1)}{n+1}$$
$$-\frac{\pi\alpha\csc(\pi\alpha)\,{}_4F_3(1,1,2-\alpha,\alpha+1;2,2,n+2;1)}{n+1}. \tag{2.3.62}$$

- We now use (2.3.46) to replace the PFQs that appear in (2.3.62) by their series representation, then combine all the sums together using standard Gamma function properties, which drastically simplifies $d_\alpha(x)$ to

$$d_\alpha(x)=\sum_{p=0}^{\infty}\frac{(\Gamma(n+p+2)-\Gamma(n+1)\Gamma(p+2))\Gamma(p-\alpha+1)\Gamma(p+\alpha+1)}{\Gamma(p+2)^2\Gamma(n+p+2)}$$
$$+\frac{1}{\alpha^2}-\frac{\pi\csc(\pi\alpha)\Gamma(n+1)^2}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}. \tag{2.3.63}$$

- The sum is then simplified in Mathematica, or using hypergeometric functions identities. We obtain the following expression

$$d_\alpha(x)=\frac{\Gamma(1-\alpha)\Gamma(\alpha+1)}{\alpha^2}+\Gamma(-\alpha)\Gamma(\alpha)$$
$$-\frac{\Gamma(-\alpha)\Gamma(\alpha)\Gamma(n+1)^2}{\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}-\frac{\pi\csc(\pi\alpha)\Gamma(n+1)^2}{\alpha\Gamma(n-\alpha+1)\Gamma(n+\alpha+1)}. \tag{2.3.64}$$

  Using $\Gamma(1+n)=n\Gamma(n)$ the first two terms cancel out, and with the identity $\Gamma(n)\Gamma(1-n)=\pi\csc(\pi n)$ the last two terms cancel out, leaving just 0. This yields[33]

$$\forall n\in\mathbb{N},\ \mathbf{M}^+\left[(f_\alpha\star g_\alpha)_+(x)\right](n)=\mathbf{M}^+\left[f_\alpha(x)\right](n)\cdot\mathbf{M}\left[g_\alpha(x)\right](n)$$

which concludes the proof. □

Unfortunately, this result does not seem to hold for other classes of functions. For instance, let us take the following two functions:

$$f(x)=\frac{x}{x-1}\frac{1}{\sqrt{x(1-x)}},\quad g(x)=f_{w_3}(x)=\frac{1}{x\sqrt{1-x}},$$

which do not satisfy the assumptions of the Theorem 2.3.5. We compute as usual

[33] The case $n=0$ simply comes from the fact that $x^0-1=0$ so both sides are identically zero.

the individual convolutions[34]

$$\begin{aligned}\mathbf{M}^{+}\left[f(x)\right](n) &= \int_0^1 \mathrm{d}x\,(x^n-1)f(x)\\ &= \sum_{i=0}^{n-1}\int_0^1 \mathrm{d}x\,x^i(x-1)f(x)\\ &= \sum_{i=0}^{n-1}\frac{\sqrt{\pi}\Gamma\left(i+\frac{3}{2}\right)}{\Gamma(i+2)}\\ &= \frac{2\sqrt{\pi}\Gamma\left(n+\frac{3}{2}\right)}{n!}-\pi,\end{aligned} \tag{2.3.65}$$

$$\mathbf{M}\left[g(x)\right](n) = \int_0^1 \mathrm{d}x\,x^n g(x) = \frac{\sqrt{\pi}\Gamma(n)}{\Gamma\left(n+\frac{1}{2}\right)}. \tag{2.3.66}$$

We combine the expressions together and we obtain:

$$\mathbf{M}^{+}\left[f(x)\right](n)\cdot\mathbf{M}\left[g(x)\right](n) = \pi\left(\frac{1}{n}-\frac{\sqrt{\pi}\Gamma(n)}{\Gamma\left(n+\frac{1}{2}\right)}+2\right). \tag{2.3.67}$$

To compute the RHS we once again compute the convolution between f and g:

$$\begin{aligned}(f_+\star g)(x) &= \int_x^1 \mathrm{d}y\,f(y)\left[\frac{1}{y}g\left(\frac{x}{y}\right)-g(x)\right]-g(x)\int_0^x \mathrm{d}y\,f(y)\\ &= \int_x^1 \mathrm{d}y\,\frac{x\left(\sqrt{1-x}-\sqrt{1-y}\right)}{y(x-y)\sqrt{(x-1)x(y-1)(y-x)}}\\ &\quad-\frac{1}{\sqrt{1-x}x}\int_0^x \mathrm{d}y\,\frac{y}{(y-1)\sqrt{(1-y)y}}\\ &= \frac{\pi(x-1)+2\sqrt{x}+2\sqrt{1-x}\tan^{-1}\left(\sqrt{\frac{1}{x}-1}\right)}{(x-1)x}\\ &\quad-\frac{2\sin^{-1}\left(\sqrt{x}\right)-2\sqrt{-\frac{x}{x-1}}}{\sqrt{1-x}x}\end{aligned}$$

[34] As the Mellin transform of f cannot be immediately computed, we use the geometric series [7]

$$\frac{x^n-1}{x-1} = \sum_{i=0}^{n-1} x^i$$

to compute the Mellin transforms before summing them up.

$$= \frac{\pi - \frac{2\left(\sin^{-1}\left(\sqrt{x}\right)+\tan^{-1}\left(\sqrt{\frac{1}{x}-1}\right)\right)}{\sqrt{1-x}}}{x}. \tag{2.3.68}$$

Moreover, because we have

$$\frac{\mathrm{d}}{\mathrm{d}x}\left(\sin^{-1}\left(\sqrt{x}\right)+\tan^{-1}\left(\sqrt{\frac{1}{x}-1}\right)\right)=0, \quad \sin^{-1}\left(\sqrt{\frac{1}{2}}\right)+\tan^{-1}(1)=\frac{\pi}{2}, \tag{2.3.69}$$

we finally get

$$(f_+ \star g)(x) = \frac{\pi - \frac{\pi}{\sqrt{1-x}}}{x}. \tag{2.3.70}$$

Now we can finally compute the Mellin transform in the +-prescription of $(f_+ \star g)(x)$, which yields:

$$\mathbf{M}^+\left[(f_+ \star g)(x)\right](n) = \int_0^1 \mathrm{d}x\,(x^n-1)\left(\frac{\pi - \frac{\pi}{\sqrt{1-x}}}{x}\right) = \pi\left(\frac{1}{n} - \frac{\sqrt{\pi}\Gamma(n)}{\Gamma\left(n+\frac{1}{2}\right)} + \log(4)\right). \tag{2.3.71}$$

We therefore see that there is a constant discrepancy between the LHS and RHS of (2.3.45). More precisely, we get

$$\mathbf{M}^+\left[(f_+ \star g)(x)\right](n) - \mathbf{M}^+\left[f(x)\right](n)\cdot\mathbf{M}\left[g(x)\right](n) = 2\pi(\log 2 - 1). \tag{2.3.72}$$

This indicates that the formula (2.3.45) will be valid only for certain classes of functions and is not the full generalisation of (2.1.4) when $f(x)g(x)$ has a pole of order 2 at $x = 1$. The problem seems far from being trivial, and given that after several attempts the most general result we have managed to prove is Theorem 2.3.5, unless we encounter a situation of the form presented in the said theorem, we prevent `RICA` from performing such convolutions.

2.4. A theoretical overview of asymptotics and series expansions

In the previous section, we explored different techniques which aimed at finding an integral representation of (extended) binomial sums using the (inverse) Mellin transform. Despite the fact that the expression obtained as a result might look more complicated than the original sum, it has one major advantage over its sum counterpart — studying the integral's asymptotics is easier, for reasons that will soon become clear. Asymptotics of such expressions is of importance because if we know them, it is possible to obtain analytic continuation to $\mathbb{C}$ of Mellin transforms

and *in fine* of the original sums. We remind in this section some notations and classical results concerning asymptotics (Section 2.4.1), before proving some new results concerning asymptotic expansions of functions defined as integrals (Section 2.4.2). The last Section 2.5 of this chapter will be dedicated to the actual presentation of different techniques to compute asymptotic expansions of (extended) binomial sums.

2.4.1. About series expansions and asymptotics

First, we remind some classical definitions related to asymptotics in general, and in particular to the Bachmann-Landau notation following Donald Knuth's convention [104].

Definition 2.4.1 (Big-O notation)**.** *Let f, g be two functions defined in a neighbourhood of $a \in \mathbb{R}$. We write that*

$$f(x) \underset{x\to a}{=} \mathcal{O}\left(g(x)\right) \tag{2.4.1}$$

if there exists a constant $M > 0$ and $\epsilon > 0$ such that

$$|f(x)| \leq M|g(x)| \text{ when } |x-a| \leq \epsilon. \tag{2.4.2}$$

If the context is clear, we will omit the $x \to a$.

Definition 2.4.2 (Little-o notation)**.** *Let f, g be two functions defined in a neighbourhood of $a \in \mathbb{R}$. We write that*

$$f(x) \underset{x\to a}{=} o\left(g(x)\right) \tag{2.4.3}$$

if for all $\eta > 0$, there exists an $\epsilon_\eta > 0$ such that

$$|f(x)| \leq \eta|g(x)| \text{ when } |x-a| \leq \epsilon_\eta. \tag{2.4.4}$$

If the context is clear, we will omit the $x \to a$.

Remark 2.4.1. *The equalities (2.4.1) and (2.4.3) should be seen as a convenient abuse of notation, as in this case the equality is not symmetric, and a more proper notation would in principle be $f(x) \underset{x\to a}{\in} \mathcal{O}\left(g(x)\right)$ and $f(x) \underset{x\to a}{\in} o\left(g(x)\right)$.*

Definition 2.4.3 (Asymptotic equivalence)**.** *Let $I \subset \mathbb{R}$ be a finite union of intervals, $f : I \to \mathbb{R}$, $g : I \to \mathbb{R}$ two functions and $a \in I$ or a bound, possibly infinite, of I. We say that they are **asymptotically equivalent at** a if and only if*

$$\lim_{x\to a} \frac{f(x)}{g(x)} = 1. \tag{2.4.5}$$

We then write

$$f(x) \underset{x\to a}{\sim} g(x), \tag{2.4.6}$$

where we omit the $x \to a$ if it is clear from the context.

Remark 2.4.2. *An alternative definition of $f(x) \sim g(x)$ that takes care of situations where g is zero infinitely often as x goes to the limit value, is to define it as*

$$f(x) \sim g(x) \iff f(x) = g(x)(1 + o(1)). \tag{2.4.7}$$

Example 2.4.1. *Some examples of the usage of those notations are*

$$x^3 \sin\left(\frac{1}{x^2}\right) \underset{x\to 0}{=} \mathcal{O}\left(x^3\right),$$

$$\frac{1}{x^3} \underset{x\to+\infty}{=} o\left(\frac{1}{x}\right),$$

$$x^2 + x + 5 \underset{x\to+\infty}{\sim} x^2.$$

Using the asymptotic notation, we can now introduce Taylor's Theorem:

Theorem 2.4.1 (Taylor's Theorem)**.** *Let $n \in \mathbb{N}^\star$, $a, b \in \mathbb{R}$, $a < b$ and let a function $f : (a; b) \to \mathbb{R}$ be n-times differentiable at a point $c \in (a; b)$. Then there exists a function $R_n : (a; b) \to \mathbb{R}$ such that*

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(c)}{k!}(x - c)^k + R_n(x) \tag{2.4.8}$$

where $R_n(x) \underset{x\to c}{=} o\left((x - c)^n\right)$. If f is $(n + 1)$-differentiable, there exist different possible representations for the remainder, in particular:

- *The Lagrange form for the remainder*[35]*:*

$$\exists \xi \in (c; x), \quad R_n(x) = \frac{f^{(n+1)}(\xi)}{(n + 1)!}(x - c)^{n+1}. \tag{2.4.9}$$

- *The integral form for the remainder, which can be defined if $f^{(n)}$ is absolutely continuous on $[c; x]$:*

$$R_n(x) = \int_a^x \mathrm{d}t\, \frac{f^{(n+1)}(t)}{n!}(x - t)^n. \tag{2.4.10}$$

[35] For simplificity, we suppose that $x \geq a$ so all intervals will be written as $(a; x)$, but if it is the other way around then those should be $(x; a)$.

Definition 2.4.4 (Taylor series)**.** *Let $n \in \mathbb{N}^\star$, $a, b \in \mathbb{R}$, $a < b$ and let a function $f : (a; b) \to \mathbb{R}$ be infinitely many times differentiable at a point $c \in (a; b)$. We call its* ***Taylor series*** *around the point c the quantity*

$$\sum_{n=0}^{\infty} \frac{f^{(k)}(c)}{k!}(x-c)^k, \tag{2.4.11}$$

that can be convergent or divergent.

Theorem 2.4.2. *Let U be an (open) annulus in the complex plane defined by*

$$U = U(r_1, r_2) := \{z \in \mathbb{C} : r_1 < |z - a| < r_2\} \quad 0 < r_1 < r_2.$$

Let $\mathcal{C}(a, r)$ denote the circle of center $a \in \mathbb{C}$ and radius $r \in \mathbb{R}_+$, let f be an holomorphic, i.e., complex differentiable, function over U, $c \in U$. Then

$$f(z) = \sum_{n\in\mathbb{Z}} a_n(x-c)^n, \quad a_n = \oint_{\mathcal{C}(a,r)} \mathrm{d}z \, \frac{f(z)}{(z-a)^{n+1}}. \tag{2.4.12}$$

where $\oint_{\mathcal{C}(a,r)}$ indicates the contour integral along the $\mathcal{C}(a, r)$.

Definition 2.4.5 (Laurent series, Laurent coefficients)**.** *The series on the right hand side of (2.4.12) is called the* ***Laurent series of f around*** *c, and the a_n are called* ***Laurent coefficients****. If $a_n = 0$ for $n \in \mathbb{Z}_-$ the series is also called* ***power series expansion*** *and the coefficients* ***power series coefficients****.*

Definition 2.4.6 (Puiseux series)**.** *Let $k_0 \in \mathbb{Z}$, $n \in \mathbb{N}^\star$. A* ***Puiseux series*** *is a formal expression of the form*

$$f = \sum_{k=k_0}^{\infty} c_k T^{\frac{k}{n}} \quad c_k \in \mathbb{K}.$$

Example 2.4.2. *Here are examples of different series expansions:*

$$\exp(x) \underset{x\to 0}{=} \sum_{k=0}^{\infty} \frac{x^n}{n!},$$

$$\frac{1}{1+z+z^2} = \frac{1}{3}\sum_{n=-1}^{\infty} \left(\frac{i}{\sqrt{3}}\right)^n (z-j)^n, \quad j = -\frac{1}{2} - \frac{i\sqrt{3}}{2},$$

$$\frac{1}{x\sqrt{1-x}} \underset{x\to 1}{=} \sum_{k=-1}^{\infty} (1-x)^{k+\frac{1}{2}}.$$

While discussing the series expansion, we did not take into account the convergence concept. In particular, the strength of a series expansion such as Taylor's Theorem 2.4.1 lies in the fact that it makes a link with analytic functions.

Definition 2.4.7 (Real analytic function)**.** *A function $f : \mathbb{R} \to \mathbb{R}$ is said to be* ***real analytic*** *on an open set $\mathcal{D}$ if for any $x_0 \in \mathcal{D}$ one can write a series*

$$\sum_{k=0}^{n} a_k (x - x_0)^k, \qquad a_i \in \mathbb{R} \tag{2.4.13}$$

that converges to $f(x)$ for x in a certain neighbourhood of x_0.

In particular, when the Taylor series of a function f is defined at any point $x_0 \in I$, i.e., f is $\mathcal{C}^\infty(I)$, and converges to f in a certain neighbour of x_0, it means f is real analytic[36] on I.

Theorem 2.4.3. *Let $f : (a; b) \to \mathbb{R}$ infinitely many times differentiable and suppose that for all $x \in (a; b)$*

$$\lim_{n\to\infty} R_n(x) = 0 \tag{2.4.14}$$

then $\sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x-x_0)^k$ converges uniformly on $C_1(a, r)$ to the real analytic function

$$T_f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n.$$

If it turns out that for $x \in C_1(x_0, r)$, $f(x) = T_f(x)$ then f is real analytic at x_0.

Remark 2.4.3. *We will be mostly considering real analytic functions, which is the complicated case and where we need to deal properly with the remainder $R_n(x)$, i.e. show that it goes to 0 as $n \to +\infty$. Complex analytic functions have a much more rigid structure because they are exactly the holomorphic functions, so differentiability is equivalent to analyticity.*

Therefore, there exists three possible outcomes when computing the power series for a function around some point x_0 using Taylor's Theorem 2.4.1:

- The series converges uniformly to f, which means that f is analytic at x_0.
- The series converges uniformly to a certain function that is not f, it means that f is not analytic but only smooth, i.e. $\mathcal{C}^\infty$, at the point x_0.
- The series does not converge uniformly.

Uniform convergence might be impossible to prove in certain cases, or we might want to makes sense of series expansion "at infinity" — in this case, what we consider are asymptotic expansions, which require weaker conditions to be defined and are more general. In particular Taylor series expansions are one kind of asymptotic expansions. In order to define them, we have first to introduce an additional notion[37] [106]:

[36] See any classical calculus textbook, for instance [105].
[37] We will consider functions defined on an interval $I \subset \mathbb{R}$.

Definition 2.4.8 (Asymptotic sequences)**.** *Let $x_0 \in I$. The sequence of functions $(\varphi_n(x))_n$, $n \in \mathbb{N}$, is called an **asymptotic sequence** as $x \to x_0$ in I if, for every n, $\varphi_n(x)$ is defined and continuous on I and*

$$\varphi_{n+1}(x) \underset{x\to x_0}{=} o\left(\varphi_n(x)\right). \tag{2.4.15}$$

Example 2.4.3. *In particular, we have the following classical examples of asymptotic sequences, where x_0 is finite:*

- $\left((x-x_0)^n\right)_n$, $x \to x_0$.
- $\left(x^{-n}\right)_n$, $x \to +\infty$.
- $\left([g(x)]^n\right)_n$, $x \to x_0$ *and* $g(x_0) = 0$.

Definition 2.4.9 (Asymptotic expansion)**.** *Let $f(x)$ be continuous on $I \subset \mathbb{R}$, and let $(\varphi_n(x))_n$ be an asymptotic sequence as $x \to x_0$ in I. Then the formal series*

$$\sum_{n=0}^{\infty} a_n \varphi_n(x) \tag{2.4.16}$$

*is said to be an **infinite asymptotic expansion** of $f(x)$, as $x \to x_0$, with respect to $(\varphi_n)_n$ if one of the equivalent conditions*

$$f(x) \underset{x\to x_0}{=} \sum_{k=0}^{n} a_k \varphi_k(x) + \mathcal{O}\left(\varphi_{n+1}(x)\right), \quad n \in \mathbb{N} \tag{2.4.17}$$

$$\lim_{x\to x_0} \left[\frac{f(x) - \sum_{k=0}^{n} a_k \varphi_k(x)}{\varphi_n(x)}\right] = 0, \quad n \in \mathbb{N} \tag{2.4.18}$$

is satisifed. If that is the case, we will write

$$f(x) \underset{x\to x_0}{\sim} \sum_{n=0}^{\infty} a_n \varphi_n(x). \tag{2.4.19}$$

Remark 2.4.4. *Of course this definition can be be broadened to the weaker asymptotic power series expansions of order n, which simply means that writing*

$$f(x) \underset{x\to x_0}{\sim} \sum_{k=0}^{n} a_n \varphi_n(x) \tag{2.4.20}$$

is equivalent to satisfying the following two conditions:

- $\forall k \in \{0, \ldots, n-1\}$, $\varphi_{n+1}(x) \underset{x\to x_0}{=} o\left(\varphi_n(x)\right)$.

- $\lim_{x\to x_0}\left[\frac{f(x)-\sum_{k=0}^{n}a_k\varphi_k(x)}{\varphi_n(x)}\right]=0.$

Remark 2.4.5. *As opposed to the analytic function discussion above, the asymptotic series need not converge and is considered as a formal series, so it might be divergent. More precisely, the difference between convergent power series and asymptotic power series can be summarized as follows:*

- *For a convergent power series, we want the remainder $R_n(x)$ to converge to 0 as x is fixed and $n\to+\infty$, i.e., we require* ***uniform convergence****.*
- *For an asymptotic series, we want the remainder $R_n(x)$ to converge to 0 as n is fixed and $x\to x_0$, i.e., we only need* ***simple convergence****.*

Example 2.4.4. *Suppose we want to compute an asymptotic expansion as $x\to+\infty$ of the exponential integral*

$$\mathrm{Ei}(x):=\int_x^\infty \mathrm{d}t\,\frac{e^{-t}}{t}. \tag{2.4.21}$$

We can use iterated integration by parts to find that

$$\begin{aligned}\mathrm{Ei}(x)&=\left[-\frac{e^{-t}}{t}\right]_x^\infty-\int_x^\infty \mathrm{d}e^{-t}\,t^2\\&=\frac{e^{-x}}{x}+\left[-\frac{e^{-t}}{t^2}\right]_x^\infty+2\int_x^\infty \mathrm{d}e^{-t}\,t^3\\&=\cdots\\&=e^{-x}\sum_{k=1}^{n}\frac{(-1)^{k+1}(k-1)!}{x^k}+\underbrace{(-1)^n n!\int_x^\infty \mathrm{d}t\,\frac{e^{-t}}{t^{n+1}}}_{R_n(x)}.\end{aligned}$$

This is clearly an asymptotic expansion. Indeed, one can easily verify that the $\varphi_k(x)=\frac{e^{-x}}{x^n}$ form an asymptotic sequence as $x\to+\infty$. Moreover, we can bound the remainder by

$$|R_n(x)|\le\frac{n!}{x^{n+1}}\int_x^\infty \mathrm{d}t\,e^{-t}=\frac{n!e^{-x}}{x^{n+1}}=\frac{n!}{x}\varphi_n(x) \tag{2.4.22}$$

which means in particular that

$$\lim_{x\to+\infty}\left|\frac{\mathrm{Ei}(x)-\sum_{k=1}^{n}(-1)^{k+1}(k-1)!\varphi_k(x)}{\varphi_n(x)}\right|\le\lim_{x\to+\infty}\frac{n!}{x}=0. \tag{2.4.23}$$

Nevertheless, if we fix x, the asymptotic representation of Ei *diverges as $n\to+\infty$, which means we only get a simple convergence and no analycity at $x\to+\infty$.*

Asymptotic expansions behave nicely under summation, multiplication and also integration [106]:

Property 2.4.1. *Let f and g be two functions, both defined on an interval $I \subset \mathbb{R}$ with an asymptotic expansion around a point $x_0 \in I$, i.e.,*

$$f(x) \sim \sum_{n=0}^{\infty} a_n (x - x_0)^n, \quad g(x) \sim \sum_{n=0}^{\infty} b_n (x - x_0)^n. \tag{2.4.24}$$

Then we have the following:

$$f(x) + g(x) \sim \sum_{n=0}^{\infty} (a_n + b_n)(x - x_0)^n, \tag{2.4.25}$$

$$f(x)g(x) \sim \sum_{n=0}^{\infty} c_n (x - x_0)^n, \quad c_n = \sum_{k=0}^{n} a_k b_{n-k}. \tag{2.4.26}$$

Moreover, if f is integrable, then

$$\int_{x_0}^{x} \mathrm{d}t\, f(t) \sim \sum_{n=0}^{\infty} \frac{a_n}{n+1} (x - x_0)^{n+1}. \tag{2.4.27}$$

Proof. The family $\{\varphi_n(x) = (x - x_0)^n\}_{n \in \mathbb{N}}$ is clearly an asymptotic sequence as $x \to x_0$, so we only need to prove in all three cases that we have either (2.4.17) or (2.4.18).

- For (2.4.25), we have for any given $n \in \mathbb{N}$:

$$\begin{aligned}
&\lim_{x \to x_0} \left[\frac{(f+g)(x) - \sum_{k=0}^{n} (a_k + b_k)(x - x_0)^k}{(x - x_0)^n} \right] \\
&\quad = \lim_{x \to x_0} \left[\frac{f(x) - \sum_{k=0}^{n} a_k (x - x_0)^k + g(x) - \sum_{k=0}^{n} b_k (x - x_0)^k}{(x - x_0)^n} \right] \\
&\quad = \lim_{x \to x_0} \left[\frac{f(x) - \sum_{k=0}^{n} a_k (x - x_0)^k}{(x - x_0)^n} \right] \\
&\qquad + \lim_{x \to x_0} \left[\frac{g(x) - \sum_{k=0}^{n} b_k (x - x_0)^k}{(x - x_0)^n} \right] \\
&\quad = 0
\end{aligned}$$

  using (2.4.24).

- For (2.4.26), using (2.4.17) is easier. For any $n \in \mathbb{N}$:

$$\begin{aligned}
f(x)g(x) &= \left(\sum_{k=0}^{n} a_k (x - x_0)^k + \mathcal{O}\left((x - x_0)^{n+1} \right) \right) \left(\sum_{k=0}^{n} b_k (x - x_0)^k + \mathcal{O}\left((x - x_0)^{n+1} \right) \right) \\
&= \sum_{k=0}^{n} a_k (x - x_0)^k \cdot \sum_{k=0}^{n} b_k (x - x_0)^k
\end{aligned}$$

$$+\underbrace{\sum_{k=0}^{n} a_k (x-x_0)^k \mathcal{O}\left((x-x_0)^{n+1}\right) + \sum_{k=0}^{n} b_k (x-x_0)^k \mathcal{O}\left((x-x_0)^{n+1}\right)}_{\mathcal{O}((x-x_0)^{n+1})}$$
$$+\mathcal{O}\left((x-x_0)^{n+1}\right)$$
$$=\sum_{k=0}^{n} \underbrace{\left(\sum_{p=0}^{k} a_k b_{k-p}\right)}_{c_k} (x-x_0)^k + \mathcal{O}\left((x-x_0)^{n+1}\right).$$

- For (2.4.27), let us bound the error term[38] $\mathcal{R}_n(x)$ of the antiderivative of f for a given $n \in \mathbb{N}$:

$$\begin{aligned}
|\mathcal{R}_n(x)| &= \left| \int_{x_0}^{x} \mathrm{d}t\, f(t) - \sum_{k=0}^{n} \frac{a_k}{k+1} (x-x_0)^{k+1} \right| \\
&= \left| \int_{x_0}^{x} \mathrm{d}t \left(f(t) - \sum_{k=0}^{n} a_k (t-x_0)^k \right) \right| \\
&\leq \int_{x_0}^{x} \mathrm{d}t \left| f(t) - \sum_{k=0}^{n} a_k (t-x_0)^k \right| \\
&= \int_{x_0}^{x} \mathrm{d}t\, |R_n(t)|\,.
\end{aligned}$$

  We know from Definition 2.4.9, and more precisely from (2.4.17) that $R_n(t) = \mathcal{O}\left((t-x_0)^{n+1}\right)$ which means in particular from Definition 2.4.1 that there exist some $M > 0$ and $\epsilon > 0$ such that whenever $|t - x_0| \leq \epsilon$, we have

$$|R_n(x)| \leq M |t-x_0|^{n+1}.$$

  In particular, this implies

$$|\mathcal{R}_n(x)| \leq M \int_{x}^{x_0} \mathrm{d}t\, |t-x_0|^{n+1} = \frac{M}{n+2} |x-x_0|^{n+2}$$

  and therefore

$$\lim_{x \to x_0} \left[\frac{\int_{x_0}^{x} \mathrm{d}t\, f(t) - \sum_{n=0}^{\infty} \frac{a_n}{n+1} (x-x_0)^{n+1}}{(x-x_0)^n} \right] = 0.$$

Therefore, all properties are proved. □

[38] We will note with $\mathcal{R}_n(x)$ the error term for the antiderivative and with $R_n(x)$ the error term for f itself.

2.4.2. Integration and asymptotics

When one considers expressions involving for instance Mellin convolutions, we get parametric integrals of the form

$$I(x) = \int_0^1 \mathrm{d}z\, f(x,z), \quad 0 \leq x \leq 1, \tag{2.4.28}$$

and one natural question is under which conditions we can compute asymptotic expansions of such integrals when[39] $x \to x_0 \in [0;1]$. There is a general theorem [106] that gives us a simple method to compute theses asymptotics under certain conditions.

Theorem 2.4.4 (Asymptotics of parametric integrals)**.** *Let us consider the parametric integral*

$$F(x) = \int_{z_0}^{z_1} \mathrm{d}z\, f(x,z) \tag{2.4.29}$$

defined on an interval $I \subset \mathbb{R}$, i.e., $x \in I$. We suppose that as $x \to x_0 \in I$ the function f has the following asymptotic expansion up to order n:

$$f(x,z) \sim \sum_{k=1}^{n} a_k(z)\varphi_k(x) \tag{2.4.30}$$

where $\{\varphi_k\}_{k\in\{1,\dots,n\}}$ is an asymptotic sequence. Furthermore, we suppose this asymptotic expansion holds uniformly in z for $z \in [z_0; z_1]$. Let us suppose that the integrals

$$A_k = \int_{z_0}^{z_1} \mathrm{d}z\, a_k(z), \quad k \in \{1,\dots,n\} \tag{2.4.31}$$

exist. Then, as $x \to x_0$ we have

$$F(x) \sim \sum_{k=1}^{n} A_k \varphi_k(x) \tag{2.4.32}$$

that is to say the asymptotic expansion of f in x can be integrated term-by-term with respect to the parameter z.

Proof. From the Definition 2.4.9 of an asymptotic expansion we can write

$$F(x) = \int_{z_0}^{z_1} \mathrm{d}z \left(\sum_{k=1}^{n} a_k(z)\varphi_k(x) + R_n(x,z) \right)$$

[39] Most of the time we will consider $x_0 = 1$.

with $R_n(x,z) = \mathcal{O}_z\left(\varphi_{n+1}(x)\right)$, where we use the notation $\mathcal{O}_z$ to mean the function depends explicitely on z. By hypothesis, we suppose that the asymptotic expansion is uniform in z which means that there exists a certain $A > 0$ such that

$$\forall z \in [z_0; z_1],\ |R_n(x,z)| \leq A\,|\varphi_{n+1}(x)| = \mathcal{O}\left(\varphi_{n+1}(x)\right).$$

In particular, this means that we have

$$\int_{z_0}^{z_1} \mathrm{d}z\, R_n(x,z) \underset{x\to x_0}{=} \mathcal{O}\left(\varphi_{n+1}(x)\right)$$

and hence

$$F(x) \underset{x\to x_0}{=} \sum_{k=1}^{n} A_n \varphi_n(x) + \mathcal{O}\left(\varphi_{n+1}(x)\right)$$

which concludes the proof. □

We note here that the main hypothesis that allows us to perform so intuitively the "term-by-term asymptotics" is the uniformity hypothesis. More precisely, we can turn the error term from $\mathcal{O}_z\left(\varphi_{n+1}(x)\right)$ to $\mathcal{O}\left(\varphi_{n+1}(x)\right)$, i.e., we can bound it by a z-independent term. However, we can go even futher if we look at the remainder. More precisely, we look at the Taylor expansion around the point x_0:

$$f(x,z) \underset{x\to x_0}{=} \underbrace{\sum_{k=0}^{n} \frac{(x-x_0)^k}{k!} \partial_x^{(n)}\, f(x,z)\big|_{x=x_0}}_{P_n(x,z)} + \underbrace{\sum_{k=n+1}^{\infty} \frac{(x-x_0)^k}{k!} \partial_x^{(n)}\, f(x,z)\big|_{x=x_0}}_{R_n(x,z)} \tag{2.4.33}$$

where $P_n(x,z)$ is the Taylor polynomial and $R_n(x,z)$ the remainder in its explicit form. More generally, if f is defined only when $x \leq x_0$ or $x \geq x_0$, i.e., it has a factor $f(x) \propto (x_0 - x)^\alpha$ where $\alpha \in (-1,1)$ and is a non-integer number[40], then we write

$$\begin{aligned} f(x,z) \underset{x\to x_0}{=} & \underbrace{\sum_{k=0}^{n} \frac{(x_0-x)^\alpha (x-x_0)^k}{k!} \partial_x^{(n)} \left[\frac{f(x,z)}{(x_0-x)^\alpha}\right]\Bigg|_{x=x_0}}_{P_n(x,z)} \\ & + \underbrace{\sum_{k=n+1}^{\infty} \frac{(x_0-x)^\alpha (x-x_0)^k}{k!} \partial_x^{(n)} \left[\frac{f(x,z)}{(x_0-x)^\alpha}\right]\Bigg|_{x=x_0}}_{R_n(x,z)}. \end{aligned} \tag{2.4.34}$$

While cumbersome at first sight, this expression gives us a direct method to deal with the remainder: if we manage to bound efficiently enough the summand of the

[40] Without loss of generality we will consider here the case $x < x_0$.

remainder, i.e., bound it by an expression where the dependence in x and z is split, where we can perform the sum from $k = n + 1$ to $+\infty$, and where the obtained result converges to 0 as $x \to x_0$ (asymptotic expansion) or, even better, there exist a range of x in a neighbourhood of $x = x_0$ where it converges to 0 as $n \to +\infty$ (convergent expansion), then we are in the position to apply Theorem 2.4.4 and to derive the desired expansion. In the spirit of the ratio test, we prove a theorem that gives us a certain criterion helping us to automatically conclude about the convergence of certain parametric integrals.

Theorem 2.4.5. *Let $f : (0;1] \times [0;1] \to \mathbb{R}$ be a smooth function with a series expansion around $x = 1$ for all $n \in \mathbb{N}$:*

$$f(x,z) = \sum_{k=0}^{n} A_k(z)(1-x)^k + R_n(x,z), \tag{2.4.35}$$

where

$$A_n(z) := \frac{(-1)^n}{n!}\partial_x^n \, f(x,z)\big|_{x=1}, \quad R_n(x,z) = \sum_{k=n+1}^{\infty} A_k(z)(1-x)^k. \tag{2.4.36}$$

Assume there exists $C \in \mathbb{R}_{\geq 1}$ and $g(z)$ integrable on $[0;1]$ such that for all $n \in \mathbb{N}$ the following holds:

$$|A_n(z)| \leq g(z)C^n. \tag{2.4.37}$$

Assume also that for all $n \in \mathbb{N}$, $\int_0^1 \mathrm{d}z\, A_n(z)$ exists. Then for every $n \in \mathbb{N}$:

$$F(x) := \int_0^1 \mathrm{d}z\, f(x,z) = \underbrace{\sum_{k=0}^{n}(1-x)^k \int_0^1 \mathrm{d}z\, A_k(z)}_{\mathcal{P}_n(x)} + \mathcal{R}_n(x) \tag{2.4.38}$$

with

$$\mathcal{R}_n(x) = \int_0^1 \mathrm{d}z\, R_n(x,z), \tag{2.4.39}$$

where we have

$$\exists r \in (0;1),\ \forall x \in (r;1),\ \mathcal{R}_n(x) \underset{n\to+\infty}{\longrightarrow} 0. \tag{2.4.40}$$

Proof. Starting with

$$R_n(x,z) = \sum_{k=n+1}^{\infty} A_k(z)(1-x)^k \tag{2.4.41}$$

and using the bounding from our hypothesis (2.4.37) we get

$$|R_n(x,z)| \leq \sum_{k=n+1}^{\infty} \left|A_k(z)(1-x)^k\right| \leq \sum_{k=n+1}^{\infty} g(z)\,|1-x|^k\, C^k. \tag{2.4.42}$$

For $0 < x < 1$ we have

$$C\,|1-x| < 1 \Leftrightarrow |1-x| < \frac{1}{C} \Longleftrightarrow 1 - \frac{1}{C} < x < 1. \tag{2.4.43}$$

Thus for $r := 1 - \frac{1}{C}$ with $r \in (0;1)$ we have $C|1-x| < 1$ and thus

$$|R_n(x,z)| \leq g(z)\frac{C^{1+n}\,|1-x|^{1+n}}{1-C\,|1-x|} = g(z)C\,|1-x| \cdot \frac{(C\,|1-x|)^n}{1-C\,|1-x|}, \tag{2.4.44}$$

and therefore

$$\forall x \in (r;1)\,,\; R_n(x,z) \underset{n\to+\infty}{\longrightarrow} 0. \tag{2.4.45}$$

This means that the series expansion (2.4.35) of $f(x,z)$ converges uniformly to $f(x,z)$ in $(r;1)$. Integrating over z the terms $A_k(z)(1-x)^k$ in (2.4.35), where the dependence in z and x is split, naturally yields the corresponding term of order k of the expansion of F. More precisely, for $x \in (r;1)$ we can write:

$$\begin{aligned} F(x) &= \int_0^1 \mathrm{d}z\, f(x,z) \\ &= \int_0^1 \mathrm{d}z\, \left(\sum_{k=0}^{n} A_k(z)(1-x)^k + R_n(x,z)\right) \\ &= \sum_{k=0}^{n}(1-x)^k \underbrace{\int_0^1 \mathrm{d}z\, A_k(z)}_{=:\bar{A}_k} + \int_0^1 \mathrm{d}z\, R_n(x,z), \end{aligned} \tag{2.4.46}$$

which allows us to identify the remainder $\mathcal{R}_n$ of F as the integral of the remainder R_n of f as long as we manage to show that there exists an $r < 1$ such that

$$\forall x \in (r;1),\, \mathcal{R}_n(x) = \int_0^1 \mathrm{d}z\, R_n(x,z) \underset{n\to+\infty}{\overset{?}{\longrightarrow}} 0.$$

Using the bounding from (2.4.44), we get

$$\begin{aligned} |\mathcal{R}_n(x)| &\leq \int_0^1 \mathrm{d}z\; |R_n(x,z)| \\ &\leq \frac{(C\,|1-x|)^{n+1}}{1-C\,|1-x|} \underbrace{\int_0^1 \mathrm{d}z\, g(z)}_{=:G<\infty} \end{aligned} \tag{2.4.47}$$

and therefore, with (2.4.43) we obtain

$$\forall x \in (r;1),\, \mathcal{R}_n(x) \underset{n\to+\infty}{\longrightarrow} 0. \tag{2.4.48}$$

As a consequence $\mathcal{R}_n$ in (2.4.46) is the remainder of the asymptotic expansion, which concludes the proof. □

Remark 2.4.6. *If the function $f(x,z)$ is such that*

$$f(x,z) \underset{x\to 1}{\sim} A(z)(1-x)^{a+p}, a \in (-1;1), a \neq 0, p \in \mathbb{N}$$

it suffices to define

$$\bar{f}(x,z) := \frac{f(x,z)}{(1-x)^{a+p}}$$

and write

$$F(x) = \int_0^1 \mathrm{d}z\, f(x,z) = (1-x)^{a+p} \underbrace{\int_0^1 \mathrm{d}z\, \bar{f}(x,z)}_{=:\bar{F}(x)},$$

then apply Theorem 2.4.5 to $\bar{F}(x)$, and the Taylor expansion of $F(x)$ is obtained simply by multiplying that of $\bar{F}(x)$ by $(1-x)^{a+p}$.

The theorem above gives us a clear criterion to conclude immediately about the validity of the asymptotic expansion and analycity in a certain neighbourhood of $x = 1$ — if we manage to bound derivatives of our integrand $f(x,z)$ as in (2.4.37). Then we know the asymptotics of the integral $F(x) = \int_0^1 \mathrm{d}z\, f(x,z)$ is obtained simply by term-by-term integration of the expansion of $f(x,z)$. A natural question that arises concerns the bounding (2.4.37) itself, i.e., how we can obtain it. Essentially, the bound is obtained looking at the n^{th} partial derivative with respect to x of the integrand $f(x,z)$, and we can follow the method described below to try to bound the derivative.
In order to activate Theorem 2.4.5 and to bound the terms of the series expansion, at least for certain classes of functions[41], we rely on the following two lemmas:

Lemma 2.4.1. *Let $k \in \mathbb{N}$. Then there exists an a such that*

$$\forall n \in \mathbb{N},\ n^k \leq a^n. \tag{2.4.49}$$

Proof. Let us prove the lemma by induction. We start by solving $n \leq 2^n$ for $n \in \mathbb{N}$. We have that

$$n \leq 2^n \Leftrightarrow \log(n) \leq \log(2)n$$

because $n \geq 0$. Now we just can study the function $f : n \mapsto \log(2)n - \log(n)$, whose derivative $\log(2) - \frac{1}{n}$ vanishes at $n_m := \frac{1}{\log(2)} \approx 1,44$, and is negative on $(0; \frac{1}{\log(2)})$ and positive on $(\frac{1}{\log(2)}; +\infty)$. We have $f(n) \underset{n\to 0}{\longrightarrow} +\infty$ and by plugging n_m in f we get

$$f(n_m) = \log(n_m) - \log(2)n = 1 + \log\log(2) \approx 0.63,$$

[41] See Definition 2.4.10 below.

which means that

$$\forall n \in \mathbb{N},\ f(n) \geq 0 \Leftrightarrow \forall n \in \mathbb{N},\ n \leq 2^n.$$

This proves the initial case[42] $k = 1$. Now suppose there exists some b such that

$$\forall n \in \mathbb{N},\ n^k \leq b^n.$$

We also know from the initialisation that

$$\forall n \in \mathbb{N},\ n \leq 2^n$$

so because all the quantities are positive we can multiply one equality by the other and get

$$\forall n \in \mathbb{N},\ n^{k+1} \leq (2b)^n,$$

which tells us that by taking for instance $a = 2^k$ we have

$$\forall n \in \mathbb{N},\ n^k \leq a^n,$$

which in turn concludes the proof. □

Lemma 2.4.2. *Let $n \in \mathbb{N}$ and define*

$$B_n := P(n) \sum_{1 \leq k_1, \ldots, k_p \leq n} \alpha_1^{\ell_1} \cdots \alpha_q^{\ell_q} \tag{2.4.50}$$

where $P(n) \in \mathbb{K}[n]$, the $\alpha_i \in \mathbb{R}_+$ are all distinct, ℓ_i are linear combinations of the k_i and n of the form

$$\forall j \in \{1, \ldots, q\},\ \ell_j = \gamma_j^0 n + \sum_{i=1}^{p} \gamma_j^i k_i \text{ with } \gamma_j^i \in \mathbb{R}. \tag{2.4.51}$$

Then there exists a constant $C \in \mathbb{R}_{\geq 1}$ such that

$$\forall n \geq 0,\ B_n \leq C^n. \tag{2.4.52}$$

Proof. First we start by rewriting B_n as

$$B_n = P(n)\beta_0^n \prod_{i=1}^{p} \left(\sum_{k_i=0}^{n} \beta_i^{k_i} \right). \tag{2.4.53}$$

Indeed, because the ℓ_i are linear combinations in the k_i, n, i.e.,

$$\forall j,\ \ell_j = \gamma_j^0 n + \sum_{i=1}^{p} \gamma_j^i k_i \quad \gamma_j^i \in \mathbb{R} \tag{2.4.54}$$

[42] The case $k = 0$ is solved easily because any $a > 1$ solves for all $n \in \mathbb{N}$ the equation $1 \leq a^n$.

we can write the β_i as

$$\beta_i = \prod_{j=1}^{q} \alpha_j^{\gamma_1^i} \cdots \alpha_q^{\gamma_q^i} \in \mathbb{R}_+. \tag{2.4.55}$$

Using geometric sums, we get:

$$B_n = P(n)\beta_0^n \left(\prod_{i=1}^{m_p} \frac{\beta_i^{n+1} - 1}{\beta_i - 1} \right) (n+1)^{p-m_p} \text{ with } \beta_i \neq 1,\ 0 \leq m_p \leq p. \tag{2.4.56}$$

In the product, i does not run from 1 to p and we get the extra $(n+1)^{p-m_p}$ factor because we have to deal with the situation that while we recombine the α_i into β_i, we can obtain cases where $\beta_i = 1$, and therefore

$$\sum_{k_i=0}^{n} 1 = (n+1);$$

so m_p counts the number of such cases where $\beta_i = 1$, i.e.,

$$m_p := \sum_{i=1}^{p} \delta_{1,\beta_i}.$$

Let us now bound this expression of B_n. First, using the Lemma 2.4.1 above as well as the fact that for all $p \geq 0$ and $n \geq 0$ we have $p(n+1) \leq 2pn$, we bound

$$\forall n \in \mathbb{N},\ (n+1)^{p-m_p} \leq 2^{(p-m_p)(n+1)} \leq 2^{2(p-m_p)n}. \tag{2.4.57}$$

Now, for all $\beta_i \neq 1$ we have

$$\left| \frac{\beta_i^{n+1} - 1}{\beta_i - 1} \right| \leq |\beta_i^n| \left| \frac{\beta_i}{\beta_i - 1} \right| + \left| \frac{1}{\beta_i - 1} \right|$$

$$C_i := \max\left(|\beta_i|, \left| \frac{\beta_i}{\beta_i - 1} \right|, \left| \frac{1}{\beta_i - 1} \right| \right) + 1 \to \leq 2C_i^{2n}$$

$$D_i := \max(2, C_i)^3 \to \leq D_i^n.$$

If we write the polynomial $P(n) = \sum_{u=0}^{v} \delta_u n^u$, using Lemma 2.4.1, we can bound it by

$$P(n) \leq \Delta A^n \text{ with } \Delta := \max_{1 \leq u \leq v} |\delta_u|, A \in \mathbb{R}_+.$$

Combining everything together we get

$$\forall n \in \mathbb{N},\ B_n \leq \Delta A^n \beta_0^n \left(\prod_{i=1}^{m_p} D_i^n \right) 2^{2(p-m_p)n} = \Delta D^n$$

where we defined

$$D := A\beta_0\, 4^{p-m_p} \prod_{i=1}^{m_p} D_i.$$

Finally, because $\bar{\Delta} := \max\left(\Delta, \frac{1}{\Delta}\right) \geq 1$, we have

$$\Delta D^n \leq \bar{\Delta} D^n \leq \left(\bar{\Delta} D\right)^n \leq C^n$$

with $C := \max(1, \bar{\Delta} D)$. Thus

$$\forall n \in \mathbb{N},\ B_n \leq C^n$$

which concludes the proof. □

Due to the lemmas above, it is actually enough to bound the n^{th} term of the series expansion of $f(x, z)$ in x in the following way

$$|A_n(z)| \leq P(n) g(z) \sum_{1 \leq k_1, \ldots, k_p \leq n} \alpha_1^{\ell_1} \cdots \alpha_q^{\ell_q}, \tag{2.4.58}$$

where $P(n) \in \mathbb{K}[n]$ and the sum over the α_i is of the same kind as in Lemma 2.4.2. By this same lemma, we know there exists a $C \in \mathbb{R}_{\geq 1}$ such that (2.4.37) is satisfied.

From now on we will be considering a particular subclass of algebraic functions that is relevant to us due to the nature of the alphabet (2.1.43) we are considering:

Definition 2.4.10 (Root-valued algebraic function)**.** *We say that f is a **root-valued algebraic function** in variables $x_1, \ldots, x_n$ if there exists a rational function $r(x_1, \ldots, x_n)$ such that*

$$f(x_1, \ldots, x_n)^2 = r(x_1, \ldots, x_n).$$

More precisely, we will be able to utilize the above bounds and obtain the following method.

Input: A root-valued algebraic function $f(x, z)$ defined and integrable on $(0; 1) \times (0; 1)$.
Output: The **series expansion** of $F(x) = \int_0^1 \mathrm{d}z\, f(x, z)$ around $x = 1$ or **Failed**.

Method 1: Bounding series expansion terms

1. First, we factor as much as possible the arising expressions in $f(x,z)$ and obtain a list of all its irreducible factors up to constants. If needed, we take analytic continuations of roots of polynomials, such that we get at least one non-constant polynomial term in the numerator. If no such term is present, it suffices to take a root term $\sqrt{p(x,z)}$ where $p(x,z)$ is a polynomial in x,z and replace it by $\frac{p(x,z)}{\sqrt{p(x,z)}}$; if no root term is present, we can simply multiply and divide the whole expression by $1+x$, $1+x$ artificially becoming a numerator. The meaning of this manipulation will become clear in the next steps.

Example 2.4.5. *We take an explicit example to illustrate the method, more precisely we consider the sum representation of the following GCPL function:*

$$\mathcal{H}^*_{\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{1}{\lambda-\frac{1}{4}}\sqrt{\frac{\lambda}{1-\lambda}};\emptyset\right]\right)}(x) = 4\int_x^1 \mathrm{d}y\, \frac{\sqrt{\frac{y}{x(x-y)(y-1)}}}{4y-1}$$
$$y=\frac{x}{1+(x-1)z} \to = -4\int_0^1 \mathrm{d}z\, \frac{\sqrt{-\frac{x}{(z-1)z((x-1)z+1)}}}{x(z-4)-z+1}. \tag{2.4.59}$$

Therefore, we consider

$$F(x)=\int_0^1 \mathrm{d}z\, f(x,z) \text{ with } f(x,z)=\frac{\sqrt{-\frac{x}{(z-1)z((x-1)z+1)}}}{x(z-4)-z+1}. \tag{2.4.60}$$

In our case the irreducible factors appearing in $f(x,z)$ in (2.4.60) are obtained by properly factorizing the expression

$$\begin{aligned} f(x,z) &= \frac{\sqrt{-\frac{x}{(z-1)z((x-1)z+1)}}}{x(z-4)-z+1} \\ &= \sqrt{x}\cdot\frac{1}{\sqrt{1-z}}\cdot\frac{1}{\sqrt{z}}\cdot\frac{1}{\sqrt{z(x-1)+1}}\cdot\frac{1}{x(z-4)-z+1} \end{aligned}$$

As we see, there are no polynomial terms in the numerator. So to deal with this we can simply replace $x\to\frac{x}{\sqrt{x}}$ which means we write

$$f(x,z)=x\cdot\frac{1}{\sqrt{x}}\cdot\frac{1}{\sqrt{1-z}}\cdot\frac{1}{\sqrt{z}}\cdot\frac{1}{\sqrt{z(x-1)+1}}\cdot\frac{1}{x(z-4)-z+1}. \tag{2.4.61}$$

Therefore, the list of irreducible factors is

$$\left\{x, \frac{1}{\sqrt{x}}, \frac{1}{\sqrt{1-z}}, \frac{1}{\sqrt{z}}, \frac{1}{\sqrt{xz-z+1}}, \frac{1}{xz-4x-z+1}\right\}. \tag{2.4.62}$$

2. We factor out all the x-constants, in particular terms that involve only z, and then compute the n^{th}-derivative of the remaining terms. If the expression is a polynomial, we keep the derivative symbolic, i.e., for a polynomial $p(x,z)$ we just write

$$\partial_x^n p(x,z).$$

Remark 2.4.7. *Sometimes Mathematica does not return a closed form, e.g., in terms of binomial coefficients and factorial in n, but a linear recurrence in n. In these rare cases, we use* `SolveREusingSolveDE` *from* `HarmonicSums` *and the function* `FindLinearCombination` *from* `Sigma` *to obtain a closed form of the n^{th}-derivative. If we cannot obtain a closed form for one of the derivatives, we stop and the method* ***fails****.*

Example 2.4.5 (continued)**.** *In our case, the x-independent factors in (2.4.62) are $\frac{1}{\sqrt{z}}, \frac{1}{\sqrt{1-z}}$, and for the rest the derivatives are:*

$$x \longrightarrow \partial_x^n(x)$$

,

$$\begin{aligned}\frac{1}{\sqrt{x}} &\longrightarrow \left(-\frac{1}{2}\right)^{\underline{n}} x^{-n-\frac{1}{2}} \\ &= \left(-\frac{1}{4}\right)^n n! \binom{2n}{n} x^{-n-\frac{1}{2}}\end{aligned}$$

$$\begin{aligned}\frac{1}{\sqrt{1+(x-1)z}} &\longrightarrow \left(-\frac{1}{2}\right)^{\underline{n}} z^n (xz-z+1)^{-n-\frac{1}{2}} \\ &= \left(-\frac{1}{4}\right)^n n! \binom{2n}{n} z^n (xz-z+1)^{-n-\frac{1}{2}}\end{aligned}$$

$$\frac{1}{\sqrt{x(z-4)-z+1}} \longrightarrow (-1)^{\underline{n}} (z-4)^n (xz-4x-z+1)^{-n-1}$$

$$= (-1)^n n!(z-4)^n(xz-4x-z+1)^{-n-1}$$

where to simplify the expression we used the fact that the falling factorial $(a)^{\underline{n}}$ *can be represented with Gamma functions*

$$(a)^{\underline{n}} = \frac{\Gamma(a+1)}{\Gamma(a-n+1)}.$$

3. Since we want to compute the n^{th} derivative of $f(x,z)$ and we now know the n^{th} derivative of the individual constituents, we simply use the formula for the n^{th} derivative of a product of p functions $f_1(x,z)\cdots f_p(x,z)$, i.e.,

$$\partial_x^n\left[f_1(x,z)\cdots f_p(x,z)\right] = \sum_{1\le k_1,\ldots,k_{p-1}\le n} \binom{n}{k_1,\ldots,k_{p-1},n-k_1-\cdots-k_{p-1}} \times \partial_x^{k_1} f_1(x,z)\cdot\cdots\cdot\partial_x^{n-k_1-\cdots-k_{p-1}} f_p(x,z) \tag{2.4.63}$$

with the multinomial coefficient being defined as

$$\binom{n}{k_1,k_2,\ldots,k_{p-1},k_p} := \frac{n!}{k_1!k_2!\cdots k_{p-1}!k_p!} \tag{2.4.64}$$

and where we took care that the most complicated derivative $\partial_x^{n-k_1-k_2\cdots-k_{p-1}}$ acts on the polynomial term, i.e., that $f_p(x,z)$ is the polynomial term.

Example 2.4.5 (continued)**.** *Plugging the derivatives above in* (2.4.63)*, we obtain*

$$\sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n}\frac{1}{(n-k_1-k_2-k_3)!}(-1)^{k_1+k_2+k_3}4^{-k_2-k_3}x^{-k_3-\frac{1}{2}}(x(z-4)-z+1)^{-k_1-1}$$
$$(z-4)^{k_1}z^{k_2}((x-1)z+1)^{-k_2-\frac{1}{2}}\binom{2k_2}{k_2}\binom{2k_3}{k_3}\partial_x^{n-k_1-k_2-k_3}(x).$$

4. Because we are interested in bounding the n^{th} Taylor term rather than the derivative itself, i.e.,

$$\frac{(1-x)^n}{n!}\,\partial_x^n f(x,z)|_{x=1}\,,$$

we can add back all the x-independent terms and the $\frac{(1-x)^n}{n!}$ contribution. Afterwards we send $x\to 1$.

Example 2.4.5 (continued). *We obtain*

$$\begin{aligned}\frac{(1-x)^n}{n!}\,\partial_x^n f(x,z)\big|_{x=1} =& \frac{(1-x)^n}{\sqrt{1-z}\sqrt{z}} \\ &\times \sum_{0\le k_1,k_2,k_3\le n} \frac{1}{(-k_1-k_2-k_3+n)!}(-3)^{-k_1-1} \\ &\times (-1)^{k_1+k_2+k_3} \\ &\times 4^{-k_2-k_3}\binom{2k_2}{k_2}\binom{2k_3}{k_3}(z-4)^{k_1} \\ &\times z^{k_2}\,\partial_x^{n-k_1-k_2-k_3}(x)\Big|_{x=1}. \end{aligned} \tag{2.4.65}$$

5. Now that we wrote the n^{th} Taylor coefficient in this form, all what remains to do is to bound the individual coefficients in such a way that we achieve the form appearing on the right hand side of (2.4.58). There exist several classical boundings that we can use:

   - We can bound polynomials $p(x,z)$ and their derivatives by a constant A because we consider those on the interval $[0;1]\times[0;1]$. We now realise why two steps ago we introduced a non-constant polynomial in the numerator, and why this one took the most complicated derivative in the product formula (2.4.63) — indeed, we do not need to care about the derivatives, no matter how complicated they look, as we will anyway bound them by a constant.
   - For $n\in\mathbb{N}$ due to

     $$\frac{\binom{2n}{n}}{\binom{2(n-1)}{n-1}} = 4\left(1-\frac{1}{2n}\right) \Rightarrow \binom{2n}{n} = 4^n\prod_{k=1}^{n}\left(1-\frac{1}{2k}\right),$$

     we have the bound

     $$\binom{2n}{n} \le 4^n. \tag{2.4.66}$$

   - For $k_i, p\in\mathbb{N}$, we can bound

     $$(k_i+p)! \le k_i!(n+p)^p.$$

   - If $p(n,k_1,\dots,k_{p-1})$ is a polynomial such that for all $n,k_1,\dots,k_p$ we

have $p(n, k_1, \ldots, k_{p-1}) \leq 0$ then we can bound

$$z^{p(n,k_1,\ldots,k_{p-1})} \leq 1 \tag{2.4.67}$$

because $z \in (0; 1)$.

- Suppose we have an expression of the form

$$\frac{1}{p(n, k_1, \ldots, k_{p-1})}, \quad p \in \mathbb{Z}[X_1, \ldots, X_p].$$

By construction of the full derivative expression, which is actually non-divergent, it means that we never hit roots or they are canceled by a coefficient in the numerator — in any case, we can safely remove the denominator as a coarse bound is enough anyway in our case. Thus, we get the bound

$$\frac{1}{p(n, k_1, \ldots, k_{p-1})} \leq 1. \tag{2.4.68}$$

- If we have inverse factorials, i.e., expressions of the form

$$\frac{1}{p!},$$

then we simply can bound them by 1. We note that we have sometimes expressions of the form $(n-k_1-\ldots-k_{p-1})!$ in the denominator, due to the appearance of the multinomial coefficient, but the expression always stay positive so its inverse is always less or equal to[a] 1.

- If we have an expression of the form

$$p(z)^{q(n,k_1,\ldots,k_{p-1})} \text{ with } q \in \mathbb{Q}[X_1, \ldots, X_p] \tag{2.4.69}$$

we can deal with it in the following way: if we have to consider for all values of $n, k_1, \ldots, k_{p-1}$ that $q(n, k_1, \ldots, k_{p-1}) \geq 0$, then a bound is given by

$$\max_{z\in[0;1]} |p(z)|^{q(n,k_1,\ldots,k_{p-1})}.$$

Otherwise, we choose $p_{\min}, p_{\max}$ respectively as the minimal and maximal values of $|p(z)|$ on $[0; 1]$, and $q_{\min}, q_{\max}$ respectively the minimal

and maximal values of q on $\{0,\dots,n\}^{p-1}$. Then for a given fixed $n \in \mathbb{N}, n \geq 0$, we can bound (2.4.69) by

$$\max\left(p_{\max}^{q_{\max}}, p_{\min}^{q_{\max}}, p_{\min}^{q_{\min}}\right). \tag{2.4.70}$$

6. We obtained an upper bound which is of the form (2.4.58). Thus we can apply Lemma 2.4.2 to get a bound C with Lemma 2.4.52. This finally enables us to apply Theorem 2.4.5 to get a **series expansion of the integral around** $x = 1$. Otherwise our method **fails to proceed**.

[a] We however note that even if we had expressions of the form $\frac{1}{\Gamma(p)}$ with $p \in \mathbb{Z}_-$, those would be poles of the Gamma function and therefore their inverse is 0, so the condition $\frac{1}{\Gamma(p)} \leq 1$ would still hold.

Example 2.4.5 (continued)**.** *In the case of our example, by applying the bounding rules we arrive at the following expression*

$$\frac{A|1-x|^n}{\sqrt{1-z}\sqrt{z}} \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n}\sum_{k_4=0}^{n} 2^{2k_1-k_4-1}3^{-k_1-1} \tag{2.4.71}$$

which fits the expression (2.4.58) *if we set* $\alpha_1 = 2, \alpha_2 = 3, \ell_1 = 2k_1 - k_4 - 1, \ell_2 = -k_1 - 1, g(z) = \frac{A}{\sqrt{1-z}\sqrt{z}}$*, and therefore* $F(x) = \int_0^1 \mathrm{d}z\, f(x,z)$ *has an analytic series expansion around* $x = 1$ *obtained by integrating term-by-term the expansion of* $f(x,z)$ *around* $x = 1$*:*

$$\begin{aligned}\mathcal{H}^*\left(\left[\tfrac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\tfrac{1}{\lambda-\frac{1}{4}}\sqrt{\tfrac{\lambda}{1-\lambda}};\emptyset\right]\right)(x) \underset{x\to 1}{=} &\frac{4\pi}{3} - \frac{11}{9}\pi(x-1) + \frac{61}{48}\pi(x-1)^2 \\ &- \frac{7223\pi(x-1)^3}{5184} + \frac{1571137\pi(x-1)^4}{995328} \\ &- \frac{7291793\pi(x-1)^5}{3981312} + O\left((x-1)^6\right)\end{aligned}$$

We can also see Theorem 2.4.5 as the initialisation of a recursion, allowing us to get asymptotic expansions of nested integrals, or more precisely of nested Mellin convolutions, if we manage to get an expansion for its constituent convolutions. We make this more precise and describe how we turn this into an explicit method.

Input: A GCPL function $\mathcal{H}^*_{\underline{W}}(x)$ where $\underline{W}$ is a generalized word such that none of its constituents is in the $+$-prescription setting.
Output: An **asymptotic expansion** of it at $x = 1$ up to order n.

Method 2: Checking validity of asymptotics expansion of GCPL

1. First, using Property 2.3.3, we rewrite $\mathcal{H}^*_{\underline{W}}(x)$ as $\mathcal{H}^*_{\underline{V}}(x)$, where $\underline{V}$ a pure generalized word.

2. We find all the fundamental generalized words $\underline{V_i}$ contained in $\underline{V}$ of the form
$$\underline{V_i} := ([f_i(\lambda); \emptyset], [g_i(\lambda); \emptyset])$$
where f_i, g_i are letters. Those expressions correspond to standard Mellin convolutions from which are built the nested convolutions.

3. Apply Method 1 from page 96 — based on Theorem 2.4.5 and Lemma 2.4.2 — on all of them. If at least one of them fails, **stop and return fail**. Otherwise, we get the asymptotic expansion of $\mathcal{H}^*_{\underline{V_i}}(x)$.

4. There exist two possible cases at this point for each of the $\underline{V_i}$ we computed.
   - **Case 1** : We consider without loss of generality[a] convolutions of the form
   $$J(x) = h(x) \star \mathcal{H}^*_{\underline{V_i}}(x)$$
   where $h(x)$ is a letter that is integrable at $x = 1$, i.e., it takes the form
   $$h(x) \underset{x\to 1}{\sim} \beta(1-x)^\gamma, \quad \beta \in \mathbb{R}^\star, \gamma > -1. \tag{2.4.72}$$
   In the previous step, because Theorem 2.4.5 was valid for all of the $\underline{V_i}$, in particular for all $n \in \mathbb{N}$, we obtained $\alpha_k \in \mathbb{R}$, $k \leq n$ such that
   $$\mathcal{H}^*_{\underline{V_i}}(x) = \underbrace{\sum_{k=0}^{n} \alpha_k (1-x)^{k+a}}_{P_n(x)} + R_n(x), \quad R_n(x) = \mathcal{O}\left((1-x)^{a+n+1}\right) \tag{2.4.73}$$
   where $a \in (-1; 1)$. We can split
   $$J(x) = h(x)\star P_n(x) + h(x)\star R_n(x) = \sum_{k=0}^{n} \alpha_k h(x) \star (1-x)^k + h(x) \star R_n(x)$$
   because the $P_n(x)$ and $R_n(x)$ parts are by definition regular enough at $x = 1$ and do not induce divergent patterns at $x = 1$. Concerning

the $P_n(x)$ part, we can write for each of the terms:

$$\begin{aligned}
\left|h(x) \star (1-x)^{k+a}\right| &= \left|\int_x^1 \mathrm{d}y\, \frac{h(y)}{y}\left(1-\frac{x}{y}\right)^{k+a}\right| \\
y = \frac{x}{1+(x-1)z} \to &= |(1-x)\times \\
&\quad \times \int_0^1 \mathrm{d}z\, \frac{(1-x)^{k+a} z^{k+a}}{1+(x-1)z} h\left(\frac{x}{1+(x-1)z}\right)\Bigg| \\
&= \left|(1-x)^{a+k+1}\int_0^1 \mathrm{d}z\, z^{a+k} \frac{h\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}\right| \\
&\leq |1-x|^{a+k+1}\,|G(x)|\,.
\end{aligned}$$

with

$$G(x) := \int_0^1 \mathrm{d}z\, \frac{h\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}. \tag{2.4.74}$$

For the $R_n(x)$ part we have

$$\begin{aligned}
|h(x) \star R_n(x)| &= \left|\int_x^1 \mathrm{d}y\, \frac{h(y)}{y} R_n\left(\frac{x}{y}\right)\right| \\
y = \frac{x}{1+(x-1)z} \to &= |(1-x)\times \\
&\quad \times \int_0^1 \mathrm{d}z\, \frac{R_n(1+(x-1)z)}{1+(x-1)z} h\left(\frac{x}{1+(x-1)z}\right)\Bigg|\,.
\end{aligned}$$

If $z \in [0;1]$ and $x \in (r;1)$ for some $r \in (0;1)$, then $1+(x-1)z \in (r;1)$. Thus by Definition 2.4.1 of the Big-O notation there exists an $r \in (0;1)$, $M \in \mathbb{R}_+$ such that

$$\begin{aligned}
|R_n(1+(x-1)z)| &\leq M\,[1-(1+(x-1)z)]^{a+n+1} \\
&\leq M(1-x)^{a+n+1} z^{a+n+1}.
\end{aligned}$$

Hence for $x \in (r;1)$,

$$\begin{aligned}
|h(x) \star R_n(x)| &\underset{x\in(r;1)}{\leq} M\,|1-x|^{a+n+2}\left|\int_0^1 \mathrm{d}z\, z^{a+n+1}\frac{h\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}\right| \\
&\underset{x\in(r;1)}{\leq} M\,|1-x|^{a+n+2}\,|G(x)|
\end{aligned}$$

with $G(x)$ given in (2.4.74). Therefore, the problem boils down to checking whether

$$G(x) \underset{x\to 1}{\sim} A(1-x)^{\delta}, \quad A \in \mathbb{R},\, \delta > -1. \tag{2.4.75}$$

For that, we simply can apply again Theorem 2.4.5 to $G(x)$. If it fails, **stop and return fail**. Otherwise, we obtain the expansion of $G(z)$ at order 0. If as in (2.4.75) we have $\delta > -1$, it means that $h(x) \star R_n(x)$ is really the remainder and the asymptotic/series expansion up to order n is given by looking at $h(x) \star P_n(x)$. Moreover, because we have

$$\forall k \in \{0, \dots, n\}, \quad |h(x) \star (1-x)^{k+a}| \leq |1-x|^{a+k+1}|G(x)|$$

it means in particular that

$$\forall k \in \{0, \dots, n\},\, h(x) \star (1-x)^{k+\alpha} \underset{x\to 1}{=} \mathcal{O}\left((1-x)^{a+a+\delta+1}\right).$$

Therefore, because Theorem 2.4.5 applies to each

$$f_k(x) = h(x) \star (1-x)^{k+a} = (1-x)^{k+a+1} \int_0^1 \mathrm{d}z\, z^{a+k} \frac{h\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z},$$

to get the expansion, we can expand each $f_k(x)$ around $x = 1$ up to the required order, or equivalently, due to the summation property of Taylor series, expand immediately the full $h(x) \star P_n(x)$.

- **Case 2**: It might be that two of our $\underline{V_i}$ and $\underline{V_j}$ are convolved with each other. Therefore, we suppose we have

$$J(x) = \mathcal{H}^*_{\underline{V_i}}(x) \star \mathcal{H}^*_{\underline{V_j}}(x).$$

We know from the previous step that we can write

$$\mathcal{H}^*_{\underline{V_i}}(x) = \underbrace{\sum_{k=0}^{n} \alpha_k (1-x)^{k+a}}_{P_n(x)} + R_n(x), \quad R_n(x) = \mathcal{O}\left((1-x)^{a+n+1}\right), \tag{2.4.76}$$

$$\mathcal{H}^*_{\underline{\mathrm{V_j}}}(x) = \underbrace{\sum_{k=0}^{n} \beta_k (1-x)^{k+b}}_{Q_n(x)} + S_n(x), \quad S_n(x) = \mathcal{O}\left((1-x)^{b+n+1}\right) \tag{2.4.77}$$

where $a, b \in (-1;1)$. Expanding, we get

$$\begin{aligned} J(x) &= P_n(x) \star Q_n(x) + P_n(x) \star S_n(x) \\ &\quad + R_n(x) \star Q_n(x) + R_n(x) \star S_n(x) \\ &= \sum_{p,q} \alpha_p \beta_q (1-x)^{p+a} \star (1-x)^{q+b} + \sum_p \alpha_p (1-x)^{p+a} \star S_n(x) \\ &\quad + \sum_p \beta_p R_n(x) \star (1-x)^{p+b} + R_n(x) \star S_n(x). \end{aligned}$$

In the same spirit as in the Case 1 and following the same steps, using in particular that for a certain $r \in (0;1)$, when $x \in (r;1)$ there exist $M, N \in \mathbb{R}_+$ such that

$$|R_n(x)| \leq M |1-x|^{n+1}, \quad |S_n(x)| \leq N |1-x|^{n+1}. \tag{2.4.78}$$

We bound each of the terms above:

$$\left|(1-x)^{p+a} \star (1-x)^{q+b}\right| \leq |1-x|^{p+q+a+b+1} \int_0^1 \mathrm{d}z \, \frac{z^{p+a}(1-z)^{q+b}}{[1+(x-1)z]^{q+b+1}} \tag{2.4.79}$$

$$\left|R_n(x) \star (1-x)^{p+b}\right| \leq M |1-x|^{n+p+b+2} \int_0^1 \mathrm{d}z \, \frac{z^{n+1}(1-z)^{p+b}}{[1+(x-1)z]^{p+b+1}} \tag{2.4.80}$$

$$\left|(1-x)^{p+a} \star S_n(x)\right| \leq N |1-x|^{n+p+a+2} \int_0^1 \mathrm{d}z \, \frac{z^{n+1}(1-z)^{p+a}}{[1+(x-1)z]^{p+a+1}} \tag{2.4.81}$$

$$|R_n(x) \star S_n(x)| \leq MN |1-x|^{2n+a+b+2} \int_0^1 \mathrm{d}z \, \frac{z^{n+1}(1-z)^{n+1}}{[1+(x-1)z]^{n+1}}. \tag{2.4.82}$$

We need therefore to prove that the class of integrals

$$H_{p,q}(x) := \int_0^1 \mathrm{d}z \, f_{p,q}(x,z), \quad f_{p,q}(x,z) := \frac{z^p(1-z)^q}{[1+(x-1)z]^{q+1}}$$

have an expansion of the form

$$H_{p,q}(x) \underset{x \to 1}{\sim} \gamma (1-x)^{\delta}, \quad \gamma \in \mathbb{R}, \delta > -1,$$

in other words, that they do not contribute to $\mathcal{O}\left((1-x)^{n+1}\right)$. We can simply apply again Theorem 2.4.5 here. First, we compute the

n^{th} derivative with respect to x of $f_{p,q}(x,z)$, send $x \to 1$ in it and bound it:

$$\left|\partial_x^n f_{p,q}(x,z)|_{x\to 1}\right| = \left|(1-z)^q(-r-1)^{(n)}z^{n+p}\right| \leq n!(n+r)^r(1-z)^q z^{n+p}.$$

Therefore we have that the n^{th} term in the (for the moment formal) series expansion of $f_{p,q}(x,z)$ is

$$\left|\frac{(x-1)^n}{n!}\,\partial_x^n f_{p,q}(x,z)|_{x\to 1}\right| \leq (n+r)^r(1-x)^n(1-z)^q z^{n+p}.$$

Since for our purposes we can consider $n \geq 1$, because $|z|<1$ we can even bound further as long as[b] $n+p \geq -1$

$$\left|\frac{(x-1)^n}{n!}\,\partial_x^n f_{p,q}(x,z)|_{x\to 1}\right| \leq (n+r)^r(1-x)^n(1-z)^q.$$

$(1-z)^q$ is also integrable on $[0;1]$ because even if we replace q by $q+b$, we have $b>-1$. In particular, we get

$$\int_0^1 \mathrm{d}z\,(1-z)^q = \frac{1}{1+q}.$$

We see that it is a particular case of Theorem 2.4.5 where $C=1$, which means we have uniform convergence on $(0;1)$ and in particular this tells us that

$$H_{p,q}(x) \underset{x\to 1}{\sim} \int_0^1 \mathrm{d}z\, z^p(1-z)^r = \frac{\Gamma(1+p)\Gamma(1+r)}{\Gamma(2+p+r)}$$

which is regular at $x=1$. This shows that in the equations (2.4.79) — (2.4.82) the integral does not contribute to the $|1-x|^p$, and moreover because (2.4.80),(2.4.81) and (2.4.82) are in $\mathcal{O}\left((1-x)^{n+1}\right)$ we can compute the asymptotics up to order n simply with (2.4.79).

[a] When there are no +-prescriptions involved, by Definition 2.1.12, the Mellin convolution is symmetric.

[b] Which is always the case, even if we replace as in the original expression $n+p$ by $n+p+a$, since $a \in (-1;1)$.

Remark 2.4.8. *During the bounding of the convolution, we made the change of variable* $y = \frac{x}{1+(x-1)z}$ *in order to obtain integral bounds that were independent of* x. *It is of course possible to make other changes of variables to achieve similar results.*

For instance, if we would have chosen $y = (1-z)x + z$, which send $y = x \to z = 0$ and $y = 1 \to z = 1$, the expression of the G function would have been a bit different:

$$\tilde{G}(x) = \int_0^1 \mathrm{d}z \, \frac{h\left(\frac{x}{(1-z)x+z}\right)}{(1-z)x+z}.$$

In practice, this does not change much for the bounding procedure, so we stick to the first choice of change of variables.

Example 2.4.6. *Let us illustrate the method above on an explicit example. Consider the following GCPL function*

$$\mathcal{H}^*_{\left(\left[\frac{1}{1+\lambda};\emptyset\right],\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{1}{2+\lambda};\emptyset\right]\right)\right)}(x) =: \mathcal{H}^*_{\underline{\mathrm{V}}}(x), \tag{2.4.83}$$

for which we want to compute the asymptotic expansion up to the order $p = 5$. Note that:

$$\mathcal{H}^*_{\underline{\mathrm{V}}}(x) = \int_x^1 \mathrm{d}x_1 \, \frac{1}{x+x_1} \int_{x_1}^1 \mathrm{d}x_1 \, \frac{1}{(x_2+2)\sqrt{x_1\,(x_2-x_1)}}. \tag{2.4.84}$$

- *There is only one element of the form*

  $$\underline{V_i} := \left([f_i(\lambda);\emptyset]\,,[g_i(\lambda);\emptyset]\right)$$

  in (2.4.83), which is:

  $$\underline{V_1} := \left(\left[\underbrace{\frac{1}{\sqrt{\lambda(1-\lambda)}}}_{=:g_1(\lambda)};\emptyset\right],\left[\underbrace{\frac{1}{2+\lambda}}_{=:g_2(\lambda)};\emptyset\right]\right). \tag{2.4.85}$$

- *We have to check Theorem 2.4.5 on $\underline{V_1}$, i.e., we have to show that*

  $$F(x) := (g_1 \star g_2)(x) = \int_0^1 \mathrm{d}z \, \underbrace{\frac{1-x}{\sqrt{(x-1)x(z-1)((x-1)z+3)}}}_{=:\bar{g}(x,z)} \tag{2.4.86}$$

  can be Taylor expanded around $x = 1$ by integrating term by term over z the expansion of $\bar{g}(x,z)$ around $x = 1$. Once again we split the expression of $(g_1 \star g_2)(x)$ into subfactors

  $$\left\{x+1, 1-x, \frac{1}{\sqrt{x-1}}, \frac{1}{\sqrt{x}}, \frac{1}{\sqrt{z-1}}, \frac{1}{xz-z+3}, \frac{1}{x+1}\right\},$$

then we differentiate the x-dependent factors, except for the $(1-x)$ that will already account for the series expansion. Plugging everything in (2.4.63) we obtain:

$$\frac{(1-x)^n}{n!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}\left(f(x,z)\right)\Big|_{x=1} = \frac{(1-x)(x-1)^{n-\frac{1}{2}}}{\sqrt{1-z}} \times \sum_{k_1,k_2,k_3=0}^{n} \frac{(-1)^{k_1+k_2+k_3}2^{-2k_2-k_3-1}3^{-k_1-1}\binom{2k_2}{k_2}z^{k_1}\,\frac{\mathrm{d}^{n-k_1-k_2-k_3}}{\mathrm{d}x^{n-k_1-k_2-k_3}}(x+1)\Big|_{x=1}}{(-k_1-k_2-k_3+n)!},$$

which then can be bounded by

$$\frac{A|x-1|^{n+\frac{1}{2}}\sum\limits_{k_1=0}^{n}\sum\limits_{k_2=0}^{n}\sum\limits_{k_3=0}^{n}2^{-k_3-1}3^{-k_1-1}}{\sqrt{1-z}}, \tag{2.4.87}$$

which means Theorem 2.4.5 applies. Exanding $\bar{g}(x,z)$ around $x=1$ up to order $p=5$ and integrating over z yields the expansion:

$$F(x) \underset{x\to 1}{=} \underbrace{\frac{2\sqrt{1-x}}{3}+\frac{13}{27}(1-x)^{3/2}+\frac{589(1-x)^{5/2}}{1620}+\frac{6689(1-x)^{7/2}}{22680}+\frac{1236433(1-x)^{9/2}}{4898880}}_{=:P_5(x)} + \mathcal{O}\left((1-x)^{11/2}\right).$$

Remark 2.4.9. *In order to perform the last step of the method, we point out that we naturally extend the values of letters appearing in generalized letters of GCPLs to (truncated) Taylor series.*

5. Once all the asymptotics of the form $J(x) = h(x) \star \mathcal{H}^*_{\underline{V_i}}(x)$ or $J(x) = \mathcal{H}^*_{\underline{V_i}}(x) \star \mathcal{H}^*_{\underline{V_j}}(x)$ are computed, we simply replace $J(x)$ by its expansion up to order n and then go to the next level of convolution. Here we have two possibilities:
   - We get a letter f multiplying the previous convolution $f(x)\cdot J(x)$ and then it suffices to expand $f(x)$ around $x=1$ and multiply both expansions then go to the next level of convolution, and so on.
   - We get a new convolution of the form $K(x)\star J(x)$, and then we redo step 2 where we replace in $\underline{V}$ the convolutions of the form $J(x)$ by their expansion, and then go the next level of convolution, and so on. We **stop and return fail if checking the theorem on one of the convolutions fails**.

6. We **stop** when **we reduced** $\underline{V}$ **to**

$$\underline{V} = [f(\lambda); \emptyset] .$$

In this case, $f(\lambda)$ is the asymptotic expansion of the GCPL around $x = 1$ up to order n.

Example 2.4.6 (continued)**.** *We finish the example in the following way:*

- *We replace*

$$\mathcal{H}^*_{\left(\left[\frac{1}{1+\lambda};\emptyset\right],\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{1}{2+\lambda};\emptyset\right]\right)\right)}(x) \longrightarrow \mathcal{H}^*_{\left(\left[\frac{1}{1+\lambda};\emptyset\right],[\mathrm{P}_5(\lambda);\emptyset]\right)}(x), \tag{2.4.88}$$

i.e., we consider the expansion instead of the full convolution corresponding to $\underline{V_1}$*, which means that we decreased the depth of the GCPL.*

- *Now, we are in the Case 1 where*

$$h(x) := \frac{1}{1+x},$$

so it means that we need to check Theorem 2.4.5 on

$$F_1(x) = \int_0^1 \mathrm{d}z \underbrace{\frac{h\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}}_{=:\bar{h}(x,z)}. \tag{2.4.89}$$

Once again, computing as above the n^{th} *Taylor series expansion term gives us*

$$\frac{(1-x)^n}{n!}\frac{\mathrm{d}^n}{\mathrm{d}x^n}\left(\bar{h}(x,z)\right)\Big|_{x=1} = (x-1)^n \sum_{k_1,k_2=0}^{n} \frac{(-1)^{k_1+k_2}2^{-k_1-k_2-2}(z+1)^{k_1}\frac{\mathrm{d}^{n-k_1-k_2}}{\mathrm{d}x^{n-k_1-k_2}}(x+1)\Big|_{x=1}}{(n-k_1-k_2)!},$$

which is bounded by

$$A|1-x|^n \sum_{0\le k_1,k_2\le n} 2^{-k_2-2} \tag{2.4.90}$$

and therefore Theorem 2.4.5 applies. Therefore, it means that we can now compute the asymptotic expansion of the convolution

$$\mathcal{H}^*_{\left(\left[\frac{1}{1+\lambda};\emptyset\right],[\mathrm{P}_5(\lambda);\emptyset]\right)}(x) = \int_0^1 \mathrm{d}z \underbrace{\frac{1-x}{1+\frac{x}{1+(x-1)z}}\frac{P_5(1+(x-1)z)}{1+(x-1)z}}_{=:q(x,z)}$$

by computing the series expansion of $q(x,z)$ *around* $x=1$ *up to order* 5 *and then integrating it out term by term. Finally, we get*

$$\underline{V} = [H(\lambda);\emptyset]$$

with

$$H(x) = \frac{2}{9}(1-x)^{3/2} + \frac{37}{135}(1-x)^{5/2} + \frac{3181(1-x)^{7/2}}{11340} + \frac{18611(1-x)^{9/2}}{68040}.$$

This yields

$$\begin{aligned}\mathcal{H}^*_{\left(\left[\frac{1}{1+\lambda};\emptyset\right],\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{1}{2+\lambda};\emptyset\right]\right)\right)}(x) &\underset{x\to 1}{=} \mathcal{H}^*_{[\mathrm{H}(\lambda);\emptyset]}(x)\\ &\underset{x\to 1}{=} \frac{2}{9}(1-x)^{3/2} + \frac{37}{135}(1-x)^{5/2}\\ &\quad + \frac{3181(1-x)^{7/2}}{11340} + \frac{18611(1-x)^{9/2}}{68040}\\ &\quad + \mathcal{O}\left((1-x)^{11/2}\right). \qquad (2.4.91)\end{aligned}$$

In order to compute this expansion using `RICA` we use the function `GLCbSeries` which takes two (main) arguments, the GCPL function and the order p up to which we want to expand it. If we use the example above, we obtain:

```
In[15]:= sum1 = GLCb[{{{1/(VarGL + 1), ∅}},
  {{{1/(Sqrt[VarGL(1 - VarGL)]); ∅}}, {{1/(VarGL + 2), ∅}}}}, x];
In[16]:= asymptotics1 = GLCbSeries[sum1, 5]
Out[16]= 2(1 - x)^(3/2)(x - 1)^(3/2)/(9(x - 1)^(3/2)) + 37(1 - x)^(5/2)(x - 1)^(5/2)/(135(x - 1)^(5/2)) +
  3181(1 - x)^(7/2)(x - 1)^(7/2)/(11340(x - 1)^(7/2)) + 18611(1 - x)^(9/2)(x - 1)^(9/2)/(68040(x - 1)^(9/2)) +
  O((x - 1)^(11/2))
```

Remark 2.4.10. *By default, the* `GLCbSeries` *functions applies the method described above to check the validity of the asymptotics expansion of the GCPL, but it is also possible to deactivate this check if one wants to see how a formal expansion would look like*[43]*. This is managed by setting the option of* `GLCbSeries` *to* `False`*.*

[43] But in this case there is no proof that this expansion is uniformly convergent in a certain region around $x=1$.

2.4.3. A note about singular integrands

Sometimes the integrand we deal with can contain singularities. In order to deal with them, as we will see in Section 2.5.3[44], we will use the Cauchy principal value prescription for such arising divergent Mellin transforms or other integrals:

Definition 2.4.11 (Cauchy principal value)**.** *Let $f = f(x)$ be a function defined on an interval $I = [a; b] \subset \mathbb{R}$ except for a point $c \in I$. If the improper integral $\int_a^b \mathrm{d}x\, f(x)$ is divergent but the quantity*

$$\fint_a^b \mathrm{d}x\, f(x) := \lim_{\epsilon \to 0} \left(\int_a^{c-\epsilon} \mathrm{d}x\, f(x) + \int_{c+\epsilon}^b \mathrm{d}x\, f(x) \right) \tag{2.4.92}$$

is finite, then this limit is the ***Cauchy principal value*** *of $\int_a^b \mathrm{d}x\, f(x)$. This generalises naturally to $p \in \mathbb{N}^\star$ singularities, where in the expression (2.4.92) we decompose the integral into $p+1$ expressions of a similar kind.*

Example 2.4.7. *Suppose we have the function $f : x \mapsto \frac{1}{(x-1)(x-2)}$ and we want to compute its integral on $[0; 3]$. It has two singularities at $x = 1$ and $x = 2$, so we have to use Cauchy's principal value:*

$$\begin{aligned}
\fint_0^3 \mathrm{d}x\, \frac{1}{(x-1)(x-2)} &= \lim_{\epsilon \to 0} \left(\int_0^{1-\epsilon} \mathrm{d}x\, \frac{1}{(x-1)(x-2)} + \int_{1+\epsilon}^{2-\epsilon} \mathrm{d}x\, \frac{1}{(x-1)(x-2)} \right. \\
&\qquad \left. + \int_{2+\epsilon}^3 \mathrm{d}x\, \frac{1}{(x-1)(x-2)} \right) \\
&= \lim_{\epsilon \to 0} \left[-4 \tanh^{-1}(1 - 2\epsilon) + 2 \log\left(\frac{1+\epsilon}{\epsilon} \right) \right] \\
&= -\log 4.
\end{aligned}$$

Remark 2.4.11. *Another possible approach [16] is to define the Mellin transform in the following way*

$$\mathbf{M}\left[f(x)\right](n) := \lim_{\epsilon \to 0} \int_0^1 \mathrm{d}x\, x^n \frac{f(x + i\epsilon) + f(x - i\epsilon)}{2}. \tag{2.4.93}$$

More generally, for an integral of a function f over an interval $I = [a; b] \subset \mathbb{R}$ where f admits some singularities, we can define it as

$$\lim_{\epsilon \to 0} \int_a^b \mathrm{d}x\, \frac{f(x + i\epsilon) + f(x - i\epsilon)}{2}. \tag{2.4.94}$$

[44] See in particular Remarks 2.5.4 and 2.5.8.

2.5. Computing asymptotic expansions using Mellin representations

As starting point, we assume that the binomial sum[45] is given in terms of its Mellin transform, after we have applied inversion techniques from the previous section. We get the following representation:

$$\mathcal{BS}(n) = \sum_{i=1}^{k} C_i \int_0^1 \mathrm{d}x[(a_i x)^n - 1] f_i(x) + K \tag{2.5.1}$$

where $C_i \in \{1, \ldots, k\}$ and $K \in \mathbb{K}$ are constants that we suppose can be computed either exactly analytically or expressed in a form that allows for a numerical evaluation to high precision. To illustrate the notation above, let us consider the following binomial sum:

$$\sum_{i=1}^{n} \frac{1}{i\binom{2i}{i}} \sum_{j=1}^{i} \binom{2j}{j} (-2)^j. \tag{2.5.2}$$

We compute its Mellin inverse using `RICA`:

```
In[17]:= sum3 = GS[{1/(VarGL*Binomial[2*VarGL, VarGL]), (-2)^VarGL*Binomial[2*VarGL, VarGL]}, n];
In[18]:= melRep3 = SumToMellin[sum3, C, x]
```

$$\text{Out[18]=} \left\{\frac{2}{3}\text{Mellin}\left[(-2)^{\mathtt{n}}\mathtt{x}^{\mathtt{n}} - 1, \frac{3\sqrt{8\mathtt{x}+1}-1}{(2\mathtt{x}+1)\sqrt{8\mathtt{x}+1}}\right] - \frac{\mathtt{C}_1 \text{Mellin}\left[4^{-\mathtt{n}}\mathtt{x}^{\mathtt{n}} - 1, \frac{1}{\sqrt{1-\mathtt{x}}(\mathtt{x}-4)}\right]}{\pi}, \left\{\mathtt{C}_1 \to \text{GLb}\left[\left\{\frac{\sqrt{\frac{\text{VarGL}}{1-\text{VarGL}}}}{\text{VarGL}+\frac{1}{8}}\right\}, 0\right]\right\}\right\}$$

We see here that we obtain a constant that can be computed explicitely in this case:

```
In[19]:= ReleaseHold[ToHarmonicSumsIntegrate[GLb[{Sqrt[VarGL/(1-VarGL)]/(VarGL+1/8)}, 0]]]
```

$$\text{Out[19]=} \frac{2\pi}{3}$$

Plugging it in and rearranging a bit the expression we obtain the result as in [16][46]:

$$\sum_{i=1}^{n} \frac{1}{i\binom{2i}{i}} \sum_{j=1}^{i} \binom{2j}{j} (-2)^j = \int_0^1 \mathrm{d}x \, \frac{(-2x)^n - 1}{x + \frac{1}{2}} \left(1 - \frac{1}{6\sqrt{2}\sqrt{x+\frac{1}{8}}}\right)$$

[45] As per Definition 2.1.2, we will sometimes call binomial sums IBS.
[46] Equation (5.20) page 26.

$$-\frac{2}{3}\int_0^1 \mathrm{d}x\,\frac{\left(\frac{x}{4}\right)^n-1}{x-4}\frac{1}{\sqrt{1-x}}. \tag{2.5.3}$$

In this particular case, (2.5.1) specialises to:

$$C_1=1,\ C_2=-\frac{2}{3},\ a_1=-2,\ a_2=\frac{1}{4},$$

$$f_1(x)=\frac{1}{x+\frac{1}{2}}\left(1-\frac{1}{6\sqrt{2}}\sqrt{x+\frac{1}{8}}\right),\ f_2(x)=\frac{1}{x-4}\frac{1}{\sqrt{1-x}}.$$

We want to compute the asumptotic expansion of the binomial-weighted sums as $n\to+\infty$ up to a certain order, from now on denoted with $p\in\mathbb{N}$, using their Mellin representation. There exist several methods to do so, depending on the integrand of the Mellin representation and its regularity. In the following we will detail them as well as the conditions under which they work and their technical and/or mathematical limits.

Given that all the integrals are well-defined on their own for $n\geq 1$, we can compute them separately. Thus we will focus on an explicitely given integral which has the following form:

$$\int_0^1 \mathrm{d}x\,\frac{(ax)^n-1}{x-\frac{1}{a}}f(x)=\int_0^1 \mathrm{d}x\,\left[(ax)^n-1\right]\tilde{f}(x)=\tilde{\mathbf{M}}_a\left[\tilde{f}(x)\right](n) \tag{2.5.4}$$

where

$$\tilde{f}(x):=\frac{f(x)}{x-\frac{1}{a}}.$$

We recognize here an extended Mellin transform[47], as expected.

Remark 2.5.1. *After the computation of the Mellin inverse in Section 2.3, we might end up with a sum of several Mellin representations of the form*

$$\mathcal{BS}_{(a_1,\dots,a_k)}(n)=\sum_{p=1}^{q}\tilde{\mathbf{M}}_{a_p}\left[f_p(x)\right](n)+C,\quad C\in\mathbb{K}.$$

In this case, we do not actually need to compute asymptotics for each of the $\tilde{\mathbf{M}}_{a_p}\left[f_p(x)\right](n)$*, as*

$$|a|>|b|\Rightarrow b^n\underset{n\to+\infty}{=}o(a^n).$$

[47] See Definition 2.1.11

For example:

$$2^n\left(\frac{\alpha_1}{n}+\frac{\alpha_2}{n^2}+\cdots\right)+3^n\left(\frac{\beta_1}{n^{10}}+\frac{\beta_2}{n^{11}}+\cdots\right)\to 3^n\left(\frac{\beta_1}{n^{10}}+\frac{\beta_2}{n^{11}}+\cdots\right)\ \textit{dominates.}$$

To speed up calculations, and to avoid complicated cases for the non-dominant contributions, this tactic is applied by default in our implementation in `RICA`[48]*. In other instances it might be useful to calculate the full information, in particular if cancellations in combinations of other contributions arise. This extra feature is activated with the option* `ExtendedAsymptotics→True`*.*

2.5.1. Factorial series approach (FSA)

The first method is the simplest one, and consists in using iterated integration by parts[49]. Here we need the following conditions to perform the expansion using this approach:

- $\tilde{f}$ is at least p times differentiable at $x=1$.
- $\tilde{f}$ is $\mathcal{C}^0$ on $(0;1]$ and integrable on $[0;1]$[a] or at least integrable as a Cauchy principal value.

[a] Since $(ax)^n-1$ is regular at 0, f must be integrable at 0 in order to be integrable.

Remark 2.5.2. *Among all the approaches given below in Sections 2.5.2 — 2.5.3, this one requires the strongest conditions on f.*

If that is the case, we can proceed as follows:

1. Split the integral in two parts, the n-dependent part where the a^n is factored out from the integral, and the constant part:

$$\begin{aligned}\int_0^1 \mathrm{d}x[(ax)^n-1]\tilde{f}(x) &= a^n\int_0^1 \mathrm{d}x\, x^n\tilde{f}(x)-\int_0^1 \mathrm{d}x\,\tilde{f}(x)\\ &= a^n\mathbf{M}\left[f(x)\right](n)-\mathbf{M}\left[f(x)\right](0).\end{aligned}$$

2. Depending on f, compute the constant value $M[f(x)](0)$ using functions such as `ReduceConstants` from the `HarmonicSums` package, or try to obtain the integral using directly Mathematica.

[48] For more details see the Appendix A.

[49] The name factorial series comes from the form of the repeated integration by parts, for more details see [70, 75].

3. Perform integration by parts p times on the n-dependent part:

$$\begin{aligned}\mathbf{M}\left[f(x)\right](n) &= \left[\frac{x^{n+1}}{n+1}\tilde{f}(x)\right]_0^1 - \frac{1}{n+1}\int_0^1 \mathrm{d}x\, x^{n+1}\tilde{f}'(x)\\ &= \frac{\tilde{f}(1)}{n+1} - \frac{1}{n+1}\left[\left[\frac{x^{n+2}}{n+2}\tilde{f}'(x)\right]_0^1 - \frac{1}{n+2}\int_0^1 \mathrm{d}x\, x^{n+2}\tilde{f}''(x)\right]\\ &\vdots \qquad\qquad\qquad \vdots\\ &= \sum_{i=0}^{p}(-1)^i\frac{\tilde{f}^{(i)}(1)}{(n+1)_{i+1}} + \underbrace{\frac{(-1)^p}{(n+1)_p}\int_0^1 \mathrm{d}x\, x^{n+p+1}\tilde{f}^{(p+1)}(x)}_{\mathcal{O}\left(\frac{1}{n^{p+1}}\right)},\end{aligned} \tag{2.5.5}$$

where we made use of the rising factorial $(n+1)_{i+1}$ as defined in the Conventions. It is enough to integrate by parts p times, as $\frac{1}{(n+1)_p} \underset{n\to+\infty}{\sim} \frac{1}{n^p}+\mathcal{O}\left(\frac{1}{n^{p+1}}\right)$, so we get all the needed contributions for the lower power terms.

The constant parts of the integration by parts (IBP) simplify the way they do for the following reason: since f must be integrable at $x=0$, it means we have $\tilde{f}(x) \underset{x\to 0}{\sim} x^\alpha \log^\beta x$, with $\alpha \geq -\frac{1}{2}$, $\beta > 0$. Since $n \geq 1$, this means for $i \geq 0$, that $x^{n+i}\tilde{f}^{(i)}(x) \underset{x\to 0}{\sim} x^{\alpha+n+i}\log^\beta x$ and $\alpha+n+i \geq \frac{1}{2}$ so $x^{n+i}\tilde{f}^{(i)}(x)$ always vanishes at $x=0$. Thus in $\left[x^{n+1}\tilde{f}^{(i)}(x)\right]_0^1$ only the part at $x=1$ remains.

4. Using Mathematica's `Series` or `HarmonicSums`'s `HarmonicSumsSeries`, expand (2.5.5) for n at $+\infty$ up to the order[50] p and add the computed constant finally yield

$$\tilde{\mathbf{M}}_a\left[f(x)\right](n) \underset{n\to+\infty}{=} -\mathbf{M}\left[f(x)\right](0) + a^n\left(\sum_{k=1}^{p}\frac{h_k}{n^k} + \mathcal{O}\left(\frac{1}{n^{p+1}}\right)\right)$$

where $h_k \in \mathbb{R}$ for $k \in \{1,\dots,p\}$.

Example 2.5.1. *Consider the first integral of* (2.5.3)*, i.e.,*

$$\tilde{\mathbf{M}}_{-2}\left[\tilde{f}_1(x)\right](n) = \int_0^1 \mathrm{d}x\, \frac{(-2x)^n - 1}{x+\frac{1}{2}} \underbrace{\left(1 - \frac{1}{6\sqrt{2}\sqrt{x+\frac{1}{8}}}\right)}_{=:f_1(x)}$$

[50] Since $\frac{1}{(n+1)_k} \underset{p\to+\infty}{\sim} \frac{1}{n^k}$, doing the integration by part up to order p is enough to get all the terms contributing to lower powers of the expansion.

with

$$\tilde{f}_1(x) := \frac{f_1(x)}{x+\frac{1}{2}}.$$

The function $f_1(x)$ is integrable, well-defined and even infinitely many times differentiable at $x = 1$, so we can apply the method to compute an expansion, e.g., up to the order 5. First we compute the constant term:

$$\mathbf{M}\left[f_1(x)\right](0) = \int_0^1 \mathrm{d}x\, \tilde{f}_1(x) = \log(3) - \frac{\pi}{9\sqrt{3}}.$$

Since we want to get the expansion up to order five, we carry out the integration by parts five times. This gives:

$$\begin{aligned}\mathbf{M}\left[f(x)\right](n) =& \frac{16}{27(n+1)} + \frac{88}{243(n+1)(n+2)} + \frac{320}{729(n+1)(n+2)(n+3)} \\ &+ \frac{5120}{6561(n+1)(n+2)(n+3)(n+4)} \\ &+ \frac{104960}{59049(n+1)(n+2)(n+3)(n+4)(n+5)} \\ &+ \frac{1}{(n+1)_5}\int_0^1 \mathrm{d}x\, x^{n+6}\tilde{f}^{(6)}(x).\end{aligned}$$

Finally, by expanding the expression above at $+\infty$ for n up to order $p = 5$, adding the constant part and multiplying by the $(-2)^n$ factor, we get:

$$\begin{aligned}\tilde{\mathbf{M}}_{-2}\left[f_1(x)\right](n) \underset{n\to+\infty}{=} (-2)^n &\left(\frac{16}{27n} - \frac{56}{243n^2} - \frac{40}{729n^3}\right. \\ &\left. + \frac{584}{6561n^4} + \frac{6392}{59049n^5} + \mathcal{O}\left(\frac{1}{n^6}\right)\right) + \frac{\pi}{9\sqrt{3}} - \log(3).\end{aligned}$$

This computation can be done by calling the function `AsymptoticsMellin` *from the* `RICA` *package*[51] *— which incidentally also calls all the other methods according to the type of inverse Mellin transform we deal with. This function takes four arguments: the Mellin integral we want to expand (here $\tilde{M}_{-2}[f_1(x)](n)$), the variable (here x), n as well as the order up to which we want the expansion (here 5). Because the expansion is exponentially divergent at $n \to +\infty$, we have to use the option* `ExtendedAsymptotics` *to print also the constant:*

```
In[20]:= AsymptoticsMellint[Mellin[(-2x)^n - 1, (1 - 1/(6√2 √(x+1/8)))/(x+1/2)], x, n, 5,
         ExtendedAsymptotics → True]
```

[51] See Appendix A.3.1.

$$\text{Out[20]=}\ \frac{1}{27}\left(\sqrt{3}\pi - 27\log(3)\right) + \frac{(-1)^n 2^{n+4}}{27n} - \frac{7(-1)^n 2^{n+3}}{243n^2} - \frac{5(-1)^n 2^{n+3}}{729n^3} + \frac{73(-1)^n 2^{n+3}}{6561n^4} + \frac{799(-1)^n 2^{n+3}}{59049n^5}$$

2.5.2. Geometric series approach (GSA)

If we want to compute $\tilde{M}_a[\tilde{f}(x)](n)$ but the function f is not regular enough at $x = 1$, i.e., either f or one of its derivatives are not defined at $x = 1$, we have to apply another method. In this section, we present a simple method that relies on the geometric series and requires two conditions[52]:

- The function $|f|$ is integrable on $[0; 1]$.
- We require $|a| < 1$

Both of these conditions imply that the constant term can once again be split and computed separately[53]. In this method, the idea is to expand the geometric series $\frac{1}{x-\frac{1}{a}}$ and to reduce the problem to the computation of the shifted version $\mathbf{M}\left[f(x)\right](n+k)$ instead of dealing with the full $\tilde{f}(x) = \frac{f(x)}{x-\frac{1}{a}}$ function — as we can see, this method basically "gets rid" of the $\frac{1}{x-\frac{1}{a}}$ factor[54]. More precisely, we perform the following steps:

1. We split once again the constant term $C = \mathbf{M}\left[f(x)\right](0)$.
2. We start by expanding the geometric series:

$$\frac{1}{x-\frac{1}{a}} = (-a)\frac{1}{1-ax} = (-a)\sum_{k=0}^{\infty}(ax)^k.$$

3. We plug this result into the Mellin transform:

$$\begin{aligned}(-a)\int_0^1 \mathrm{d}x\, x^n \sum_{k=0}^{\infty}(ax)^k f(x) &= (-a)\int_0^1 \mathrm{d}x \sum_{k=0}^{\infty} a^k x^{n+k} f(x) \\ &= (-a)\sum_{k=0}^{\infty} a^k \int_0^1 \mathrm{d}x\, x^{n+k} f(x)\end{aligned}$$

[52] We use here the notations of (2.5.4).

[53] Indeed, $|a| < 1$ implies that the function does not have an (apparent) singularity on $[0; 1]$. If that is the case, we will use another method described in Section 2.5.3.

[54] However, because it entails computations of Mellin transforms, it will work only in simple cases, as we will see below

$$= (-a) \sum_{k=0}^{\infty} a^k M[f(x)](n+k)$$

where we used the fact that $|a^k x^{n+k} f(x)| \leq |f(x)|$ on $[0;1]$ and $|f|$ is integrable on $[0;1]$.

4. We remember that we can use (2.1.1) to relate the shifted Mellin transform to the non-shifted one $k \leftarrow n+k$, and then we compute the actual Mellin transform using for example `Mellin` or more generally `GeneralMellin` from `HarmonicSums` [9, 10, 12, 72–74] using the method sketched out in Remark 2.3.1. Because we try to compute the Mellin transform, we will end up with an expression that can again contain sums — if that is the case and/or if the method takes too much time[55], we basically consider that the method fails and switch to the next one, described in Section 2.5.3.

5. Once step 4 is carried out, the idea is to asymptotically expand $M[f(x)](n+k)$ for n around $+\infty$ up to order p. As we have mentioned earlier, we suppose that the expression is simple enough[56] to not contain sums which allows us to do this automatically:

$$\mathbf{M}\left[f(x)\right](n+k) \underset{n\to+\infty}{=} \sum_{i=0}^{p} \frac{\alpha_i(k)}{n^i} + \mathcal{O}\left(\frac{1}{n^{p+1}}\right)$$

where the $\alpha_i(k)$ are coefficients that depend on k.

6. Finally, we compute the infinite sum:

$$(-a) \sum_{k=0}^{\infty} \sum_{i=0}^{p} a^k \frac{\alpha_i(k)}{n^i} = \sum_{i=0}^{p} \frac{A_i}{n^i} \text{ with } A_i := \sum_{k=0}^{\infty} a^k \alpha_i(k).$$

To this end, adding the constant term we get:

$$\tilde{\mathbf{M}}_a\left[f(x)\right](n) \underset{n\to+\infty}{=} -C + \sum_{i=0}^{p} \frac{A_i}{n^i} + \mathcal{O}\left(\frac{a^n}{n^{p+1}}\right).$$

Example 2.5.2. *Let us use as an example the second Mellin integral from the expression (2.5.3), i.e., more precisely:*

$$\tilde{\mathbf{M}}_{\frac{1}{4}}\left[\tilde{f}_2(x)\right](n) = \int_0^1 \mathrm{d}x \, \frac{\left(\frac{x}{4}\right)^n - 1}{x-4} \underbrace{\frac{1}{\sqrt{1-x}}}_{=:f_2(x)}, \quad \tilde{f}_2(x) := \frac{f_2(x)}{x-4}.$$

Let us follow the steps of the method above.

[55] There are basically time constrains in the `RICA` code that will prevent the computation from running amok.

[56] In the Example 2.5.2 below we present such a situation.

1. *First we compute the constant which gives us:*
$$\mathbf{M}\left[f_2(x)\right](0) = \int_0^1 \mathrm{d}x\, \frac{1}{x-4}\frac{1}{\sqrt{1-x}} = -\frac{\pi}{3\sqrt{3}}.$$

2. *Expanding* $\frac{1}{x-4}$ *yields*
$$\frac{1}{x-4} = \left(-\frac{1}{4}\right)\frac{1}{1-\frac{x}{4}} = \left(-\frac{1}{4}\right)\sum_{k=0}^{\infty}\left(\frac{x}{4}\right)^k.$$

3. *Plugging this expansion into the integral, we get:*
$$\mathbf{M}\left[\tilde{f}_2(x)\right](n) = \int_0^1 \mathrm{d}x\, \frac{x^n}{x-4} f_2(x) = -\frac{1}{4}\sum_{k=0}^{\infty}\frac{\mathbf{M}\left[f_2(x)\right](n+k)}{4^k}.$$

4. *Using* `GeneralMellin` *from* `HarmonicSums`*, we compute* $M[f_2(x)](n+k)$*:*
$$\frac{\mathbf{M}\left[f_2(x)\right](n+k)}{4^k} = \frac{2^{2n+1}((k+n)!)^2}{(2k+2n+1)(2k+2n)!}$$
which is a simple hypegeometric expression in n *and* k *without sums.*

5. *Therefore, directly expanding around* $n=+\infty$ *yields*
$$\begin{aligned}\frac{\mathbf{M}\left[f_2(x)\right](n+k)}{4^k} \underset{n\to+\infty}{=} \frac{e^n}{4^k}\Bigg(&\frac{\sqrt{\pi}}{\sqrt{n}} - \frac{\sqrt{\pi}(4k+3)}{8n^{3/2}} + \frac{\sqrt{\pi}\left(48k^2+72k+25\right)}{128n^{5/2}}\\ &- \frac{5\sqrt{\pi}(4k+3)\left(16k^2+24k+7\right)}{1024n^{7/2}}\\ &+ \frac{7\sqrt{\pi}\left(1280k^4+3840k^3+4000k^2+1680k+237\right)}{32768n^{9/2}}\\ &+\mathcal{O}\left(\frac{1}{n^{11/2}}\right)\Bigg)\end{aligned}$$

6. *Finally, summing for* k *and finally substracting the constant term we get:*
$$\begin{aligned}\tilde{\mathbf{M}}_{\frac{1}{4}}\left[f_2(x)\right](n) =& \frac{\pi}{3\sqrt{3}} + 4^{-n}\Bigg(-\frac{\sqrt{\pi}}{3\sqrt{n}} + \frac{13\sqrt{\pi}}{72n^{3/2}} - \frac{227\sqrt{\pi}}{1152n^{5/2}}\\ &+\frac{9565\sqrt{\pi}}{27648n^{7/2}} - \frac{2308873\sqrt{\pi}}{2654208n^{9/2}} + \mathcal{O}\left(\frac{1}{n^{11/2}}\right)\Bigg)\end{aligned}$$

Once again, the expansion is computed using `AsymptoticsMellint`*:*

```
In[21]:= AsymptoticsMellint[Mellin[(x/4)^n - 1, 1/((x - 4)Sqrt[1 - x])], x, n, 5]

Out[21]= 13Sqrt[π]2^(-2n-3)/9n^(3/2) - 227Sqrt[π]2^(-2n-7)/9n^(5/2) + 9565Sqrt[π]2^(-2n-10)/27n^(7/2) -
  2308873Sqrt[π]2^(-2n-15)/81n^(9/2) - Sqrt[π]2^(-2n)/3Sqrt[n] + π/3Sqrt[3]
```

Note that the `ExtendedAsymptotics` *option is no longer really necessary here, as the asympotitcs expansion is dominated by a* $\left(\frac{1}{4}\right)^n$ *term.*

2.5.3. Change of variables (COV)

The first two methods described above, implemented in our `RICA` package and being usually rather fast, have several drawbacks: in the factorial series approach, the integrand needs to be regular enough at $x = 1$ for the method to work. Since in physics applications distributions can be always separated[57], one may end up with regular representations, separating convolutions of $[S_1(n)]^k$ properly. In the geometric series approach, we have to come back to the discrete world through the Mellin transform, and therefore it can be costly and inefficient — so it will work mostly with simpler cases[58]. For all the other cases, which which make up the majority, we have to take another approach. We follow the method sketched in [16], partly based on the "factorial series" described in [75, 76] and related as well to saddle point approximation techniques (such as Laplace's method). Details will be given below together with some extensions to fit the extended binomial sums case. We will call this method simply the "change of variable" (COV) method for reasons that will become clear below. This general method can be subdivided in two cases, depending on the singularities of the Mellin transform argument.

Before describing the actual methods, we start by proving some properties that will be used in both cases. In particular, we will have to perform integrations of certain kind of integrals. We start with the following formula from [16]

$$\int_0^\infty \mathrm{d}z\, e^{-nz} z^c \log(z)^k = \frac{\partial^k}{\partial c^k} \frac{\Gamma(c+1)}{n^{c+1}} = \sum_{i=0}^{k} (-1)^i \binom{k}{i} \Gamma^{(k-i)}(c+1) \frac{\log(n)^i}{n^{c+1}} \quad (2.5.6)$$

and extend it to a more general case.

Lemma 2.5.1. *For $a, c, n \in \mathbb{N}$, $b, d \in \mathbb{R}_+$, we have:*

$$\begin{aligned} \mathcal{I}_{a,b,c,d}(n) &:= \int_{-b}^{+\infty} \mathrm{d}x\, e^{-nx} x^a (x+b)^d \log(x+b)^c \\ &= e^{nb} \sum_{k=0}^{a} \sum_{i=0}^{c} \binom{a}{k} \binom{c}{i} (-1)^i (-b)^{a-k} \Gamma^{(c-i)}(k+d+1) \frac{\log(n)^i}{n^{k+d+1}}. \end{aligned} \quad (2.5.7)$$

Proof. We proceed as follows:

$$\begin{aligned} \mathcal{I}_{a,b,c,d}(n) &= \int_{-b}^{+\infty} \mathrm{d}x\, e^{-nx} x^a (x+b)^d \log(x+b)^c \\ (y = x+b) \to &= \int_0^{+\infty} \mathrm{d}y\, e^{-n(y-b)} (y-b)^a y^d \log(y)^c \end{aligned}$$

[57] See [13].

[58] Usually those where the inverse Mellin transform does not contain high order GRPL, coming from nested sums on the binomial sums side.

$$\left((y-b)^a = \sum_{k=0}^{a} \binom{a}{k} (-b)^{a-k} y^k\right) \to = e^{nb} \sum_{k=0}^{a} \binom{a}{k} (-b)^{a-k} \int_0^{+\infty} \mathrm{d}y\, e^{-ny} y^{d+k} \log(y)^c .$$

We can now apply (2.5.6) where we set $n \to n, c \to k+d, k \to c$, and we obtain:

$$\mathcal{I}_{a,b,c,d}(n) = e^{nb} \sum_{k=0}^{a} \binom{a}{k} (-b)^{a-k} \sum_{i=0}^{c} (-1)^i \binom{c}{i} \Gamma^{(c-i)}(k+d+1) \frac{\log(n)^i}{n^{k+d+1}}$$

which concludes the proof. □

Since we will need to obtain asymptotic integrals with an exponential kernel, we will heavily rely on the following classical result [30].

Lemma 2.5.2 (Watson's lemma)**.** *Let f be a function defined on $\mathbb{R}_+$ such that $|f(z)| \leq e^{cz}$ for some $c \in \mathbb{R}$ when $z \to +\infty$ and*

$$\forall m \in \mathbb{N}, \quad f(z) \underset{z\to 0^+}{=} \sum_{k=0}^{m} \alpha_k z^{\beta+\gamma k} + R_m(z), \quad \beta > -1,\ \gamma > 0 \tag{2.5.8}$$

where $\alpha_k \in \mathbb{R}$ and

$$R_m(z) \underset{z\to 0}{=} \mathcal{O}\left(z^{\beta+\gamma(m+1)}\right). \tag{2.5.9}$$

Then we have

$$\forall p \in \mathbb{N},\ I(n) := \int_0^{+\infty} \mathrm{d}z\, e^{-zn} f(z) \underset{n\to+\infty}{=} \sum_{k=0}^{m} \alpha_k \int_0^{+\infty} \mathrm{d}z\, e^{-zn} z^{\beta+\gamma k} + \mathcal{R}_m(n) \tag{2.5.10}$$

where

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{1}{n^{\beta+\gamma(m+1)}}\right). \tag{2.5.11}$$

Proof. Plugging the expansion (2.5.8) into $I(n)$ we get

$$\begin{aligned} I(n) &= \int_0^{+\infty} \mathrm{d}z\, e^{-zn} \left(\sum_{k=0}^{m} \alpha_k z^{\beta+\gamma k} + R_m(z)\right) \\ &= \sum_{k=0}^{m} \alpha_k \underbrace{\int_0^{+\infty} \mathrm{d}z\, e^{-zn} z^{\beta+\gamma k}}_{=:B_k(n)} + \int_0^{+\infty} \mathrm{d}z\, e^{-zn} R_m(z). \end{aligned} \tag{2.5.12}$$

Now we have to show that

$$\mathcal{R}_m(n) = \int_0^{+\infty} \mathrm{d}z\, e^{-zn} R_m(z) \overset{?}{=} \mathcal{O}\left(\frac{1}{n^{m+1}}\right).$$

By hypothesis (2.5.9) and Definition 2.4.2 we get

$$\forall \eta > 0,\ \exists \epsilon_\eta > 0,\ \forall z \in (0;\eta) \quad |R_m(z)| \leq \epsilon_\eta z^{\beta+\gamma(m+1)}.$$

Because we also require $|f(z)| \leq e^{cz}$ as $z \to +\infty$ we can also bound

$$|R_m(z)| \leq M_\eta e^{cz}$$

for $z \in (\eta; +\infty)$ for some $M_\eta \in \mathbb{R}_+$. Therefore for any $\eta > 0$ we have

$$\begin{aligned} |\mathcal{R}_m(n)| &\leq \epsilon_\eta \int_0^\eta \mathrm{d}z\, e^{-zn} z^{\beta+\gamma(m+1)} + M_\eta \int_\eta^{+\infty} \mathrm{d}z\, e^{-zn} e^{cz} \\ \text{positivity of integrand} \to &\leq \epsilon_\eta \int_0^{+\infty} \mathrm{d}z\, e^{-zn} z^{\beta+\gamma(m+1)} + \frac{M_\eta}{n-c} e^{-(n-c)\eta} \\ &= \epsilon_\eta B_{m+1}(n) + \frac{M_\eta}{n-c} e^{-(n-c)\eta}. \end{aligned} \tag{2.5.13}$$

To compute $B_{m+1}(n)$ we can use the Lemma 2.5.1 above with $a, b, c = 0$ and we get

$$B_{m+1}(n) = \frac{\Gamma(\beta + \gamma(m+1))}{n^{\beta+\gamma(m+1)}}.$$

Moreover, we have

$$\forall \epsilon > 0,\ \exists n_\epsilon > 0,\ \forall n \geq n_\epsilon,\ \left| \frac{M_\eta}{n-c} e^{-(n-c)\eta} \right| \leq \frac{\epsilon}{n^{\beta+\gamma(m+1)}}$$

which means that

$$\forall \epsilon > 0,\ \exists n_\epsilon > 0,\ \forall n \geq n_\epsilon,\ |\mathcal{R}_m(n)| \leq \epsilon \frac{A_m}{n^{\beta+\gamma(m+1)}}$$

for $A_m = \Gamma(\beta + \gamma(m+1)) + 1$, or in other words

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{1}{n^{\beta+\gamma(m+1)}} \right)$$

which concludes the proof. □

Given that we may have in our computations integrals that are a bit more complicated, we can extend Watson's lemma by considering also log-contributions.

Lemma 2.5.3 (Extended Watson's lemma)**.** *Let $b \in \mathbb{R}_+$, f be a function defined on $(-b; +\infty)$ such that $|f(z)| \leq e^{cz}$ for some $c \in \mathbb{R}$ when $z \to +\infty$ and*

$$\forall m \in \mathbb{N}, \quad f(z) \underset{z\to-b}{=} \sum_{k=0}^{m} \alpha_k (z+b)^{\beta+\gamma k} \log(z+b)^c + R_m(z) \tag{2.5.14}$$

where $\beta > -1$, $\gamma > 0$, $c \in \mathbb{N}$, $\alpha_k \in \mathbb{R}$ *and*

$$R_m(z) \underset{z\to -b}{=} \mathcal{O}\left((z+b)^{\beta+\gamma(m+1)}\log(z+b)^c\right). \tag{2.5.15}$$

Then we have

$$\begin{aligned}\forall p \in \mathbb{N},\ I(n) &:= \int_{-b}^{+\infty} \mathrm{d}z\, e^{-zn} f(z) \\ &\underset{n\to+\infty}{=} \sum_{k=0}^{m} \alpha_k \int_{-b}^{+\infty} \mathrm{d}z\, e^{-zn}(z+b)^{\beta+\gamma k}\log(z+b)^c + \mathcal{R}_m(n)\end{aligned} \tag{2.5.16}$$

where

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{e^{nb}\log(n)^c}{n^{\beta+\gamma(m+1)}}\right). \tag{2.5.17}$$

Proof. The proof goes is analogous to that of Watson's lemma, except that this time we have to use Lemma 2.5.1 where we only set $a = 0$ to obtain the $B_k(n)$ defined in (2.5.12). More precisely this time we define

$$\begin{aligned}B_k(n) &:= \int_{-b}^{+\infty} \mathrm{d}z\, e^{-zn}(z+b)^{\beta+\gamma k}\log(z+b)^c \\ &= e^{nb}\sum_{i=0}^{c}\binom{c}{i}(-1)^i\Gamma^{(c-i)}(\beta+\gamma k+1)\frac{\log(n)^i}{n^{\beta+\gamma k+1}}.\end{aligned} \tag{2.5.18}$$

This means in particular that when we try to bound $|\mathcal{R}_m(n)|$ as in (2.5.13), we end up with

$$\forall \eta,\ \exists \epsilon_\eta, M_\eta > 0, \quad |\mathcal{R}_m(n)| \le \epsilon_\eta B_{m+1}(n) + \frac{M_\eta}{n-c}e^{-(n-c)\eta}. \tag{2.5.19}$$

Then

$$\forall \epsilon,\ \exists n_\epsilon > 0,\ \forall n \ge n_\epsilon, \quad |R_m(n)| \le \epsilon\frac{A_m e^{nb}\log(n)^c}{n^{\beta+\gamma(m+1)}} \tag{2.5.20}$$

where

$$A_m = \max_{0\le i\le c}\left(\left|\binom{c}{i}\Gamma^{(c-i)}(\beta+\gamma(m+1)+1)\right|\right) + 1.$$

Thus we get

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{e^{nb}\log(n)^c}{n^{\beta+\gamma(m+1)}}\right)$$

which proves the extended Watson's lemma. □

While working for functions that are divergent but integrable at $x = 1$, the GSA

method in Section 2.5.2 will not work for cases when the geometric series $\frac{1}{x-\frac{1}{a}}$ is such that $a \geq 1$, because it is then divergent at $\frac{1}{a} \in (0;1]$. This singularity is actually only apparent because the full integrand of the Mellin integral is actually[59]:

$$\frac{(ax)^n - 1}{x - \frac{1}{a}} f(x) \underset{x \to \frac{1}{a}}{=} a\, n f\left(\frac{1}{a}\right) + \mathcal{O}\left(x - \frac{1}{a}\right).$$

Nevertheless, this means that since the $(ax)^n - 1$ is exactly the factor compensating for the singularity, we cannot split the integral to compute the constant separately. The idea is therefore to perform a change of variables[60] to bring the integral to a particular form matching Watson's Lemma 2.5.2 or its extended version Lemma 2.5.3, and then to use a regulation that allows us to take care of the divergence and extract the constant part from the integral. The specific hypotheses on f are rather general for an expansion up to order p:

- f is integrable on $[0;1]$ or integrable as a Cauchy principal value[a].
- f is smooth enough, i.e., at least p-times differentiable on $(0;1)$.
- $a \geq 1$.

[a] In particular, all integrals of f over $[0;1]$ have to be seen as Cauchy principal value integrals if needed and by abuse of notation we will still write "$\int \mathrm{d}x\, f(x) = \fint \mathrm{d}x\, f(x)$.

We start by dealing with the case $a \neq 1$, as it presents a particularity for which we need to do some extra steps. Once again using the convention (2.5.4), we do the following:

1. We perform the substitution:

$$x = \frac{e^{-z}}{a}, \quad \mathrm{d}x = -\frac{1}{a} e^{-z} \mathrm{d}z$$

which is indeed a $\mathcal{C}^\infty$-diffeomorphism. This gives:

$$\begin{aligned}\int_0^1 \mathrm{d}x\, \frac{(ax)^n - 1}{x - \frac{1}{a}} f(x) &= -\frac{1}{a} \int_{+\infty}^{-\log a} \mathrm{d}z\, e^{-z} a \frac{e^{-zn} - 1}{e^{-z} - 1} f\left(\frac{e^{-z}}{a}\right) \\ &= \int_{-\log a}^{+\infty} \mathrm{d}z\, (e^{-zn} - 1) \underbrace{\frac{e^{-z}}{e^{-z} - 1} f\left(\frac{e^{-z}}{a}\right)}_{=:g(z)}.\end{aligned}$$

[59] Provided that f, using the notations of (2.5.4), is regular at $x = \frac{1}{a}$.
[60] We will use interchangeably "change of variable" and "substitution" in the following.

Due to the change of variables, the (possibly) singular points were shifted in the following way:

$$\begin{array}{rcl} x=0 & \to & z=+\infty \\ x=\frac{1}{a} & \to & z=0 \\ x=1 & \to & z=-\log a. \end{array}$$

To justify the proposed method below, we expand the function g defined above around $z=0$

$$g(z) \underset{z\to 0}{=} \frac{\alpha_{-1}}{z}+\alpha_0+\mathcal{O}(z)$$

where $\alpha_{-1},\alpha_0 \in \mathbb{C}$. At this point, given that g is divergent around $z=0$ while the whole integrand $(e^{-zn}-1)g(z)$ has only an apparent singularity, the idea is to eliminate it. Note that we get rid of the $\frac{1}{z}$ by integrating over g:

$$\int_{-\log a}^{z} \mathrm{d}w\, g(w) \underset{z\to 0}{=} C+\alpha_{-1}\log z+\alpha_0 z+\mathcal{O}(z^2)$$

where C will be the integration constant, and the lower bound of the integral will be motivated below. Since $\log z$ is integrable at $z=0$, we handled the non-integrable point. We note here that this process is possible since we consider $a\neq 1$ and therefore the function is regular enough at the lower bound of the integral $-\log a$.

2. Motivated by the above considerations, we define[61]

$$\begin{array}{c} u'(z)=g(z) \\ v(z)=e^{-zn}-1 \end{array} \to \begin{array}{c} u(z)=\int_{-\log a}^{z}\mathrm{d}w\, g(w) \\ v'(z)=-ne^{-zn} \end{array}$$

and perform an integration by parts (IBP):

$$\begin{aligned} \int_{-\log a}^{+\infty}\mathrm{d}z\,(e^{-zn}-1)g(z) = &\left[(e^{-zn}-1)\int_{-\log a}^{z}\mathrm{d}w\, g(w)\right]_{-\log a}^{+\infty} \\ &+n\int_{-\log a}^{+\infty}\mathrm{d}z\, e^{-zn}\int_{-\log a}^{z}\mathrm{d}w\, g(w). \end{aligned}$$

3. Next, we simplify the constant terms appearing in the expression above. We have two constants:

$$C_\infty := \lim_{z\to+\infty}(e^{-zn}-1)\int_{-\log a}^{z}\mathrm{d}w\, g(w) = -\int_{-\log a}^{+\infty}\mathrm{d}w\, g(w), \tag{2.5.21}$$

$$C_{-\log a} := \lim_{z\to-\log a}(e^{-zn}-1)\int_{-\log a}^{z}\mathrm{d}w\, g(w) = 0. \tag{2.5.22}$$

[61] We comment more precisely on the primitive of $g(z)$ in the Remark 2.5.8 below.

By construction the primitive of g vanishes at $-\log a$ which implies that $C_{-\log a} = 0$. On the other hand, since $\frac{f(x)}{x-\frac{1}{a}}$ is integrable at $x = 0$ and $x = 1$, $g(z)$ at $z = -\log a$ and $z = +\infty$, it means exactly that the value of the integral $\int_{-\log a}^{+\infty} \mathrm{d}w\, g(w)$ is finite. We also note here that while the constant is in general not simplifiable to an expression involving well-known constants, it is well defined as a symbolic representation which can be used to compute numerical approximation to arbitrary precision.

Remark 2.5.3. *While we described how we obtain the constant part of the integration by parts, when $a > 1$ it will actually be part of the subdominant asymptotics and, as per Remark 2.5.1, it can simply be thrown away. It will be kept only if the* `ExtendedAsymptotics` *flag is activated and/or if*[62] *$a \leq 1$.*

Remark 2.5.4. *If the function $g(w)$ has poles on $(-\log a; +\infty)$, we will implicitely use the Cauchy principle value, as defined in Section 2.4.3, both for its primitive $\int_{-\log a}^{z} \mathrm{d}w\, g(w)$ and for the constant C_∞.*

4. We expand $\int_{-\log a}^{z} \mathrm{d}w\, g(w)$ up to order p around $z = -\log a$, because we made a change of variables of the form $x = \frac{e^{-z}}{a}$:

$$\int_{-\log a}^{z} \mathrm{d}w\, g(w) \underset{z\to-\log a}{\sim} \sum_{\alpha\leq p} g_\alpha (z+\log a)^\alpha, \quad \alpha \in \frac{1}{2}\mathbb{Z}_{\geq -1},\ g_\alpha \in \mathbb{R}.$$

Note that α can take half-integer values, because we have a root-valued alphabet. Therefore the expansion we get is actually a Puiseux expansion. We also note that $(z+\log a)^\alpha$ is still integrable for $\alpha = \frac{1}{2}$, but not longer for $\alpha = -1$.

Remark 2.5.5. *In general, for the function g it is often difficult to find a closed form of the definite integral $\int_{-\log a}^{z} \mathrm{d}w\, g(w)$, in particular if g contains GRPLs or GCPLs. Nevertheless, note that for the final result we are only interested in its series expansion around $z = -\log a$. Because we consider here the case $a \neq 1$ the function $g(w)$ is regular enough at $-\log a$ and then we can proceed as follows:*

- *Expand $g(w)$ around $w = -\log a$.*
- *Integrate the expansion from $w = -\log a$ to $w = z$, which then fixes the integration constant to 0.*

From a technical point of view, `RICA` *always tries to compute explicitely the*

[62] The validity of the regime $a < 1$ will be described in the Section 2.5.9 below.

integral of g and then only resorts to the integration of series expansion, because the closed form is helpful in computing the expression of the constant C_∞. When this is not possible, the constants will be returned as an inactivated integral.

5. Finally, using the extended Watson's Lemma 2.5.3, we integrate term by term, using the expansion above:

$$\begin{aligned}\int_{-\log a}^{+\infty} \mathrm{d}z\, n e^{-zn} \sum_{\alpha \leq p} g_\alpha (z+\log a)^\alpha &= \sum_{\alpha \leq p} g_\alpha \int_{-\log a}^{+\infty} \mathrm{d}z\, n e^{-zn} (z+\log a)^\alpha \\ &= \sum_{\alpha \leq p} \frac{h_\alpha}{n^\alpha}, \quad h_\alpha \in \mathbb{R} \end{aligned} \tag{2.5.23}$$

and putting everything together we get the full expansion:

$$\tilde{M}_a[f(x)](n) \underset{n\to+\infty}{=} C_\infty + \sum_{\alpha \leq p} \frac{h_\alpha}{n^\alpha} + \mathcal{O}\left(\frac{1}{n^{p+1}}\right).$$

Remark 2.5.6. *As stated at the end of Step 1 where we motivate the IBP, it turns out that when $a = 1$ the integral we consider takes the following form:*

$$I(n) = \int_0^{+\infty} \mathrm{d}z\, (e^{-zn} - 1) g(z) \text{ with } g(z) = \frac{e^{-z}}{e^{-z}-1} f(e^{-z}) \underset{z\to 0}{\sim} \frac{\alpha}{z}. \tag{2.5.24}$$

Because we cannot choose the lowest integration bound to be $-\log a = 0$, there are two possible strategies. The simplest one is to choose a different lower bound. For instance we can take 1 as the lower bound as long as there are no singularities for g here. This means in the steps $2 \to 5$ we replace

$$\int_{-\log a}^{z} \mathrm{d}w\, g(w) \to \int_1^z \mathrm{d}w\, g(w).$$

Nevertheless, this method still suffers from one particular caveat, in particular if we either cannot find a closed form at all for the definite or indefinite integral of g, or if $g(z)$ itself is such that we only know its expansion around[63] *$z = 1$. In this case, the idea is to perform the integration by parts in such a way that we regularize the primitive of g around 0. We define for that the following primitive of g, with $\epsilon > 0$ for the moment:*

$$G(z,\epsilon) := \int_\epsilon^z \mathrm{d}w\, g(w) = \underbrace{\int_\epsilon^z \mathrm{d}w \left(g(w) - \frac{\alpha}{w}\right)}_{=:\tilde{G}(z,\epsilon)} + \alpha \log z - \alpha \log \epsilon. \tag{2.5.25}$$

[63] In particular if g is a GCPL, such cases are described in Section 2.5.4.

We also denote

$$\bar{g}(w) := g(w) - \frac{\alpha}{w}.$$

Now the IBP from Step 2 writes as:

$$\int_0^{+\infty} \mathrm{d}z\,(e^{-zn}-1)g(z) = \Big[(e^{-zn}-1)\left(\bar{G}(z,\epsilon) - \alpha\log\epsilon\right)\Big]_0^{+\infty} + n\int_0^{+\infty}\mathrm{d}z\,e^{-zn}\left(\bar{G}(z,\epsilon)-\alpha\log\epsilon\right). \tag{2.5.26}$$

When we compute the constants like in Step 3, because by construction $\bar{G}(z,\epsilon)$ is behaving at worst logarithmically at 0, once again we have $C_0 = 0$, but now C_∞ takes a different form:

$$C_\infty(\epsilon) = \lim_{z\to+\infty}\Big[(e^{-zn}-1)\left(\bar{G}(z,\epsilon)-\alpha\log\epsilon\right)\Big] = \alpha\log\epsilon - \bar{C}_\infty(\epsilon) \tag{2.5.27}$$

where we defined

$$\bar{C}_\infty(\epsilon) := \lim_{z\to+\infty}\bar{G}(z,\epsilon). \tag{2.5.28}$$

We now simplify (2.5.26) further:

$$\begin{aligned}\int_0^{+\infty}\mathrm{d}z\,(e^{-zn}-1)g(z) &= \alpha\log\epsilon - \bar{C}_\infty(\epsilon) + n\int_0^{+\infty}\mathrm{d}z\,e^{-zn}\left(\bar{G}(z,\epsilon)-\alpha\log\epsilon\right)\\ &\xrightarrow{(2.5.7)} = -\bar{C}_\infty(\epsilon) + n\int_0^{+\infty}\mathrm{d}z\,e^{-zn}\left(\int_\epsilon^z \mathrm{d}w\,\bar{g}(w) + \alpha\log z\right) + \cancel{\alpha\log\epsilon - \alpha\log\epsilon}.\end{aligned} \tag{2.5.29}$$

Because the divergent ϵ part is gone, we can send $\epsilon\to 0$ and we obtain:

$$\int_0^{+\infty}\mathrm{d}z\,(e^{-zn}-1)g(z) = -\bar{C}_\infty(0) - \alpha\log n - \alpha\gamma + n\int_0^{+\infty}\mathrm{d}z\,e^{-zn}\int_0^z \mathrm{d}w\,\bar{g}(w) \tag{2.5.30}$$

where we used from (2.5.7) that $n\int_0^{+\infty}\mathrm{d}z\,e^{-zn}\log(z) = -\gamma - \log n$, with γ being the Euler-Mascheroni constant; we use $\log z = \int_1^z \mathrm{d}w\,\frac{1}{w}$ to write

$$\begin{aligned}\bar{C}_\infty(0) &= \lim_{z\to+\infty}\left(\int_0^z \mathrm{d}w\,\left[g(w)-\frac{\alpha}{w}\right] + \alpha\int_1^z \mathrm{d}w\,\frac{1}{w}\right)\\ &= \lim_{z\to+\infty}\left(\int_0^1 \mathrm{d}w\,\left[g(w)-\frac{\alpha}{w}\right] + \int_1^z \mathrm{d}w\,\left[g(w)-\frac{\alpha}{w}\right] + \alpha\int_1^z \mathrm{d}w\,\frac{1}{w}\right)\\ &= \lim_{z\to+\infty}\left(\int_0^1 \mathrm{d}w\,\left[g(w)-\frac{\alpha}{w}\right] + \int_1^z \mathrm{d}w\,g(w)\right)\\ &= \lim_{z\to+\infty}\left(\int_0^z \mathrm{d}w\,\Theta(1-w)\left[g(w)-\frac{\alpha}{w}\right] + \int_0^z \mathrm{d}w\,\Theta(w-1)g(w)\right)\end{aligned}$$

$$= \lim_{z\to+\infty} \int_0^z \mathrm{d}w \left[\Theta(1-w)\left(g(w) - \frac{\alpha}{w}\right) + \Theta(w-1)g(w)\right]$$
$$= \lim_{z\to+\infty} \int_0^z \mathrm{d}w \left(g(w) - \alpha\cdot\frac{\Theta(1-w)}{w}\right),$$

so finally $C_\infty(0)$ *has the following compact form:*

$$\bar{C}_\infty(0) = \int_0^{+\infty} \mathrm{d}z \left(g(z) - \alpha\cdot\frac{\Theta(1-z)}{z}\right). \tag{2.5.31}$$

After that, we can proceed in the usual way from Step 4 onwards, except that instead of expanding $\int_{-\log a}^z \mathrm{d}w\, g(w)$ *we expand* $\int_0^z \mathrm{d}w\, \bar{g}(w)$*. Following Remark 2.5.5, because* $\bar{g}$ *is obtained by regularising* g *around* $z=0$*, we can again apply the same technique to get the expansion of* $\int_0^z \mathrm{d}w\, \bar{g}(w)$*, i.e., expand* $\bar{g}(w)$ *around* $w=0$ *and then integrate from* $w=0$ *to* $w=z$*.*

Remark 2.5.7. *We derived the most general formula in Lemma 2.5.1 to fit all possible cases that can happen in our setup as well as to include the cases given in (2.5.6). In particular, if we apply it to the situation given in (2.5.23) we get the following explicit value for the* $h_{\alpha,\beta}$ *terms:*

$$h_\alpha = g_\alpha a^n \frac{\Gamma(\alpha+1)}{n^{\alpha+1}}.$$

Remark 2.5.8. *Let us come back to Step 2 above where we perform an IBP. We have to find a primitive of* $g(z)$ *and in order to cancel one constant of the IBP we fix the primitive to be 0 at* $z=-\log a$*. Nevertheless, we have to remember that* $g(z)$ *is obtained through a change of variable from* f*, which has to be integrable only weakly, i.e., up to integration using the Cauchy principal value. It means in particular that we accept (symmetric) singurities* $z_s \in (-\log a; +\infty)$*, and when we define the primitive, we have to perform it cautiously. To understand this more precisely, let us consider a simpler example. Suppose we are given the function* $g : z \mapsto \frac{1}{z}$ *and we want to compute* $\int_{-1}^z \mathrm{d}w\, g(w)$ *where* $z \in [-1; +\infty)$*. Now the problem is that the primitive takes a different form depending on the value of* z*, i.e.,*

$$\int_{-1}^z \mathrm{d}w\, g(w) = \begin{cases} \log(-z) & \text{if} \quad z<0 \\ \log(z) & \text{if} \quad z>0 \end{cases}$$

where if $z>0$ *we have to use the Cauchy principal value to cancel out the pole at* $z=0$*. Hence the primitive defined for any value of* z *except for* $z=0$*. In this particular case, using the absolute value we can even summarise the primitive as* $\log|z|$*. Performing a primitive without clear specification for the range of the parameters, Mathematica will usually return only a partial primitive being true*

only in one of the regions. Therefore, we have implemented in `RICA` *an advanced integration system specially taylored for the integration by parts methods that does the following:*

1. *Given a specific integrand and e^{-a}, where a is the lower bound of the integral*[64]*, we find the integrand singularities $z_{s_1}, \ldots, z_{s_p} \in (-\log a; +\infty)$, $p \in \mathbb{N}$ using Mathematica's* `FunctionSingularities` *— in the case of GRPLs or GCPLs, we can simply look at the word over which they are defined, i.e., the singularities of the GRPL/GCPL will stem at most from all of the singularities of the letters in their words.*

2. *We sort the singularities in increasing order*[65] *and define the associated intervals*
$$\mathcal{I}_0 = (-\log a; z_{s_1}), \mathcal{I}_1 = (z_{s_1}; z_{s_2}), \ldots, \mathcal{I}_p = (z_{s_{p-1}}; z_{s_p}).$$

3. *We integrate g, specifiying that $z \in \mathcal{I}_i$ for $i \in \{0, \ldots, p\}$.*

4. *We return the full primitive defined piecewise over each interval $\mathcal{I}_i$, that we can also write as a unique expression using Heaviside functions.*

Example 2.5.3. *As a simple example for the remark above, let us suppose we need to compute the following primitive:*

$$\int_0^z \mathrm{d}x \, \frac{1}{9x^2 - 9x + 2}.$$

We find the two singularities of the integrand, which are at $x = \frac{1}{3}$ and $x = \frac{2}{3}$, which means there are three integration intervals

$$\mathcal{I}_1 = \left[0; \frac{1}{3}\right), \mathcal{I}_2 = \left(\frac{1}{3}; \frac{2}{3}\right), \mathcal{I}_3 = \left(\frac{2}{3}; +\infty\right).$$

Now, we have to integrate over the different intervals, i.e.:

- *When $z \in \mathcal{I}_1$ we have:*
$$\int_0^z \mathrm{d}x \, \frac{1}{9x^2 - 9x + 2} = \frac{1}{3} \log\left(\frac{2 - 3x}{2 - 6x}\right).$$

- *When $z \in \mathcal{I}_2$ we have:*
$$\int_0^z \mathrm{d}x \, \frac{1}{9x^2 - 9x + 2} = -\frac{1}{3} \log\left(\frac{2 - 6x}{3x - 2}\right).$$

[64] The e^{-a} comes from the fact that after performing the change of variables, the lower bound becomes $-\log a$.

[65] For simplicity, we suppose that $z_{s_1}, \ldots, z_{s_p}$ are already sorted.

- *When $z \in \mathcal{I}_3$ we have:*

$$⨍_0^z \mathrm{d}x \, \frac{1}{9x^2 - 9x + 2} = \frac{1}{3} \log\left(\frac{2-3x}{2-6x}\right).$$

Therefore the full primitive, defined for $z \in \mathbb{R}_+$, that vanishes at $z = 0$ is:

$$⨍_0^z \mathrm{d}x \, \frac{1}{9x^2 - 9x + 2} = \begin{cases} \frac{1}{3} \log\left(\frac{2-3x}{2-6x}\right) & \text{if} \quad 0 < x < \frac{1}{3} \vee x > \frac{2}{3} \\ -\frac{1}{3} \log\left(\frac{2-6x}{3x-2}\right) & \text{if} \quad \frac{1}{3} < x < \frac{2}{3} \end{cases}.$$

Example 2.5.4. *Let us now apply the "COV with IBP" method to (2.3.19) from Example 2.3.1, that is to say:*

$$\sum_{i=1}^{n} \binom{2i}{i} S_2(i) = -\frac{1}{\pi} \int_0^1 \mathrm{d}x \, \frac{(4x)^n - 1}{x - \frac{1}{4}} \sqrt{\frac{x}{1-x}} \left(\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2} - \zeta_2\right).$$

First, we simplify the polylogarithm using its explicit value:

$$\mathrm{H}^*_{\mathrm{w}_1}(x) = \arccos(2x - 1)$$

and define the function:

$$f_3(x) := \frac{1}{x - \frac{1}{4}} \sqrt{\frac{x}{1-x}} \left(\frac{\mathrm{H}^*_{\mathrm{w}_1}(x)^2}{2} - \zeta_2\right)$$

that is clearly divergent around $x = \frac{1}{4}$:

$$f(x) \underset{x \to \frac{1}{4}}{=} \frac{\zeta_2}{3\sqrt{3}\left(x - \frac{1}{4}\right)} + \mathcal{O}(1).$$

We change variables by $x \to \frac{e^{-z}}{4}$:

$$g_3(z) := \frac{e^{-z}}{4} f_3\left(\frac{e^{-z}}{4}\right) = \frac{2\zeta_2 - \arccos\left(\frac{e^{-z}}{2} - 1\right)^2}{2\left(e^z - 1\right)\sqrt{4e^z - 1}}.$$

***Note**: We failed to compute explicitely a closed form of the integral of $g_3(z)$. But if we could, we should take care of its singularities as we explained in Remark 2.5.8 above. In this case the singularities are $z = 0$, $z = 1$ and $z = -2\log 2$. Thus we would need to integrate $g_3(z)$ as a Cauchy principal value on the intervals*

$$(-2\log 2; 0),\ (0; 1),\ (1; +\infty).$$

In particular, we compute the constant C_∞ that appears in the IBP, i.e.,

$$C_\infty = -\fint_{-\log 4}^{+\infty} \mathrm{d}w\, g_3(w)$$

and we have to take it as a Cauchy principal value because we cross the singularity[66] *at $z = 0$. Here we proceed differently.*

To expand $\int_{-\log 4}^{z} \mathrm{d}w\, g_3(w)$ around $z = -\log 4$, we simply have to expand g around $w = -\log 4$ and then integrate this expansion from $w = -\log 4$ to $w = z$, which yields[67]

$$\begin{aligned}
\int_{-\log 4}^{z} \mathrm{d}w\, g_3(w) \underset{z\to -\log 4}{=} \sqrt{2\log 2 + z}\Bigg[&\log^4 2\left(\frac{578}{567} - \frac{82399\zeta_2}{174960}\right) + \log^3 2\left(\frac{388}{315} - \frac{485\zeta_2}{1134}\right)\\
&+ \log^2 2\left(-\frac{59\zeta_2}{135} - \frac{16}{45}\right) + \log 2\left(\frac{32}{9} - \frac{4\zeta_2}{27}\right) - \frac{8\zeta_2}{3}\Bigg]\\
+ z\sqrt{2\log 2 + z}\Bigg[&\log^3 2\left(\frac{1156}{567} - \frac{82399\zeta_2}{87480}\right) + \log^2 2\left(\frac{194}{105} - \frac{485\zeta_2}{756}\right)\\
&+ \log 2\left(-\frac{59\zeta_2}{135} - \frac{16}{45}\right) - \frac{2\zeta_2}{27} + \frac{16}{9}\Bigg]\\
+ z^2\sqrt{2\log 2 + z}\Bigg[&\log^2 2\left(\frac{289}{189} - \frac{82399\zeta_2}{116640}\right) + \log 2\left(\frac{97}{105} - \frac{485\zeta_2}{1512}\right)\\
&- \frac{59\zeta_2}{540} - \frac{4}{45}\Bigg]\\
+ z^3\sqrt{2\log 2 + z}\Bigg[&\log 2\left(\frac{289}{567} - \frac{82399\zeta_2}{349920}\right) - \frac{485\zeta_2}{9072} + \frac{97}{630}\Bigg]\\
+ z^4\sqrt{2\log 2 + z}\Bigg[&\frac{289}{4536} - \frac{82399\zeta_2}{2799360}\Bigg] + \mathcal{O}(z^5\sqrt{2\log 2 + z}).
\end{aligned}$$

Finally, we perform the integral with respect to z, add the constant and multiply back by the $-\frac{1}{\pi}$ factor to get the desired result:

$$\begin{aligned}
\sum_{i=1}^{n}\binom{2i}{i}S_2(i) &\underset{n\to+\infty}{=} -\frac{1}{\pi}\left(C_\infty + n\int_{-\log 4}^{+\infty}\mathrm{d}z\, e^{-zn}\int_{-\log 4}^{z}\mathrm{d}w\, g_3(w)\right)\\
&= -\frac{C_\infty}{\pi} + 4^n\left(\frac{2\pi^2}{9\sqrt{n}} - \frac{4}{3\sqrt{\pi}n^{3/2}} + \frac{1}{6\sqrt{\pi}n^{5/2}} - \frac{97}{\sqrt{\pi}96n^{7/2}}\right.\\
&\quad\left. - \frac{1445}{768\sqrt{\pi}n^{9/2}} + \frac{\pi^2}{108n^{3/2}} + \frac{59\pi^2}{1728n^{5/2}} + \frac{2425\pi^2}{41472n^{7/2}} + \frac{576793\pi^2}{3981312n^{9/2}}\right)\\
&\quad + \mathcal{O}\left(\frac{4^n}{n^{11/2}}\right).
\end{aligned}$$

[66] $z = -\log 4$ is the lower bound and an integrable singulairity

[67] The problem of Cauchy principal values in the integration is here irrelevant, as by expanding g around $z = -\log 4$, we only get an approximation of the integral around $z = -\log 4$, which turns out to be enough for what we want to do.

Once again, this can be computed automatically using the `AsymtoticsMellint` *command of* `RICA`*:*

```
In[22]:= AsymtoticsMellint[-(Mellin[(4x)^n - 1, Sqrt[x/(1-x)] (1/2 Hwb({fw1},x)^2 - z2)] / (x - 1/4)) / Pi, x, n, 5,
  ExtendedAsymptotics -> True, AdvancedIntegration -> True]
```

$$\text{Out[22]=}\quad -\frac{1}{\pi}\left(\fint_{-2\log 2}^{+\infty}\mathrm{dw}\,\frac{\mathrm{e}^{-\mathrm{w}}\sqrt{\frac{\mathrm{e}^{-\mathrm{w}}}{1-\frac{\mathrm{e}^{-\mathrm{w}}}{4}}}\left(\frac{1}{2}\mathrm{GL}\left[\left\{\frac{1}{\sqrt{(1-\mathrm{VarGL})\mathrm{VarGL}}}\right\},1-\frac{\mathrm{e}^{-\mathrm{w}}}{4}\right]^2-\frac{\pi^2}{6}\right)}{8\left(\frac{\mathrm{e}^{-\mathrm{w}}}{4}-\frac{1}{4}\right)}\right.$$
$$-\frac{(\pi-12)(12+\pi)\sqrt{\pi}2^{2\mathrm{n}-2}}{27\mathrm{n}^{3/2}}-\frac{(288+59\pi^2)\sqrt{\pi}2^{2\mathrm{n}-6}}{27\mathrm{n}^{5/2}}-\frac{97\,(25\pi^2-432)\sqrt{\pi}2^{2\mathrm{n}-9}}{81\mathrm{n}^{7/2}}$$
$$\left.-\frac{17\,(33929\pi^2-440640)\sqrt{\pi}2^{2\mathrm{n}-14}}{243\mathrm{n}^{9/2}}-\frac{\pi^{5/2}2^{2\mathrm{n}+1}}{9\sqrt{\mathrm{n}}}\right)$$

where we denote as usual the Cauchy principal value of $f(x)$ *with* $\fint \mathrm{d}x\, f(x)$*, and where we set the* `ExtendedAsymptotics` *flag to True to also obtain the constant. Setting the flag to False, this subdominant part would be ommitted.*

Remark 2.5.9. *We have dealt above with the case when* $a > 1$*, i.e., if the integrand has an apparent singularity at* $x = \frac{1}{a}$*. As we have already realised, because of the divergence at* $x = \frac{1}{a}$ *we had to perform an IBP, but when* $a < 1$ *the change of variable method still works and is actually simpler, as in this case, we simply apply the COV method after splitting the integral. In principle, in the range* $-1 < a < 1$ *the FSA or GSA methods are applicable, but depending on the regularity of* $\tilde{f}(x) = \frac{f(x)}{x-\frac{1}{a}}$ *around* $x = 1$ *(for FSA) and whether we manage to compute the Mellin transform of* $f(x)$ *(for GSA), we might have to fall back again on the COV method — technically, this is actually what* `RICA` *is doing, in particular setting a time limit on the FSA and GSA methods. As an additional note, this method will also work for the case when* $a = 1$ *and* $f(x) \underset{x\to 1}{\sim} (x-1)^\alpha$ *with* $\alpha \geq \frac{1}{2}$*, i.e., the* n*-independent part of the integrand is integrable at*[68] $x = 1$*. In particular, this can happen when* $f(x) \propto \mathrm{H}^*_{\{\mathrm{f_i}\}}(x)$*, where* f_i *are some letters of the root-valued alphabet, as by definition* $\mathrm{H}^*_{\{\mathrm{f_i}\}}$ *should vanish at 1.*
Summarising, we obtain the following simplified version for a Mellin integral of the form:

$$\int_0^1 \mathrm{d}x\,\frac{(-ax)^n-1}{x+\frac{1}{a}}f(x),\ a>1.$$

[68] In addition to being integrable at all the other points of $[0;1)$.

1. *Split the Mellin integral, factor out the $(-a)^n$, and compute the constant part:*

$$\int_0^1 \mathrm{d}x\, \frac{(-ax)^n - 1}{x + \frac{1}{a}} f(x) = (-a)^n \mathbf{M}\left[\frac{f(x)}{x+\frac{1}{a}}\right](n) - \underbrace{\mathbf{M}\left[\frac{f(x)}{x+\frac{1}{a}}\right](0)}_{=:C}.$$

2. *In $\mathbf{M}\left[\frac{f(x)}{x+\frac{1}{a}}\right](n)$, we make the same change of variable as in [16]:*

$$x = e^{-z}\, \mathrm{d}x = -e^{-z}\mathrm{d}z$$

and end up with:

$$\mathbf{M}\left[\frac{f(x)}{x+\frac{1}{a}}\right](n) = (-a)^n \int_0^{+\infty} \mathrm{d}z\, e^{-zn} \underbrace{\frac{e^{-z}}{e^{-z}+1} f(e^{-z})}_{=:g(z)}. \tag{2.5.32}$$

3. *We expand $g(z)$ around $z = 0$ up to the order p:*

$$g(z) \underset{z\to 0}{=} \sum_{\alpha \le p} g_\alpha z^\alpha + \mathcal{O}(z^{\alpha+1}), \quad \alpha \in \frac{1}{2}\mathbb{Z}_{\ge -1} - \frac{1}{2},\ g_\alpha \in \mathbb{R}.$$

4. *Finally, using Watson's Lemma 2.5.2, the expansion of g in (2.5.32), integrating, and adding the constant and the $(-a)^n$ coefficient back,*

$$\begin{aligned}\tilde{\mathbf{M}}_{-a}\left[f(x)\right](n) &\underset{n\to+\infty}{=} (-a)^n \sum_{\alpha\le p} \int_0^{+\infty} \mathrm{d}z\, e^{-zn} g_\alpha z^\alpha - C \\ &= (-a)^n \sum_{\alpha \le p} \frac{h_\alpha}{n^\alpha} - C, \quad h_\alpha \in \mathbb{R}.\end{aligned}$$

Example 2.5.5. *We illustrate this method looking at the following sum*[69]*:*

$$\sum_{i=1}^{n} \frac{4^i}{i^3} \frac{1}{\binom{2i}{i}}.$$

We first use `RICA`*'s* `SumToMellin` *function to compute its Mellin inverse*

```
In[23]:= SumToMellin[GS[{4^VarGL / (VarGL^3 Binomial[2 VarGL, VarGL])}, n], C, x, ToGLbBasis -> False]
```

[69] Appearing for instant in the example (5.35) from [16].

Out[23]= $\left\{Mellin\left[\mathtt{x^n}-1,\frac{Hwb[\{f0,fw3\},\mathtt{x})}{\mathtt{x}-1}\right],\{\}\right\}$

If we write the Mellin transform explicitely we get

$$\sum_{i=1}^{n}\frac{4^i}{i^3}\frac{1}{\binom{2i}{i}}=\int_0^1 \mathrm{d}x\,\frac{x^n-1}{x-1}\mathrm{H}^*_{0,\mathrm{w}_3}(x).$$

One can check easily that $\frac{\mathrm{H}^*_{0,\mathrm{w}_3}(x)}{x-1}\underset{x\to 1}{=}\mathcal{O}(1)$ *so it is integrable and the integral can be split. Computing the constant part, we get:*

$$-\int_0^1 \mathrm{d}x\,\frac{\mathrm{H}^*_{0,\mathrm{w}_3}(x)}{x-1}=6\zeta_2\log 2-\frac{7}{2}\zeta_3.$$

Now, defining

$$f_4(x):=\frac{\mathrm{H}^*_{0,\mathrm{w}_3}(x)}{x-1}$$

and changing the variables to $x=e^{-z}$*, defining* $g_4(z):=e^{-z}f_4(e^{-z})$*, and expanding it around* $z=0$ *up to order* p*, we get:*

$$g_4(z)\underset{z\to 0}{=}-\frac{4\sqrt{z}}{3}+\frac{3z^{3/2}}{5}-\frac{199z^{5/2}}{2520}-\frac{29z^{7/2}}{6048}+\frac{8909z^{9/2}}{5322240}+\mathcal{O}\left(z^{9/2}\right).$$

Finally, integrating out the z *dependence and adding the constant part, we get the following expansion:*

$$\sum_{i=1}^{n}\frac{4^i}{i^3}\sum_{j=1}^{i}\frac{1}{\binom{2i}{i}}\underset{n\to+\infty}{=}6\log 2\zeta_2-\frac{7}{2}\zeta_3-\frac{2\sqrt{\pi}}{3n^{3/2}}+\frac{9\sqrt{\pi}}{20n^{5/2}}-\frac{199\sqrt{\pi}}{1344n^{7/2}}-\frac{145\sqrt{\pi}}{4608n^{9/2}}+\mathcal{O}\left(\frac{1}{n^{11/2}}\right).$$

Once again, using the function `AsymptoticsMellint`*, we can compute directly this expansion:*

In[24]:= **AsymptoticsMellint**$\left[\mathbf{Mellin}\left[\mathbf{x^n-1},\frac{\mathbf{Hwb(\{f0,fw3\},x)}}{\mathbf{x-1}}\right],\mathbf{x},\mathbf{n},\mathbf{5}\right]$

Out[24]= $\frac{1}{2}\left(2\pi^2\ln 2-7z3\right)+\frac{9\sqrt{\pi}}{20\mathtt{n}^{5/2}}-\frac{2\sqrt{\pi}}{3\mathtt{n}^{3/2}}-\frac{199\sqrt{\pi}}{1344\mathtt{n}^{7/2}}-\frac{145\sqrt{\pi}}{4608\mathtt{n}^{9/2}}$

2.5.4. Asymptotics and GCPL functions

Until now, the different methods we have described to peform asymptotics of binomial sums required two main ingredients: first, to be able to use our recursive algorithm to compute a Mellin inverse representation of the sum, and second, to perform an asymptotic expansion of the said integral. One of the major cornerstones of the recursive Mellin inversion method, as described in Section 2.3, relies

on Mellin convolution computations that we performed either explicitely or relying on the set of Theorems 2.3.1 – 2.3.2 elaborated in [16]. When we do not manage to carry out this explicit calculation, we get stuck. In order to overcome this hurdle, we will utilise the following two results:

1. In Section 2.3.4, we introduced a new class of functions called the generalized convolution polylogarithms (GCPL) which aimed at representing in a compact way nested convolutions, involving in particular GRPLs and other classes of functions defined in Section 2.1.2. With this first step, it became possible to obtain the result of a Mellin inversion as a function of GCPLs functions, even if we considered extended binomial sums for which the standard method did not apply.

Example 2.5.6. *For example, for the sum below we have:*

$$\begin{aligned}\mathcal{BS}_{((1,1,0),(1,1,0))}(n) =& \sum_{i=1}^{n}\binom{2i}{i}\sum_{j=1}^{i}\binom{2j}{j}\\ &= \frac{1}{\pi^2}\tilde{\mathbf{M}}_{16}\left[\frac{x}{x-\frac{1}{16}}\mathcal{H}^{*}_{\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{\sqrt{\frac{\lambda}{1-\lambda}}}{\lambda-\frac{1}{4}};\emptyset\right]\right)}(x)\right](n)\\ &\quad -\frac{4C}{\pi^2}\tilde{\mathbf{M}}_{4}\left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)}\right](n)\end{aligned}$$

where

$$C = \mathrm{H}^{*}_{\frac{\sqrt{\frac{\lambda}{1-\lambda}}}{\lambda-\frac{1}{4}}}(0).$$

As opposed to the case presented in Example 2.3.8, here the dominating term of the form $\tilde{\mathbf{M}}_{16}\left[...\right](n)$ *does contain a GCPL function, so finding even just the dominating asymptotics entails dealing with it.*

2. In Section 2.4, we proved a theorem that gave us a rather simple criterion to check if we can perform an asymptotic and even series expansion of a function F defined as a parametric integral, i.e.,

$$F(x,z) = \int_0^1 \mathrm{d}z\, f(x,z),$$

for some function f defined over $[0;1]\times[0.1]$. We then presented Method 2 on page 102 which allowed us to check in certains cases whether, given a GCPL function, we could perform an asymptotic expansion of it or not.

Using these two ideas, we can now devise a method to obtain asymptotics of certain binomial sums that could not be obtained with the methods introduced above. For instance, after computing a Mellin inverse following Section 2.3, we obtain an expression of the form (2.5.4), i.e.,

$$\tilde{\mathbf{M}}_a\left[f(x)\right](n) := \int_0^1 \mathrm{d}x\, \frac{(ax)^n - 1}{x - \frac{1}{a}} f(x) = \int_0^1 \mathrm{d}x\, \left[(ax)^n - 1\right] \tilde{f}(x)$$

with

$$\tilde{f}(x) := \frac{f(x)}{x - \frac{1}{a}}.$$

Now, we suppose that after such a computation, we ended up with a GCPL function, i.e., the expression above takes the form

$$\tilde{\mathbf{M}}_a\left[f(x)\right](n) := \int_0^1 \mathrm{d}x\, \frac{(ax)^n - 1}{x - \frac{1}{a}} \mathcal{H}^*_{\underline{V}}(x) = \int_0^1 \mathrm{d}x\, \left[(ax)^n - 1\right] \mathcal{H}^*_{\underline{V}'}(x) \tag{2.5.33}$$

for a certain generalized word $\underline{V}$, and

$$\underline{V'} = \left(\frac{1}{\lambda - \frac{1}{a}}, \underline{V}\right), \tag{2.5.34}$$

where we used the $\mathcal{H}^*_{\{\mathrm{f},\underline{V}\}}(x) := f(x)\mathcal{H}^*_{\underline{V}}(x)$ property of the GCPL. For instance, in the Example 2.5.6, we have

$$\underline{V} = \left(\lambda, \left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}}; \emptyset\right], \left[\frac{1}{\lambda - \frac{1}{4}}\sqrt{\frac{\lambda}{1-\lambda}}; \emptyset\right]\right)\right), \quad \underline{V'} = \left(\frac{1}{x - \frac{1}{16}}, \underline{V}\right).$$

Remark 2.5.10. *While* `RICA` *usually tries all the methods described above, in particular in Section 2.3, in order to compute the Mellin inverses, we also implemented a special option called* `UseGLCb` *that can be used both in* `AsymptoticsMellint` *and* `SumToMellin`*, which forces* `RICA` *to use the GCPL representation as much as possible so that expressions take the form* (2.5.33).

We finally need one last ingredient that is a "rewriting" of Watson's lemma. Indeed, Watson's lemma written for a function $g(z)$ expanded around $z = 0$ for the asymptotics of the integral $\int_0^{+\infty} \mathrm{d}z\, e^{-zn} g(z)$ as $n \to +\infty$ can be rewritten for a function $f(x) = \frac{g(-\log x)}{x}$ expanded around the point $x = 1$ for the asymptotics of the integral $\int_0^1 \mathrm{d}x\, x^n f(x)$ as $n \to +\infty$. Let us examine that before formerly stating this alternative version of the lemma.

Because the two integrals are related by a smooth substitution $x = e^{-z}$, we can replace the conditions

- $|f(z)| < e^{cz} \to |f(x)| < \frac{1}{x^{c+1}}$,
- $f(z) = \sum_{k=0}^{m} \alpha_k z^{\beta+\gamma k} + R_m(z) \to f(x) = \sum_{k=0}^{m} \bar{\alpha}_k (1-x)^{\beta+\gamma k} + \bar{R}_m(1-x)$.

In particular, the coefficients $\bar{\alpha}_k$ of the expansion of $f(x)$ around $x = 1$ can be obtained using the fact that $f(x) = \frac{g(-\log(x))}{x}$. We have in particular

$$\begin{aligned} f(x) &\underset{x\to 1}{=} \sum_{k=0}^{n} \frac{(x-1)^k}{k!} \left.\frac{\mathrm{d}^k f(x)}{\mathrm{d}x^k}\right|_{x=1} \\ &= \sum_{k=0}^{n} \frac{(x-1)^k}{k!} \left.\frac{\mathrm{d}^k \frac{g(-\log(x))}{x}}{\mathrm{d}x^k}\right|_{x=1}. \end{aligned}$$

We need therefore to compute:

$$\begin{aligned} \left.\frac{\mathrm{d}^n \frac{g(-\log(x))}{x}}{\mathrm{d}x^n}\right|_{x=1} &= \sum_{k=0}^{n} \binom{n}{k} \frac{\mathrm{d}^{n-k}\left(\frac{1}{x}\right)}{\mathrm{d}x^{n-k}} \cdot \frac{\mathrm{d}^k g(-\log x)}{\mathrm{d}x^k} \\ &= \sum_{k=0}^{n} \binom{n}{k} \frac{(-1)^{n-k}(n-k)!}{x^{n-k+1}} \cdot \frac{\mathrm{d}^k g(-\log x)}{\mathrm{d}x^k} \\ &= \sum_{k=0}^{n} \frac{n!}{k!} \frac{(-1)^{n-k}}{x^{n-k+1}} \cdot \frac{\mathrm{d}^k g(-\log x)}{\mathrm{d}x^k}. \end{aligned}$$

In order to compute the k^{th} derivative of $g(-\log x)$ we use the Faà di Bruno formula written with the Bell exponential polynomials [107, 108], i.e.,

$$\frac{\mathrm{d}^n (f \circ g)(x)}{\mathrm{d}x^n} = \sum_{k=0}^{n} f^{(k)}(g(x)) B_{n,k}\left(g'(x), g''(x), \ldots, g^{(n-k+1)}(x)\right) \tag{2.5.35}$$

where

$$B_{n,k}(x_1, x_2, \ldots, x_{n-k+1}) = n! \sum_{\substack{\sum_{i=1}^{n-k+1} j_k = k \\ \sum_{i=1}^{n-k+1} k j_k = n}} \prod_{i=1}^{n-k+1} \frac{x_i^{j_i}}{(i!)^{j_i} j_i!} \tag{2.5.36}$$

is the partial Bell polynomial. Therefore, combining everything together and sending $x \to 1$ we obtain:

$$\left.\frac{\mathrm{d}^n \frac{g(-\log(x))}{x}}{\mathrm{d}x^n}\right|_{x=1} = \sum_{k=0}^{n} \sum_{p=0}^{k} \frac{n!}{k!} g^{(p)}(0) \cdot B_{p,k}(g'(0), \ldots, g^{(k-p+1)}(0)) \tag{2.5.37}$$

or, if we replace $g^{(k)}(0) = k!\alpha_k$ we obtain:

$$\bar{\alpha}_k = \sum_{k=0}^{n} \sum_{p=0}^{k} \frac{n!}{k!} p! \alpha_p \cdot B_{p,k}(1!\alpha_1, \ldots, (k-p+1)!\alpha_{k-p+1}). \tag{2.5.38}$$

Let us now state the reformulated Watson's lemma.

Lemma 2.5.4 (Rewritten Watson's lemma)**.** *Let f be a function defined on $(0;1)$ such that $|f(z)| \leq \frac{1}{x^{c+1}}$ for some $c \in \mathbb{R}$ when $x \to 0$ and*

$$\forall m \in \mathbb{N}, \quad f(x) \underset{x\to 1^-}{=} \sum_{k=0}^{m} \alpha_k (1-x)^{\beta+\gamma k} + R_m(x), \quad \beta > -1,\, \gamma > 0 \tag{2.5.39}$$

where $\alpha_k \in \mathbb{R}$ and

$$R_m(x) \underset{x\to 1^-}{=} \mathcal{O}\left((1-x)^{\beta+\gamma(m+1)}\right). \tag{2.5.40}$$

Then we have

$$\forall p \in \mathbb{N},\, I(n) := \int_0^0 \mathrm{d}x\, x^n f(x) \underset{n\to+\infty}{=} \sum_{k=0}^{m} \alpha_k \int_0^1 \mathrm{d}x\, x^n (1-x)^{\beta+\gamma k} + \mathcal{R}_m(n) \tag{2.5.41}$$

where

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{1}{n^{\beta+\gamma(m+1)}}\right). \tag{2.5.42}$$

Proof. Plugging the expansion (2.5.8) in $I(n)$ we get

$$\begin{aligned} I(n) &= \int_0^1 \mathrm{d}x\, x^n \left(\sum_{k=0}^{m} \alpha_k (1-x)^{\beta+\gamma k} + R_m(x)\right) \\ &= \sum_{k=0}^{m} \alpha_k \underbrace{\int_0^1 \mathrm{d}x\, x^n (1-x)^{\beta+\gamma k}}_{=:B_k(n)} + \int_0^1 \mathrm{d}x\, x^n R_m(x). \end{aligned} \tag{2.5.43}$$

Now we have to show that

$$\mathcal{R}_m(n) = \int_0^1 \mathrm{d}x\, x^n R_m(x) \overset{?}{=} \mathcal{O}\left(\frac{1}{n^{m+1}}\right).$$

By hypothesis (2.5.42) and Definition 2.4.2 we get

$$\forall \eta \in (0;1),\, \exists \epsilon_\eta > 0,\, \forall z \in (\eta;1) \quad |R_m(x)| \leq \epsilon_\eta (1-x)^{\beta+\gamma(m+1)}.$$

Because we also require $|f(z)| \leq \frac{1}{x^{c+1}}$ as $x \to 0$ we can also bound

$$|R_m(x)| \leq M_\eta \frac{1}{x^{c+1}}$$

for $x \in (0;\eta)$ for some $M_\eta \in \mathbb{R}_+$. Therefore for any $\eta > 0$ we have

$$|\mathcal{R}_m(n)| \leq \epsilon_\eta \int_\eta^0 \mathrm{d}x\, x^n (1-x)^{\beta+\gamma(m+1)} + M_\eta \int_0^\eta \mathrm{d}x\, x^n \frac{1}{x^{c+1}}$$

$$\text{positivity of integrand} \rightarrow \leq \epsilon_\eta \int_0^1 \mathrm{d}x\, x^n (1-x)^{\beta+\gamma(m+1)} + \frac{M_\eta}{n-c}\eta^{n-c}$$
$$= \epsilon_\eta B_{m+1}(n) + \frac{M_\eta}{n-c}\eta^{n-c}. \tag{2.5.44}$$

We recognise in $B_{m+1}(n)$ a beta function, i.e.,

$$B_{m+1}(n) = \frac{\Gamma(n+1)\Gamma(\beta+m\gamma+\gamma+1)}{\Gamma(n+\beta+m\gamma+\gamma+2)}.$$

Using Gautschi's inequality for ratios of Gamma functions, we obtain

$$|B_{m+1}(n)| \leq \frac{\Gamma(\beta+m\gamma+\gamma+1)}{(n+1)^{\beta+\gamma(m+1)+1}} \leq \frac{\Gamma(\beta+m\gamma+\gamma+1)}{n^{\beta+\gamma(m+1)}}.$$

Moreover, because $\eta \in (0;1)$, we have

$$\forall \epsilon > 0,\ \exists n_\epsilon > 0,\ \forall n \geq n_\epsilon,\ \left|\frac{M_\eta}{n-c}\eta^{n-c}\right| \leq \frac{\epsilon}{n^{\beta+\gamma(m+1)}}$$

which means that

$$\forall \epsilon > 0,\ \exists n_\epsilon > 0,\ \forall n \geq n_\epsilon,\ |\mathcal{R}_m(n)| \leq \epsilon \frac{A_m}{n^{\beta+\gamma(m+1)}}$$

for $A_m = \Gamma(\beta+\gamma(m+1))+1$, or in other words

$$\mathcal{R}_m(n) \underset{n\to+\infty}{=} \mathcal{O}\left(\frac{1}{n^{\beta+\gamma(m+1)}}\right)$$

which concludes the proof. □

For our most flexible method we require for (2.5.34) the following assumption.

- Give $\frac{\mathcal{H}^*_{\underline{\mathrm{V}}'}(x)}{(1-x)^\alpha}$ with $\alpha \in (-1;1)$ such that it has a convergent series expansion around the point $x = 1$.

We consider a general α but for a root-valued alphabet with (2.1.43) — (2.1.78), we focus on f_1, f_{w_1} and f_{w_3}.

Depending on the value of a we either stay in the x representation (FSA, GSA) or we make a substitution and have a z variable (COV). In both cases, if the expression we get is of the form (2.5.33)

$$\int_0^1 \mathrm{d}x\ [(ax)^n - 1]\, \mathcal{H}^*_{\underline{\mathrm{V}}'}(x), \tag{2.5.45}$$

we simply do the following:

1. Check whether $\mathcal{H}^*_{\underline{\mathrm{V}}'}(x)$ has a series expansion around $x = 1$ using Theorem 2.4.5. If that's not the case we **stop and fail**, otherwise we continue.

2. Compute the series expansion up to order p, i.e., write

$$\mathcal{H}^*_{\underline{\mathrm{V}}'}(x) \underset{x\to 1}{=} (1-x)^b \sum_{k=0}^{n} \alpha_k (1-x)^k + \mathcal{O}\left((1-x)^{b+n+1}\right), \quad b \in (-1;1). \tag{2.5.46}$$

3. Finally, depending on the method we are using, we proceed as follows:
   - FSA/GSA: Plug directly (2.5.46) in (2.5.33), i.e., write
   $$\int_0^1 \mathrm{d}x \, [(ax)^n - 1] (1-x)^a \sum_{k=0}^{n} \alpha_k (1-x)^k. \tag{2.5.47}$$
   and use the Lemma 2.5.4 and Watson's Lemma 2.5.2 to conclude.
   - COV: We suppose we have the change of variable $x = \frac{e^{-z}}{a}$, and therefore because (2.5.33) becomes
   $$\int_0^1 \mathrm{d}x \left[e^{-zn} - 1\right] \frac{e^{-z}}{a} \mathcal{H}^*_{\underline{\mathrm{V}}'}\left(\frac{e^{-z}}{a}\right). \tag{2.5.48}$$
   Therefore, it suffices this time to expand the GCPL around $x = 1$, then replace $x \to \frac{e^{-z}}{a}$, i.e.,
   $$\int_{-\log a}^{+\infty} \mathrm{d}z \left[e^{-zn} - 1\right] \underbrace{\left(1 - \frac{e^{-z}}{a}\right)^b \sum_{k=0}^{n} \alpha_k \left(1 - \frac{e^{-z}}{a}\right)^k}_{A(z)}. \tag{2.5.49}$$
   Finally we expand again around $z = 0$ the $A(z)$ part and use Watson's Lemma 2.5.2 or its extended version Lemma 2.5.3 to conclude.

Example 2.5.7. *Let us apply the method described above to Example 2.5.6, i.e.,*

$$\begin{aligned}\mathcal{BS}_{((1,1,0),(1,1,0))}(n) =& \frac{1}{\pi^2} \tilde{\mathbf{M}}_{16} \left[\underbrace{\frac{x}{x - \frac{1}{16}} \mathcal{H}^*\left(\left[\tfrac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right], \left[\tfrac{1}{\lambda - \frac{1}{4}}\sqrt{\tfrac{\lambda}{1-\lambda}};\emptyset\right]\right)(x)}_{=:G(x)} \right](n) \\ & - \frac{4C}{\pi^2} \tilde{\mathbf{M}}_4 \left[\frac{\sqrt{x}}{\sqrt{1-x}(4x-1)} \right](n),\end{aligned} \tag{2.5.50}$$

supposing that we want only the dominating asymptotics up to order $p = 5$. We single out the $\tilde{\mathbf{M}}_{16}[...](n)$ term, which turns out to be the one containing the most complicated part. Now we follow the steps described above.

1. *First we check whether*

$$\mathcal{H}^*_{\underline{\mathrm{V}}}(x) = \frac{x\mathcal{H}^*_{\underline{\mathrm{W}}}(x)}{x-\frac{1}{16}}$$

with

$$\underline{V} = \left(\frac{\lambda}{\lambda-\frac{1}{16}}, \left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right], \left[\frac{1}{\lambda-\frac{1}{4}}\sqrt{\frac{\lambda}{1-\lambda}};\emptyset\right]\right)\right),$$

$$\underline{W} = \left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right], \left[\frac{1}{\lambda-\frac{1}{4}}\sqrt{\frac{\lambda}{1-\lambda}};\emptyset\right]\right)$$

has a series expansion around $x = 1$:

- *Because we have for all $n \in \mathbb{N}$ the geometric series expansion*

$$\frac{x}{x-\frac{1}{16}} \underset{x\to 1}{=} 1 + \frac{1}{15}\sum_{k=0}^{n}\left(\frac{16}{15}\right)^k (1-x)^k + \mathcal{O}\left((1-x)^{n+1}\right),$$

  we can concentrate on the remaining GCPL part. More precisely, in our case there are no nested convolutions in the GCPL function. So we look directly at the convolution stemming from $\underline{W}$.

- *We write explicitely the GCPL and make immediately the usual change of variables to have 0 and 1 as the bounds of the integral. We get:*

$$\mathcal{H}^*_{\underline{\mathrm{W}}}(x) = -4\int_0^1 \mathrm{d}z \underbrace{\frac{\sqrt{\frac{x}{(1-z)z((x-1)z+1)}}}{x(z-4)-z+1}}_{=:f(x,z)}. \tag{2.5.51}$$

- *Following the Method 1 on page 96 in Section 2.4, we split $f(x,z)$ into the following factors:*

$$\left\{x, 1+x, \frac{1}{\sqrt{x}}, \frac{1}{\sqrt{xz-z+1}}, \frac{1}{xz-4x-z+1}, \frac{1}{x+1}\right\}, \tag{2.5.52}$$

  then we collect the polynomial terms and compute for the n^{th}-derivative:

$$\{(-1)^{(n)}(z-4)^n(xz-4x-z+1)^{-n-1}, \left(-\frac{1}{2}\right)^{(n)} z^n(xz-z+1)^{-n-\frac{1}{2}},$$
$$\left(-\frac{1}{2}\right)^{(n)} x^{-n-\frac{1}{2}}, (-1)^{(n)}(x+1)^{-n-1}, \frac{\mathrm{d}^x}{\mathrm{d}n^x}[x(1+x)]\}. \tag{2.5.53}$$

- *We combine everything using (2.4.63), sending $x \to 1$ and adding the $\frac{(x-1)^n}{n!}$ term to get the n^{th} term of the Taylor series:*

$$\frac{(x-1)^n}{\sqrt{z-1}\sqrt{z}}\sum_{k_1,k_2,k_3,k_4=0}^{n}(-3)^{-k_1-1}(-1)^{k_1+k_2+k_3+k_4}2^{-2k_2-2k_3-k_4-1}$$
$$\times(z-4)^{k_1}z^{k_2}\binom{2k_2}{k_2}\binom{2k_3}{k_3}\frac{\mathrm{d}^{n-k_1-k_2-k_3-k_4}}{\mathrm{d}x^{n-k_1-k_2-k_3-k_4}}\left(x(1+x)\right)\Big|_{x=1}. \quad (2.5.54)$$

- *Bounding the derivative of the polynomial by a constant A and the binomial coefficients using (2.4.66), we obtain*

$$\left|\frac{(x-1)^n}{n!}\partial_x^n\, f(x,z)|_{x=1}\right| \leq \frac{A\,|1-x|^{n+1}}{\sqrt{1-z}\sqrt{z}} \times\sum_{k_1,k_2,k_3,k_4=0}^{n}2^{2k_1-k_4-1}3^{-k_1-1}, \quad (2.5.55)$$

which means that by Lemma 2.4.2 there exists a constant $C\in\mathbb{R}_{\geq 1}$ such that

$$\left|\frac{(x-1)^n}{n!}\partial_x^n\, f(x,z)|_{x=1}\right| \leq \frac{AC}{\sqrt{1-z}\sqrt{z}}\,|1-x|^{n+1}\,. \quad (2.5.56)$$

Thus we can apply Theorem 2.4.5.

- *The previous point implies that the convergent series expansion of our GCPL[70] is indeed obtained by intergating term-by-term the series expansion of $f(x,z)$ around $x=1$, i.e.,*

$$\mathcal{H}^*\left(\left[\tfrac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\tfrac{\sqrt{\frac{\lambda}{1-\lambda}}}{\lambda-\frac{1}{4}};\emptyset\right]\right)(x) \underset{x\to 1}{=} \frac{4\pi}{3}-\frac{11}{9}\pi(x-1)+\frac{61}{48}\pi(x-1)^2$$
$$-\frac{7223\pi(x-1)^3}{5184}+\frac{1571137\pi(x-1)^4}{995328}$$
$$-\frac{7291793\pi(x-1)^5}{3981312}+\mathcal{O}\left((1-x)^6\right). \quad (2.5.57)$$

2. *Because in our example we consider an expression of the form $\tilde{\mathbf{M}}_{16}[\cdots](n)$, i.e., our $a=16>1$ we use the COV method taking*[71]

$$G(x) \underset{x\to 1}{=} \frac{x}{x-\frac{1}{16}}\left(\frac{4\pi}{3}-\frac{11}{9}\pi(x-1)+\frac{61}{48}\pi(x-1)^2\right.$$
$$-\frac{7223\pi(x-1)^3}{5184}+\frac{1571137\pi(x-1)^4}{995328}$$
$$\left.-\frac{7291793\pi(x-1)^5}{3981312}+\mathcal{O}\left((1-x)^6\right)\right)$$

[70] In a certain region $r<x<1$, $r<1$.
[71] We defined $G(x)$ in (2.5.50).

- *Make the change of variable* $x = \frac{e^{-z}}{16}$ *in the integral*

$$\begin{aligned}\tilde{\mathbf{M}}_{16}\left[G(x)\right](n) &= \int_0^1 \mathrm{d}x\ \left[(16x)^n - 1\right] G(x) \\ &= \int_{-\log 16}^{+\infty} \mathrm{d}z\,(e^{-zn} - 1)\underbrace{\frac{e^{-z}}{16} G\left(\frac{e^{-z}}{16}\right)}_{=:\bar{G}(z)}.\end{aligned} \tag{2.5.58}$$

- *Perform the integration by parts and because we want the dominating asymptotics we ignore the constant contribution:*

$$\tilde{\mathbf{M}}_{16}\left[G(x)\right](n) = n\int_{-\log 16}^{+\infty} \mathrm{d}z\, e^{-zn} \int_{-\log 16}^{z} \mathrm{d}w\, \bar{G}(w). \tag{2.5.59}$$

- *Now, it suffices to plug in the expansion for* $G(x)$*, expand around the point* $z = -\log 16$ *and perform the integral* $\int_{-\log 16}^{z} \mathrm{d}w\, \bar{G}(w)$*, which gives us:*

$$\begin{aligned}\tilde{\mathbf{M}}_{16}\left[G(x)\right](n) = n\int_{-\log 16}^{+\infty} \mathrm{d}z\, e^{-zn} \Bigg(&\frac{4}{45}\pi(z + 4\log(2)) - \frac{\pi(z + 4\log(2))^2}{1350} \\ &\frac{1589\pi(z + 4\log(2))^3}{486000}. \\ &+ \frac{17117\pi(z + 4\log(2))^4}{12960000} \\ &\frac{29908369\pi(z + 4\log(2))^5}{46656000000}\Bigg).\end{aligned} \tag{2.5.60}$$

- *Finally, integrating over* z *gives us the asymptotic expansion for* $n \to +\infty$ *according to Watson's Lemma* 2.5.2*:*

$$\begin{aligned}\sum_{i=1}^{n}\binom{2i}{i}\sum_{j=1}^{i}\binom{2j}{j} =& \frac{2^{4n+6}}{45\pi n} - \frac{2^{4n+4}}{675\pi n^2} + \frac{1589\ 2^{4n+1}}{10125\pi n^3} + \frac{17117\ 2^{4n-1}}{16875\pi n^4} \\ &+ \frac{29908369\ 2^{4n-5}}{759375\pi n^5} + \mathcal{O}\left(\frac{16^n}{n^6}\right).\end{aligned} \tag{2.5.61}$$

Using `RICA`, we can obtain the result from the example above calling once again the function `AsymptoticsSum`:

```
In[25]:= AsymptoticsSum[GS[Binomial[2VarGL, VarGL], Binomial[2VarGL, VarGL], n],
         n, x, 5]
```

$$\text{Out[25]=}\quad \frac{16\left(\frac{29908369\pi 2^{4n-9}}{759375n^5}+\frac{17117\pi 2^{4n-5}}{16875n^4}+\frac{1589\pi 2^{4n-3}}{10125n^3}-\frac{\pi 2^{4n}}{675n^2}+\frac{\pi 2^{4n+2}}{45n}\right)}{\pi^2}$$

One natural question at this point is to know to which class of binomial sums this procedure can apply, i.e., for which binomial sums we can get an asymptotic expansion without entering a step that fails. Let us define the following specific subclass of binomial sums.

Definition 2.5.1 (Strict binomial sums)**.** *Let $n, k \in \mathbb{N}^\star$. We say that a binomial sum $\mathcal{BS}_{(a_1,\dots,a_k)}(n)$ is a **strict binomial sum** if for all $p \in \{1,\dots,k\}$ we have*

$$a_p = a_p(i; b_p, c_p, m_p) = \binom{2i}{i}^{b_p} \frac{\left(\frac{c_p}{4^{b_p}}\right)^i}{i^{m_p}}$$

$$b_p \in \{-1, 0, 1\},\ c_p \in \mathbb{R}\backslash\{1\},\ (b_p \neq -1 \wedge m_p \in \mathbb{N}) \vee (m_p \in \mathbb{N}^\star)$$

and moreover for all $i, j \in \{1,\dots,k\}$, $i \leq j$ we have

$$\prod_{i\leq \ell\leq j} c_\ell \neq 1.$$

Remark 2.5.11. *Strict binomial sums are defined in such a way that they correspond exactly to those binomial sums that will not require plus-prescription convolutions when Mellin inverted: locally, asking for all $p \in \{1,\dots,k\}$ that $c_p \in \mathbb{K}\backslash\{1\}$ ensures that any summand a_p, if summed upon using (2.1.27), does not give rise to a $\frac{x}{1-x}$ factor; and since (2.1.27) keeps the constants even after resummation, the global condition $\prod_{i\leq\ell\leq j} c_\ell \leq 1$ ensures that even as we sequentially Mellin inverse and convolve the summands, there is no compensation between constants that would give rise to a $\frac{x}{1-x}$ factor after resummation.*

Lemma 2.5.5. *Let $a, b \in \mathbb{R}\backslash\{1\}$ such that the following are strict binomial sums*

$$\mathcal{BS}_1(n) = \sum_{i=1}^{n} \left(\frac{a}{4}\right)^i \binom{2i}{i} \sum_{j=1}^{i} \left(\frac{b}{4}\right)^j \binom{2j}{j}, \quad \mathcal{BS}_2(n) = \sum_{i=1}^{n} \frac{(4a)^i}{i\binom{2i}{i}} \sum_{j=1}^{i} \left(\frac{b}{4}\right)^j \binom{2j}{j}$$

$$\mathcal{BS}_3(n) = \sum_{i=1}^{n} \left(\frac{a}{4}\right)^i \binom{2i}{i} \sum_{j=1}^{i} \frac{(4b)^i}{j\binom{2j}{j}}, \quad \mathcal{BS}_4(n) = \sum_{i=1}^{n} \frac{(4a)^i}{i\binom{2i}{i}} \sum_{j=1}^{i} \frac{(4a)^i}{i\binom{2i}{i}}.$$

Then one can perform their asymptotic expansion using GCPL functions.

Proof. Let us consider for now $\mathcal{BS}_1(n)$. We compute first its Mellin inverse, while keeping all the convolutions as GCPL functions:

$$\mathcal{BS}_1(n) = \sum_{i=1}^{n} \left(\frac{a}{4}\right)^i \binom{2i}{i} \sum_{j=1}^{i} \frac{b^j}{\pi} \mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](j)$$

$$
\begin{aligned}
&= \frac{1}{\pi}\sum_{i=1}^{n}\left(\frac{a}{4}\right)^i\binom{2i}{i}\left(b^i\mathbf{M}\left[\frac{1}{x-\frac{1}{b}}\sqrt{\frac{x}{1-x}}\right](i) - \underbrace{\mathbf{M}\left[\frac{1}{x-\frac{1}{b}}\sqrt{\frac{x}{1-x}}\right](0)}_{=:C_1}\right)\\
&= \frac{1}{\pi^2}\sum_{i=1}^{n}(ab)^i\mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](i)\cdot\mathbf{M}\left[\frac{1}{x-\frac{1}{b}}\sqrt{\frac{x}{1-x}}\right](j)\\
&\quad - \frac{C_1}{\pi^2}\sum_{i=1}^{n}a^i\mathbf{M}\left[\frac{1}{\sqrt{x(1-x)}}\right](i)\\
&= \frac{1}{\pi^2}\sum_{i=1}^{n}(ab)^i\mathbf{M}\left[\underbrace{\mathcal{H}^*\left(\left[\tfrac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\tfrac{1}{\lambda-\frac{1}{b}}\sqrt{\tfrac{\lambda}{1-\lambda}};\emptyset\right]\right)(x)}_{=:\mathcal{H}^*_{\underline{\mathrm{W}_1}}(x)}\right](i)\\
&\quad - \frac{C_1}{\pi^2}\tilde{\mathbf{M}}_1\left[\frac{1}{x-\frac{1}{a}}\sqrt{\frac{x}{1-x}}\right](n)\\
&= \frac{1}{\pi^2}\tilde{\mathbf{M}}_1\left[\frac{x}{x-\frac{1}{ab}}\mathcal{H}^*_{\underline{\mathrm{W}_1}}(x)\right](n) - \frac{C_1}{\pi^2}\tilde{\mathbf{M}}_1\left[\frac{1}{x-\frac{1}{a}}\sqrt{\frac{x}{1-x}}\right](n).
\end{aligned}
$$

Therefore, we have to check whether one can compute asymptotic expansions of $\mathcal{H}^*_{\underline{\mathrm{W}_1}}(x)$, which written in integral form is:

$$
\bar{F}_1(x) = -b\int_0^1 \mathrm{d}z\,\underbrace{\frac{\sqrt{-\frac{x}{(z-1)z((x-1)z+1)}}}{-bx+(x-1)z+1}}_{=:\bar{f}_1(x,z)}.
$$

We can bound the n^{th} derivative of $\bar{f}_1(x,z)$ with:

$$
\left|\partial_x^n\bar{f}_1(x,z)\Big|_{x=1}\right| \leq \frac{\bar{A}_1\sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n}\sum_{k_4=0}^{n}2^{-k_4-1}|1-b|^{-k_1-1}\left|(z-b)^{k_1}\right|}{\sqrt{1-z}\sqrt{z}},\quad \bar{A}_1\in\mathbb{R}_+
$$

Because we consider a strict binomial sum, we have $b \leq 1$ and therefore Lemma 2.4.2 and Theorem 2.4.4 apply, which means we can series expand $\mathcal{H}^*_{\underline{\mathrm{W}_1}}(x)$. Now, in a very similar way, the GCPL we obtain for the other three binomial sums are:

$$
\mathcal{BS}_2(n) \longrightarrow \mathcal{H}^*\left(\left[\tfrac{1}{\lambda\sqrt{1-\lambda}};\emptyset\right],\left[\tfrac{1}{\lambda-\frac{1}{b}}\sqrt{\tfrac{\lambda}{1-\lambda}};\emptyset\right]\right)(x) =: \mathcal{H}^*_{\underline{\mathrm{W}_2}}(x),
$$

$$\mathcal{BS}_3(n) \longrightarrow \mathcal{H}^*_{\left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}};\emptyset\right],\left[\frac{1}{\left(\lambda-\frac{1}{b}\right)\sqrt{1-\lambda}};\emptyset\right]\right)}(x) =: \mathcal{H}^*_{\underline{\mathrm{W}_3}}(x),$$
$$\mathcal{BS}_4(n) \longrightarrow \mathcal{H}^*_{\left(\left[\frac{1}{\lambda\sqrt{1-\lambda}};\emptyset\right],\left[\frac{1}{\left(\lambda-\frac{1}{b}\right)\sqrt{1-\lambda}};\emptyset\right]\right)}(x) =: \mathcal{H}^*_{\underline{\mathrm{W}_4}}(x).$$

The convolution integrals are:

$$\mathcal{H}^*_{\underline{\mathrm{W}_2}}(x) = \bar{F}_2(x) = -b\int_0^1 \mathrm{d}z\ \underbrace{\frac{\sqrt{\frac{x}{z-z^2}}}{((x-1)z+1)(-bx+(x-1)z+1)}}_{=:\bar{f}_2(x,z)},$$

$$\mathcal{H}^*_{\underline{\mathrm{W}_3}}(x) = \bar{F}_3(x) = -b\int_0^1 \mathrm{d}z\ \underbrace{\frac{1}{\sqrt{-((z-1)z)}(-bx+(x-1)z+1)}}_{=:\bar{f}_3(x,z)},$$

$$\mathcal{H}^*_{\underline{\mathrm{W}_4}}(x) = \bar{F}_4(x) = -b\int_0^1 \mathrm{d}z\ \underbrace{\frac{1}{\sqrt{-((z-1)z((x-1)z+1))}(-bx+(x-1)z+1)}}_{=:\bar{f}_4(x,z)}.$$

We can bound the n^{th} derivatives in each case by:

$$\left|\partial_x^n \bar{f}_2(x,z)\Big|_{x=1}\right| \leq \frac{\bar{A}_2 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n}\sum_{k_4=0}^{n} 2^{-k_4-1}|1-b|^{-k_1-1}\left|(z-b)^{k_1}\right|}{\sqrt{1-z}\sqrt{z}}, \quad \bar{A}_2 \in \mathbb{R}_+,$$

$$\left|\partial_x^n \bar{f}_3(x,z)\Big|_{x=1}\right| \leq \frac{\bar{A}_3 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n} 2^{-k_2-1}|1-b|^{-k_1-1}\left|(z-b)^{k_1}\right|}{\sqrt{1-z}\sqrt{z}}, \quad \bar{A}_3 \in \mathbb{R}_+,$$

$$\left|\partial_x^n \bar{f}_4(x,z)\Big|_{x=1}\right| \leq \frac{\bar{A}_4 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_3-1}|1-b|^{-k_1-1}\left|(z-b)^{k_1}\right|}{\sqrt{1-z}\sqrt{z}}, \quad \bar{A}_4 \in \mathbb{R}_+.$$

In all the cases, because $b \neq 1$, Lemma 2.4.2 and Theorem 2.4.5 apply, which concludes the proof. □

Starting with the single cases given in Lemma 2.5.5, we are now in the position to verify that Method 2 on page 102 is succesfully applicable to strict binomial sums.

Theorem 2.5.1. *Let $\mathcal{BS}_{(a_1,\ldots,a_k)}(n)$ be a strict binomial sum. Then one can always perform its asymptotic expansion using GCPL functions.*

Proof. Applying Thereom 2.2.1 to each of the summand parts not containing a

sum of the strict binomial sum $\mathcal{BS}_{(a_1,\ldots,a_k)}(n)$, we obtain:

$$\begin{aligned}\mathcal{BS}_{(a_1,\ldots,a_k)}(n) &= \sum_{i_1=1}^{n}\binom{2i_1}{i_1}^{b_1}\frac{\left(\frac{c_1}{4^{b_1}}\right)^{i_1}}{{i_1}^{m_1}}\sum_{i_2=1}^{i_1}\binom{2i_2}{i_2}^{b_2}\frac{\left(\frac{c_2}{4^{b_2}}\right)^{i_2}}{{i_2}^{m_2}}\cdots\sum_{i_k=1}^{i_{k-1}}\binom{2i_k}{i_k}^{b_k}\frac{\left(\frac{c_k}{4^{b_k}}\right)^{i_k}}{{i_k}^{m_k}}\\ &= \sum_{i_1=1}^{n}\left(\frac{c_1}{4^{b_1}}\right)^{i_1}\mathbf{M}\left[\frac{\mathrm{H}^{*}_{\mathbf{0}_{m_1-1},\bar{f}^{b_1}}(x)}{x}\right](i_1)\sum_{i_2=1}^{i_1}\left(\frac{c_2}{4^{b_2}}\right)^{i_2}\mathbf{M}\left[\frac{\mathrm{H}^{*}_{\mathbf{0}_{m_2-1},\bar{f}^{b_2}}(x)}{x}\right](i_2)\\ &\quad\cdots\sum_{i_k=1}^{i_{k-1}}\left(\frac{c_k}{4^{b_k}}\right)^{i_k}\mathbf{M}\left[\frac{\mathrm{H}^{*}_{\mathbf{0}_{m_k-1},\bar{f}^{b_k}}(x)}{x}\right](i_k)\end{aligned}\tag{2.5.62}$$

where

$$\forall p\in\{1,\ldots,k\},\quad \bar{f}^{b_p}(x)=\begin{cases} f_{w_1}(x) & \text{if}\quad b_p=1\\ f_{w_3}(x) & \text{if}\quad b_p=-1\\ \emptyset & \text{if}\quad b_p=0.\end{cases}$$

We also note

$$\forall p\in\{1,\ldots,k\},\quad a_p(n)=\binom{2n}{n}^{b_p}\frac{\left(\frac{c_p}{4^{b_p}}\right)^{n}}{n^{m_k}}=\left(\frac{c_p}{4^{b_p}}\right)^{n}\mathbf{M}\left[\frac{\mathrm{H}^{*}_{\mathbf{0}_{m_k-1},\bar{f}^{b_k}}(x)}{x}\right](n).\tag{2.5.63}$$

The whole idea of the proof is to apply Method 2 on page 102 twice. More precisely, we will apply it first to the individual summands a_p given in (2.5.63) to show that they can be expanded as a series (**Step I**), and afterward apply it to the convolutions of a_p's that will appear in the end result of the Mellin inversion of $\mathcal{BS}_{(a_1,\ldots,a_k)}(n)$ (**Step II**).

Step I: Property 2.3.2 tells us that we can rewrite each GRPL function as a pure GCPL, i.e., for all $p\in\{1,\ldots,k\}$ there exists a pure generalized word $\underline{V_p}$ such that we have:

$$\mathrm{H}^{*}_{\mathbf{0}_{m_p-1},\bar{f}^{b_p}}(x)=\mathcal{H}^{*}_{\underline{V_p}}(x).$$

More precisely, the pure letter $\underline{V_p}$ is of the form

$$\underline{V_p}=\Bigg(\lambda,\Bigg(\left[f_0(\lambda);\emptyset\right],\Bigg(\cdots\Bigg(\left[f_0(\lambda);\emptyset\right],\underbrace{\Bigg(\left[f_0(\lambda);\emptyset\right],\left[\bar{f}^{b_p}(\lambda);\emptyset\right]\Bigg)}_{=:\underline{\bar{V}_{b_p}}}\Bigg)\cdots\Bigg)\Bigg)\Bigg)\tag{2.5.64}$$

where, by construction of the pure GCPL, the $\cdots$ part in the $\underline{V_p}$ index is obtained by iterative convolutions with the w-letter $[f_0;\emptyset]$. If $m_p=0$, we have

$$\underline{V_p}=\left[\bar{f}^{b_p};\emptyset\right].\tag{2.5.65}$$

According to Step 2 and 3 of Method 2 on page 102, what needs to be checked are the base cases, i.e., the recursions $\underline{\bar{V}_1}, \underline{\bar{V}_{-1}}, \underline{\bar{V}_0}$, respectivelly associated to the following convolutions:

$$F_1(x) = (f_0 \star f_{w_1})(x) = \int_0^1 \mathrm{d}z \underbrace{\frac{1-x}{\sqrt{(x-1)x(z-1)((x-1)z+1)}}}_{=:f_1(x,z)},$$

$$F_{-1}(x) = (f_0 \star f_{w_3})(x) = \int_0^1 \mathrm{d}z \underbrace{\frac{1-x}{x\sqrt{(x-1)(z-1)((x-1)z+1)}}}_{=:f_{-1}(x,z)},$$

$$F_0(x) = (f_0 \star f_0)(x) = \int_0^1 \mathrm{d}z \underbrace{\frac{1-x}{(x-1)xz+x}}_{=:f_0(x,z)}.$$

The latter one corresponds to the case when we get a term without binomial coefficients. Following the bounding of the n^{th} derivative method, we get the bounds:

$$|\partial_x^n f_1(x,z)|_{x=1}| \leq \frac{A_1 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_3-1}}{\sqrt{1-z}}, \quad A_1 \in \mathbb{R}_+,$$

$$|\partial_x^n f_{-1}(x,z)|_{x=1}| \leq \frac{A_2 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_2-1}}{\sqrt{1-z}}, \quad A_2 \in \mathbb{R}_+,$$

$$|\partial_x^n f_0(x,z)|_{x=1}| \leq A_3 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_2-1}, \quad A_3 \in \mathbb{R}_+.$$

This means that using Lemma 2.4.2, Theorem 2.4.5 applies in all the cases.
Finally we enter Step 4 of Method 2 that splits into two cases. However, due to the shape of $\underline{V_p}$ in (2.5.64), we only enter into Case 1, but in an iterated fashion. Therefore, we have to treat $[f_i(\lambda); \emptyset]$ appearing in (2.5.64). Here we need to examine the limit as $x \to 1$ of

$$G_1(x) = \int_0^1 \mathrm{d}z \frac{\frac{1}{\frac{x}{1+(x-1)z}}}{1+(x-1)z} = \frac{1}{x}\int_0^1 \mathrm{d}z = \frac{1}{x} \underset{x\to 1}{\longrightarrow} 1; \tag{2.5.66}$$

compare (2.4.74) in the method. We immediately conclude that according to the method to check the validity of asymptotics expansions of GCPLs, each recursion step when computing the series expansion of the pure GCPL function will be valid. With that, it follows that we can perform the series expansions of GCPL associated

to the individual summands a_p of our strict binomial sum.

Step II: Finally, we treat the full expression of (2.5.62), given we know that the a_p given in (2.5.63) can be expanded as a series. To apply the Steps 2 and 3 of Method 2 on page 102, we need to find out the "fundamental cases" of our GCPLs. Note that the Mellin inversion method described in Section 2.3.2 proceeds as follows. We start from the most innerward summand, here a_k given in (2.5.63), invert, resum and convolve in succession. It means in particular that the associated generalized words in the GCPL representation will be read from right to left, i.e., the most complicated convolution will be a fundamental generalized word obtaining from computing the Mellin inverse of $\mathcal{BS}_{(a_k,a_{k-1})}(i_{k-2})$. More precisely, it means that the GCPLs that will appear as a result of iterated convolutions will have their generalized words of the form

$$\underline{W_p} = \left(\frac{1}{\lambda - \frac{4^{b_1}}{c_1}}, \left(\underline{V_1}, \cdots \left(\underline{V_{p-2}}, \left(\frac{1}{\lambda - \frac{4^{b_{p-1}}}{c_{p-1}}}, \underbrace{\left(\underline{V_{p-1}}, \underline{V_p}\right)}_{=:\underline{U_p}}\right)\right)\cdots\right)\right) \tag{2.5.67}$$

for $p \in \{1, \ldots, k\}$ with the $\underline{V_p}$ being those defined in (2.5.64) or (2.5.65).
First we observe that this generalized word is properly defined. Indeed, during the resummation process, we can obtain factors of the form

$$\frac{x}{x - \frac{1}{c_p}}, \quad p \in \{1, \ldots, k\}$$

that will multiply with the generalized words and form new generalized words because of (2.3.28). Because the binomial sum we consider is strict, we have that $c_p \neq 1$ and therefore this factor contributes just as a $\frac{1}{1-\frac{1}{c_p}}$ constant at $x = 1$.

Given the representation (2.5.67) we enter into Case 2 of Step 4 of our Method 2. The case when $m_{k-1}, m_k \geq 2$ in (2.5.62) corresponds to the case that $\underline{V_p}$ is of the form (2.5.64) which corresponds in turn to convolutions of two GRPL functions, which is already done because we just proved we can expand GRPLs and from the method to compute expansion of GCPLs, the product of two expansions being an expansion is always guaranteed[72]. So the induction step is settled.

What remains to show is that the "fundamental cases", that correspond to the fundamental generalized words of the method in Steps 2 and 3, can be properly expanded as a series, and that if the $\underline{V_i}$, $i \in \{1, \ldots, p-1\}$, outside of $\underline{U_p}$ in (2.5.67)

[72] As shown in the Case 2 of the method in Section 2.4.2.

are of the form (2.5.65) the Step 4 of the method is valid.

The "fundamental cases" are those where the $\underline{U_p}$ are of the form:

$$([f_{w_1};\emptyset],[f_{w_1};\emptyset]),([f_{w_1};\emptyset],[f_{w_3};\emptyset]),([f_{w_3};\emptyset],[f_{w_1};\emptyset]),([f_{w_3};\emptyset],[f_{w_3};\emptyset])$$

which correspond to the fundamental words we have to identify and check with Theorem 2.4.5 in the second and third steps of Method 2 page 102. Those cases correspond respectively to sums of the form[73]:

$$\mathcal{BS}_1(n)=\sum_{i=1}^{n}\left(\frac{a}{4}\right)^i\binom{2i}{i}\sum_{j=1}^{i}\left(\frac{b}{4}\right)^j\binom{2j}{j},\quad \mathcal{BS}_2(n)=\sum_{i=1}^{n}\frac{(4a)^i}{i\binom{2i}{i}}\sum_{j=1}^{i}\left(\frac{b}{4}\right)^j\binom{2j}{j}$$

$$\mathcal{BS}_3(n)=\sum_{i=1}^{n}\left(\frac{a}{4}\right)^i\binom{2i}{i}\sum_{j=1}^{i}\frac{(4b)^i}{j\binom{2j}{j}},\quad \mathcal{BS}_4(n)=\sum_{i=1}^{n}\frac{(4a)^i}{i\binom{2i}{i}}\sum_{j=1}^{i}\frac{(4a)^i}{i\binom{2i}{i}}.$$

We have already proved those in Lemma 2.5.5, and therefore do not need to go to the full fundamental cases which would be simple non-nested sums.

Finally, the last thing we need to check is if the 4$^{\text{th}}$ step of Method 2 on page 102 applies if the $\underline{V_i}$, $i\in\{1,\dots,p-1\}$, outside of $\underline{U_p}$ in (2.5.67) are of the form (2.5.65). We are in the case 1 of the method, as we will be looking at expressions of the form

$$f_i(x)\star\left(\frac{x}{x-\frac{1}{c}}\mathcal{H}^*_{\underline{W_j}(x)}\right)=\mathcal{H}^*_{\left([f_i;\emptyset],\left(\frac{\lambda}{\lambda-\frac{1}{c}},\underline{W_j}\right)\right)}(x)$$

where $i\in\{w_1,w_3\}$ and $j\in\{1,\dots,4\}$. Therefore, once again using the notation of $G(x)$ from (2.4.74), we have to check the behavior around 1 of two extra functions:

$$\bar{G}_1(x)=\int_0^1 \mathrm{d}z\,\frac{f_{w_1}\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}=\frac{1}{\sqrt{x(1-x)}}\int_0^1\mathrm{d}z\,\frac{1}{\sqrt{1-z}}=\frac{2}{\sqrt{x(1-x)}}$$

$$\bar{G}_2(x)=\int_0^1 \mathrm{d}z\,\frac{f_{w_3}\left(\frac{x}{1+(x-1)z}\right)}{1+(x-1)z}=\frac{1}{\sqrt{x(1-x)}}\int_0^1\mathrm{d}z\,\underbrace{\sqrt{\frac{1+(x-1)z}{1-z}}}_{=:g_2(x,z)}.$$

One can bound the derivative of $g_2(x,z)$ by:

$$\left|\partial_x^n g_2(x,z)\big|_{x=1}\right|\leq\frac{\tilde{A}\sum\limits_{k_1=0}^{n}\sum\limits_{k_2=0}^{n}\sum\limits_{k_3=0}^{n}\sum\limits_{k_4=0}^{n}2^{-k_4-1}}{\sqrt{1-z}\sqrt{z}},\quad \tilde{A}\in\mathbb{R}_+.$$

[73] In all the cases we have $a,b\in\mathbb{R}\backslash\{1\}$.

This means as usually that Lemma 2.4.2 and Theorem 2.4.5 apply, and in our case it means we can expand $g_2(x,z)$ up to first order and integrate to get the behavior of $\bar{G}_2(x)$ around $x=1$:

$$\bar{G}_2(x) \underset{x\to 1}{\sim} \frac{1}{\sqrt{x(1-x)}} \int_0^1 \mathrm{d}z \, \frac{1}{\sqrt{1-z}} \underset{x\to 1}{\sim} G_1(x) = \frac{2}{\sqrt{x(1-x)}}.$$

Because those two functions contribute with a pole of order strictly less than 1 at $x=1$, we can compute the series expansion of generalized words of the form $([f_i;\emptyset],\underline{U})$, with $\underline{U}$ being a generalized word coming from lower Mellin inversions, resummations and convolutions in $\mathcal{BS}_{(a_1,\dots,a_k)}(n)$. This concludes **Step II**. □

Remark 2.5.12. *Summarising, we have proved first that the Mellin inverse of individual summands, represented using pure GCPLs instead of GRPLs, can always be expanded as a series (**Step I**). The full Mellin inverse of $\mathcal{BS}_{(a_1,\dots,a_k)}(n)$ is then built by summing up and convolving the resulting expressions, starting from the most inward ones — we proved that even when those are not GRPL themselves but obtained as specific convolutions of all possible combinations of f_{w_1}, f_{w_3} functions, which we call "fundamental cases", we still can expand them*[74]*. At intermediate steps, we can also get convolutions with w-letters of the form $[f_{w_i};\emptyset]$, $i\in\{w_1,w_3\}$. So we checked that those would also be valid convolutions (**Step II**). With that, having positively exhausted all the cases, the proof is done.*

With the theorem, we realise that going to the GCPL formalism and writing all our functions as nested convolutions allows us to compute asymptotics algorithmically for strict binomial sums, which exhaust a huge part of total binomial sums. We should note one negative and one positive aspect of that.
On the negative side, while it may seem alluring to always convert our Mellin inverses into the GCPL formalism and deal with it, the constants will also get more complicated and perhaps have a less tractable representation for further simplification. So it is still better to first try to use the previous methods relying on [16] described in Section 2.3.2 to try to get as simple representation as possible for them.
On the positive side, GCPL functions in principle represent their own class of functions that may work outside of the binomial sum context. So while we proved Theorem 2.5.1 which guarantees the existence of asymptotics for strict binomial

[74] The proof can also be seen as a giant (but convoluted) induction-type proof which requires two levels of "base cases", due to the fact that Method 2 page 102 checks first fundamental letters and then possible convolutions with them. Because furthermore one convolution corresponds to a double nested sum, our "base cases" cover double and triple nested sums. Once this is done, the "induction step" is obtained for free due to the remaining steps of Method 2 and the way $\underline{W_p}$ from (2.5.67) is built.

sums, and are computable with the package `RICA`, asymptotic expansions of other classes of sums or functions in general can also be obtained. For example, one can define the new class of sums below that involve higher powers of binomial coefficients:

Definition 2.5.2 (Generalized binomial sums)**.** *Let $n \in \mathbb{N}$. We define a* ***generalized binomial sum*** *(GBS) as follows:*

$$\mathcal{BS}_{(\emptyset)}(n) = 1 \tag{2.5.68}$$

$$\mathcal{BS}_{(A_1)}(n) = \sum_{i=1}^{n} A_1(i; b, c, m) \tag{2.5.69}$$

$$\mathcal{BS}_{(A_1,\ldots,A_k)}(n) = \sum_{i_1=1}^{n} A_1(i_1; b_1, c_1, m_1)\mathcal{BS}_{(A_2,\ldots,A_k)}(i_1) \tag{2.5.70}$$

where we define the letters A_p of the generalized binomial sum as:

$$A_p(i; b, c, m) = \binom{2i}{i}^b \frac{c^i}{i^m},\ b \in \mathbb{Z},\ c \in \mathbb{K}^\star,\ m \in \mathbb{N}. \tag{2.5.71}$$

k is called the ***depth*** *of the sum. As for binomial sums, one will also use the notation:*

$$\mathcal{BS}_{(A_1,\ldots,A_k)}(n) = \mathcal{BS}_{((b_1,c_1,m_1),\ldots,(b_k,c_k,m_k))}(n). \tag{2.5.72}$$

Example 2.5.8. *Typical examples of GBSs are:*

$$\mathcal{BS}_{((2,1,0))}(n) = \sum_{i=1}^{n} \binom{2i}{i}^2$$

$$\mathcal{BS}_{((-3,2,2),(0,1,3))}(n) = \sum_{i=1}^{n} \frac{2^i}{i^2\binom{2i}{i}^3} \sum_{j=1}^{i} \frac{1}{j^3}$$

$$\mathcal{BS}_{((-1,1,0),(2,1,0),(-3,1,0),(4,1,0))}(n) = \sum_{i=1}^{n} \frac{1}{\binom{2i}{i}} \sum_{j=1}^{i} \binom{2j}{j}^2 \sum_{k=1}^{j} \frac{1}{\binom{2k}{k}^3} \sum_{l=1}^{k} \binom{2l}{l}^4$$

While we cannot deal with the whole class of generalized binomial sums even with our new method — this is also related to the discussion in Section 2.1.2.1 and problems with the +-prescription described in Section 2.3.5 — we can restrict this general class to a subclass where the expansions are computable.

Definition 2.5.3 (Strict generalized binomial sums)**.** *Let $n, k \in \mathbb{N}^\star$. We say that a generalized binomial sum $\mathcal{BS}_{(A_1,\ldots,A_)}(n)$ is a* ***strict generalized binomial sum*** *if for all $p \in \{1, \ldots, k\}$ we have*

$$A_p = A_p(i; b_p, c_p, m_p) = \binom{2i}{i}^{b_p} \frac{\left(\frac{c_p}{4^{b_p}}\right)^i}{i^{m_p}}$$

$$b_p \in \mathbb{Z},\, c_p \in \mathbb{K}\backslash\{1\},\, (b_p \geq 0 \wedge m_p \in \mathbb{N}) \vee (m_p \in \mathbb{N}_{b_p})$$

and moreover for all $i, j \in \{1, \ldots, k\}$, $i \leq j$ *we have*

$$\prod_{i \leq \ell \leq j} c_\ell \neq 1.$$

Corollary 2.5.1.1. *Let* $\mathcal{BS}_{(A_1,\ldots,A_k)}(n)$ *be a strict generalized binomial sum. Then one can always perform its asymptotic expansion using GCPL functions.*

Proof. Adding higher powers of binomial sums means that the proof is basically the same as for the strict binomial sum case, except that the individual summands will be represented themselves with GCPLs of the form:

$$\forall p \in \{1, \ldots, k\}, \quad A_p \longrightarrow \left(\frac{c_p}{4^{b_p}}\right)^{i_p} \mathbf{M}\left[\mathcal{H}^*_{\underline{V_p}}(x)\right](i_p).$$

The generalized word $\underline{V_p}$ is of the form

$$\underline{V_p} = \begin{cases} \left(\left[\frac{1}{\lambda}; \mathbf{0}_{m_p-1}\right], \underline{U_{b_p}}\right) & \text{if} \quad b_p \geq 0 \\ \left(\left[\frac{1}{\lambda}; \mathbf{0}_{m_p-b_p-1}\right], \underline{W_{b_p}}\right) & \text{if} \quad b_p < 0 \end{cases},$$

where the w-letters $\left[\frac{1}{\lambda}; \mathbf{0}_\ell\right]$, $\ell \in \mathbb{N}^\star$ can be rewritten as pure generalized words that we have dealt with in the proof of Theorem 2.5.1; here $\underline{U_{b_p}}, \underline{W_{b_p}}$ are defined recursively as follows if[75] $|b_p| > 1$:

$$\underline{U_1} := \left[\frac{1}{\sqrt{\lambda(1-\lambda)}}; \emptyset\right], \underline{U_{i+1}} := \left(\left[\frac{1}{\sqrt{\lambda(1-\lambda)}}; \emptyset\right], \underline{U_i}\right),$$

$$\underline{W_1} := \left[\frac{1}{\lambda\sqrt{1-\lambda}}; \emptyset\right], \underline{W_{i+1}} := \left(\left[\frac{1}{\lambda\sqrt{1-\lambda}}; \emptyset\right], \underline{W_i}\right).$$

It means in particular that the only two new convolutions we need to check are

$$\begin{aligned} F_1(x) = (f_{w_1} \star f_{w_1})(x) &= \left(\frac{1}{\sqrt{x(1-x)}} \star \frac{1}{\sqrt{x(1-x)}}\right) \\ &= \int_0^1 \mathrm{d}z \, \underbrace{\frac{1}{\sqrt{x(1-z)z((x-1)z+1)}}}_{=:f_1(x,z)}, \end{aligned}$$

[75] If $b_p \in \{-1, 0, 1\}$, we get back a strict binomial sum and this case had already been dealt with.

$$F_2(x) = (f_{w_3} \star f_{w_3})(x) = \left(\frac{1}{x\sqrt{1-x}} \star \frac{1}{x\sqrt{1-x}}\right)$$
$$= \int_0^1 \mathrm{d}z \underbrace{\frac{1}{x\sqrt{(1-z)z((x-1)z+1)}}}_{=:f_2(x,z)}.$$

We can bound both integrands by

$$|\partial_x^n f_1(x,z)|_{x=1}| \leq \frac{A_1 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_3-1}}{\sqrt{1-z}\sqrt{z}}, \quad A_1 \in \mathbb{R}_+,$$

$$|\partial_x^n f_2(x,z)|_{x=1}| \leq \frac{A_2 \sum_{k_1=0}^{n}\sum_{k_2=0}^{n}\sum_{k_3=0}^{n} 2^{-k_2-1}}{\sqrt{1-z}\sqrt{z}}, \quad A_2 \in \mathbb{R}_+,$$

which means that Lemma 2.4.2 and therefore Theorem 2.4.5 apply. Furthermore, we can series expand the GCPL obtained from the fundamental words $([f_{w_1};\emptyset],[f_{w_1};\emptyset])$ and $([f_{w_3};\emptyset],[f_{w_3};\emptyset])$, and all GCPL configurations of strict generalized binomial sums can be obtained from those fundamental words or generalized words described in the proof of Theorem 2.5.1. Therefore, this concludes the proof. □

To conclude, switching to the GCPL approach makes computations much more flexible because the problem of asymptotics in general is reduced thanks to Theorem 2.4.5 to a local derivative bounding problem. As long as this can be achieved, uniform convergence and therefore series expansion can be achieved. More generally, GCPLs have a rich and complex structure that has been only touched upon due to lack of time, and they could very well be their own object of study in the future.

2.5.5. Asymptotics and numerics

We introduce in this subsection one useful function we implemented in `RICA`, called `VerifyExpansion`, that allows one to perform an extra numerical check on the expansions, given the original binomial sum. Given a binomial sum $\mathcal{BS}_{(a_1,\ldots,a_k)}(n)$ and its asymptotic expansion at $n \to +\infty$, up to order $p \in \mathbb{N}$, of the form

$$A_p(n) = \alpha_1(n) + \alpha_2(n) + \ldots + \alpha_p(n) + \mathcal{O}\left(\alpha_{p+1}(n)\right),$$

where the α_i form an asymptotic sequence[76], the function `VerifyExpansion` numerically checks that for all $i \in \{1, \ldots, p-1\}$ we have

$$\frac{1}{\alpha_{i+1}(n)} \left(\mathcal{BS}_{(a_1,\ldots,a_k)}(n) - \alpha_1(n) - \ldots - \alpha_i(n) \right) \underset{n\to+\infty}{\longrightarrow} 1.$$

It then returns the absolute and the average relative error. Let us see below some examples of asymptotics where we confirm the expansion with a numerical check[77]:

Example 2.5.9. *We consider the binomial sum (5.17) p. 26 from [16]:*

$$\mathcal{BS}_{((1,-1,2))}(n) = \sum_{i=1}^{n} \frac{(-1)^i}{i^2} \binom{2i}{i}. \tag{2.5.73}$$

We define it in `Mathematica` *as:*

In[26]:= **sum1 = GS[(−1)^VarGL/VarGL^2 * Binomial[2VarGL, VarGL], n];**

Using the function `SumToMellin` *of* `RICA` *one obtains:*

In[27]:= **mel1 = SumToMellin[sum1, C, x, ToGLbBasis → False]**

Out[27]= $\left\{ \frac{4\,Mellin\left[(-4)^{n}x^{n}-1, \frac{Hwb(\{f0,fw1\},x)}{4x+1}\right]}{\pi}, \{\} \right\}$

which matches the expected expression. Now we can compute the asymptotic expansion using `AsymptoticsMellint`*:*

In[28]:= **asymp1 = AsymptoticsMellint[mel1[[1]], x, n, 7]**

Out[28]= $\frac{4\left(\frac{\sqrt{\pi}(-4)^{n}}{5n^{5/2}} - \frac{\sqrt{\pi}(-1)^{n}2^{2n-3}}{n^{7/2}} - \frac{11\sqrt{\pi}(-1)^{n}2^{2n-7}}{n^{9/2}} + \frac{13\sqrt{\pi}(-1)^{n}2^{2n-10}}{n^{11/2}} + \frac{32539\sqrt{\pi}(-1)^{n}2^{2n-15}}{5n^{13/2}}\right)}{\pi}$

One can check the expansion using the `VerifyExpansion` *function. The function takes as first argument the sum, as second argument the expansion and as third argument the argument of the sum, n in our case. It also can take an optional rule-argument called* `ShowSteps`*. Activating this feature, the function will show intermediate steps where it computes the absolute and relative errors for each term of the asymptotic expansion:*

In[29]:= **check1 = VerifyExpansion[sum1, asymp1, n, ShowSteps → True]**

Computation for term number: **0**

Expression to compare to the coefficient:

$$(-1)^{-n}2^{-2n}n^{5/2}\,GS\left[\left\{\frac{(-1)^{VarGL}\binom{2\,VarGL}{VarGL}}{VarGL^{2}}\right\}, n\right]$$

[76] See Definition 2.4.8.

[77] We will always expand up to order $p = 7$ our binomial sums, unless stated otherwise.

Coefficient we should get: $\mathbf{\frac{4}{5\sqrt{\pi}}}$
Absolute error: **0.000282289**
Relative error: **0.00062543**

Computation for term number: **1**
Expression to compare to the coefficient:
$$\mathbf{(-1)^{-n}2^{-2n}n^{7/2}}\left(GS\left[\left\{\frac{(-1)^{VarGL}\binom{2\,VarGL}{VarGL}}{VarGL^2}\right\},\mathbf{n}\right]-\frac{\mathbf{(-1)^n2^{2n+2}}}{\mathbf{5\sqrt{\pi}n^{5/2}}}\right)$$
Coefficient we should get: $\mathbf{-\frac{1}{2\sqrt{\pi}}}$
Absolute error: **0.000193911**
Relative error: **0.000687397**

Computation for term number: **2**
Expression to compare to the coefficient:
$$\mathbf{(-1)^{-n}2^{-2n}n^{9/2}}\left(GS\left[\left\{\frac{(-1)^{VarGL}\binom{2\,VarGL}{VarGL}}{VarGL^2}\right\},\mathbf{n}\right]-\frac{\mathbf{(-1)^n2^{2n+2}}}{\mathbf{5\sqrt{\pi}n^{5/2}}}+\frac{\mathbf{(-1)^n2^{2n-1}}}{\mathbf{\sqrt{\pi}n^{7/2}}}\right)$$
Coefficient we should get: $\mathbf{-\frac{11}{32\sqrt{\pi}}}$
Absolute error: **0.0000290991**
Relative error: **0.000150042**

Computation for term number: **3**
Expression to compare to the coefficient:
$$\mathbf{(-1)^{-n}2^{-2n}n^{11/2}}\left(GS\left(\left\{\frac{(-1)^{VarGL}\binom{2\,VarGL}{VarGL}}{VarGL^2}\right\},\mathbf{n}\right)-\frac{\mathbf{(-1)^n2^{2n+2}}}{\mathbf{5\sqrt{\pi}n^{5/2}}}+\frac{\mathbf{(-1)^n2^{2n-1}}}{\mathbf{\sqrt{\pi}n^{7/2}}}+\frac{\mathbf{11(-1)^n2^{2n-5}}}{\mathbf{\sqrt{\pi}n^{9/2}}}\right)$$
Coefficient we should get: $\mathbf{\frac{13}{256\sqrt{\pi}}}$
Absolute error: **0.000448823**
Relative error: **0.0156656**

Computation for term number: **4**
Expression to compare to the coefficient:
$$\mathbf{(-1)^{-n}2^{-2n}n^{13/2}}\left(GS\left[\left\{\frac{(-1)^{VarGL}\binom{2\,VarGL}{VarGL}}{VarGL^2}\right\},\mathbf{n}\right]-\frac{\mathbf{(-1)^n2^{2n+2}}}{\mathbf{5\sqrt{\pi}n^{5/2}}}+\frac{\mathbf{(-1)^n2^{2n-1}}}{\mathbf{\sqrt{\pi}n^{7/2}}}+\frac{\mathbf{11(-1)^n2^{2n-5}}}{\mathbf{\sqrt{\pi}n^{9/2}}}-\frac{\mathbf{13(-1)^n2^{2n-8}}}{\mathbf{\sqrt{\pi}n^{11/2}}}\right)$$

Coefficient we should get: $\dfrac{\mathbf{32539}}{\mathbf{40960\sqrt{\pi}}}$
Absolute error: **0.000610352**
Relative error: **0.00136179**

Out[29]= {0.0184902, 0.00369805}

The first element of the output tuple, 0.0184902*, is the sum of the relative errors appearing above, while the second element* 0.00369805*, is the average of the relative errors above. In any case, we see that the relative error is below* 0.04% *on average, and one can actually increase the precision of* `VerifyExpansion` *to check that. One can use the option* `PrecisionSettings` *which is a tuple of the form* $\{n_{max}, \epsilon\}$*, where* n_{max} *is the value of* n *used to compute numerically the errors, and* ϵ *is a positive real number used as a threshold, i.e., if the relative error is greater than* ϵ*, the expansion will be deemed erroneous, and instead of the usual tuple of absolute/average relative errors,* `VerifyExpansion` *will return* `Failed` *instead. By default, the value of* `PrecisionSettings` *is* $\{1000, 0.02\}$*. Let us change for instance and take* $n_{max} = 1200$ *instead:*

In[30]:= **check1B = VerifyExpansion[sum1, asymp1, n, PrecisionSettings → {1200, 0.02}]**

Out[30]= {0.0156331, 0.00312661}

As we see, the errors decrease, which is the expected behaviour of the expansion. One could alternatively also try to expand to one order higher the sum and then check numerically:

In[31]:= **asymp1B = AsymptoticsMellint[mel1[[1]], x, n, 8]**

Out[31]= $$\frac{4}{\pi}\left(\frac{\sqrt{\pi}(-4)^n}{5n^{5/2}} - \frac{\sqrt{\pi}(-1)^n 2^{2n-3}}{n^{7/2}} - \frac{11\sqrt{\pi}(-1)^n 2^{2n-7}}{n^{9/2}} + \frac{13\sqrt{\pi}(-1)^n 2^{2n-10}}{n^{11/2}} + \frac{32539\sqrt{\pi}(-1)^n 2^{2n-15}}{5n^{13/2}} + \frac{359493\sqrt{\pi}(-1)^n 2^{2n-18}}{5n^{15/2}}\right)$$

In[32]:= **check1B1 = VerifyExpansion[sum1, asymp1B, n]**

Out[32]= $Failed

In[33]:= **check1B2 = VerifyExpansion[sum1, asymp1B, n,**
PrecisionSettings → {800, 0.02}]

Out[33]= {0.0230507, 0.00384179}

In[34]:= **check1B3 = VerifyExpansion[sum1, asymp1B, n,**
PrecisionSettings → {1300, 0.02}]

Out[34]= {0.0142557, 0.00237596}

Here we see the limits of numerical computations: indeed, for $n_{max} = 1000$ *the*

function actually returns a `$Failed`*, whereas for* $n_{max} = 800$ *and* $n_{max} = 1300$*, i.e., one significantly less and the other significantly more, the check follows through, with the expected behaviour that the error is smaller for* $n_{max} = 1300$ *than for* $n_{max} = 800$*. This means that numerical checking is highly reliant on the* n_{max} *we take and the error threshold we are willing to accept to call the result "correct" — therefore,* `VerifyExpansion` *should be seen only for what it is, i.e., a practical tool for experimenting and testing.*

2.5.6. Some examples of asymptotics

We present in this section some examples of asymptotics, going from some for which the Mellin transform has been treated in [16] to new examples that could not be achieved before using the so far available methods.

Example 2.5.10. *We consider the binomial sum (5.34) p. 27 from [16]:*

$$\mathcal{BS}_{\left((1,-2,0),\left(0,-\frac{1}{2},1\right),(0,1,2)\right)}(n) = \sum_{i=1}^{n} \frac{(-1)^i}{i^2} \binom{2i}{i} \sum_{j=1}^{i} \frac{\left(-\frac{1}{2}\right)^j}{j} \sum_{k=1}^{j} \frac{1}{k^2}. \tag{2.5.74}$$

We define it in `Mathematica` *and then compute its Mellin inverse first:*

In[35]:=
$$sum2 = GS\left[\left\{(\mathbf{-2})^{VarGL}\binom{\mathbf{2}\,VarGL}{VarGL}, \frac{\left(-\frac{\mathbf{1}}{\mathbf{2}}\right)^{VarGL}}{VarGL}, \frac{\mathbf{1}}{VarGL^{\mathbf{2}}}\right\}, \mathbf{n}\right];$$

In[36]:=
$$\mathbf{mel2 = SumToMellin[sum2, C, x, ToGLbBasis \to False]}$$

Out[36]=
$$\left\{\frac{8}{\pi}\left(-Li_3\left(-\frac{1}{2}\right) - 2Li_3\left(\frac{1}{3}\right) + Li_2\left(-\frac{1}{2}\right)(\log\left(\frac{3}{2}\right)) + \frac{1}{3}(\log\left(\frac{3}{2}\right))^3\right)\right.$$
$$\times Mellin\left[(-8)^n x^n - 1, \frac{\sqrt{x}}{\sqrt{1-x}(8x+1)}\right]$$
$$\left. - \frac{4\,Mellin\left[4^n x^n - 1, \frac{\sqrt{x}Hwb[\{fw7,fw1,fw1\},x]-\sqrt{x}z2Hwb[\{fw7\},x]}{\sqrt{x+2}(4x-1)}\right]}{\pi}, \{\}\right\}$$

We already see that the dominating term will be the one with the $(-8)^n$*, so asymptotics will be coming from this term:*

In[37]:=
$$\mathbf{asymp2 = AsymptoticsMellint[mel2[[1]], x, n, 7]}$$

Out[37]=
$$\frac{8}{\pi}\left(-\frac{13\sqrt{\pi}(-1)^n 2^{3n-3}}{81n^{3/2}} - \frac{13\sqrt{\pi}(-1)^n 2^{3n-7}}{243n^{5/2}} + \frac{2195\sqrt{\pi}(-1)^n 2^{3n-10}}{2187n^{7/2}}\right.$$
$$+ \frac{806953\sqrt{\pi}(-1)^n 2^{3n-15}}{19683n^{9/2}} + \frac{5675159\sqrt{\pi}(-1)^n 2^{3n-18}}{19683n^{11/2}}$$
$$\left. - \frac{203235593\sqrt{\pi}(-1)^n 2^{3n-22}}{177147n^{13/2}} + \frac{\sqrt{\pi}(-8)^n}{9\sqrt{n}}\right)$$
$$\times \left(-Li_3\left(-\frac{1}{2}\right) - 2Li_3\left(\frac{1}{3}\right) + Li_2\left(-\frac{1}{2}\right)(\log(3) - \log(2)) + \frac{1}{3}(\log(3) - \log(2))^3\right)$$

Actually, if one wants to see the subdominant asymptotics also, one can do so very

easily by setting the option `ExtendedAsymptotics→True`*, valid for both functions* `AsymptoticsMellint` *and* `AsymptoticsSum`[78]*:*

```
In[38]:= asymp2B = AsymptoticsMellint[mel2[[1]], x, n, 7,
           ExtendedAsymptotics → True]
```

$$\begin{aligned}
\text{Out[38]}= \; & \frac{8}{\pi}\left(-\frac{13\sqrt{\pi}(-1)^n 2^{3n-3}}{81n^{3/2}}-\frac{13\sqrt{\pi}(-1)^n 2^{3n-7}}{243n^{5/2}}+\frac{2195\sqrt{\pi}(-1)^n 2^{3n-10}}{2187n^{7/2}}\right.\\
& +\frac{806953\sqrt{\pi}(-1)^n 2^{3n-15}}{19683n^{9/2}}+\frac{5675159\sqrt{\pi}(-1)^n 2^{3n-18}}{19683n^{11/2}}\\
& \left.-\frac{203235593\sqrt{\pi}(-1)^n 2^{3n-22}}{177147n^{13/2}}+\frac{\sqrt{\pi}(-8)^n}{9\sqrt{n}}-\frac{\pi}{12}\right)\\
& \times\left(-Li_3\left(-\frac{1}{2}\right)-2Li_3\left(\frac{1}{3}\right)+Li_2\left(-\frac{1}{2}\right)(\log(3)-\log(2))+\frac{1}{3}(\log(3)-\log(2))^3\right)\\
& -\frac{4}{\pi}\left(Integrate\left[\frac{e^{-w}}{4\sqrt{\frac{e^{-w}}{4}+2}\,(e^{-w}-1)}\right.\right.\\
& \times\left(\frac{1}{2}\sqrt{e^{-w}}\,GL\left[\left\{\frac{1}{\sqrt{(3-VarGL)\,VarGL}},\frac{1}{\sqrt{(1-VarGL)\,VarGL}},\frac{1}{\sqrt{(1-VarGL)\,VarGL}}\right\},1-\frac{e^{-w}}{4}\right]\right.\\
& \left.\left.-\frac{1}{12}\pi^2\sqrt{e^{-w}}\,GL\left[\left\{\frac{1}{\sqrt{(3-VarGL)\,VarGL}}\right\},1-\frac{e^{-w}}{4}\right]\right),\{w,-2ln2,\infty\},\right.\\
& \left.PrincipalValue\to True\right]\\
& +\frac{(144+7\pi^2)\sqrt{\pi}2^{2n-4}}{81n^{5/2}}-\frac{(864+89\pi^2)\sqrt{\pi}2^{2n-8}}{27n^{7/2}}-\frac{(12065\pi^2-122544)\sqrt{\pi}2^{2n-11}}{243n^{9/2}}\\
& -\frac{(3159289\pi^2-11135808)\sqrt{\pi}2^{2n-16}}{729n^{11/2}}-\frac{(59742275\pi^2-188456688)\sqrt{\pi}2^{2n-19}}{405n^{13/2}}\\
& \left.-\frac{\pi^{5/2}2^{2n-1}}{27n^{3/2}}\right)
\end{aligned}$$

As we clearly see, the subdominat constants are now visible, and in partcular one of them is represented in the form of an integral involving GRPL functions.

Example 2.5.11. *We consider the binomial sum (5.44) p. 29 from [16]:*

$$\overline{\mathcal{BS}}_{((-1,-2,1,1+\lambda),(1,-1,1,\lambda^3))}(n)=\sum_{i=1}^{n}\frac{(-2)^i}{(i+1)\binom{2i}{i}}\sum_{j=1}^{i}\binom{2j}{j}\frac{(-1)^j}{j^3}. \tag{2.5.75}$$

We define it in `Mathematica`

```
In[39]:= sum3 = GS[{(-2)^VarGL/((VarGL+1)Binomial[2VarGL, VarGL]), (-1)^VarGL Binomial[2VarGL, VarGL]/VarGL^3}, n];
```

and then compute its Mellin inverse first:

```
In[40]:= mel3 = SumToMellin[sum3, C, x, ToGLbBasis → False]
```

$$\begin{aligned}
\text{Out[40]}= \; & \left\{\left(2Li_3\left(\frac{1}{2}\left(\sqrt{5}-1\right)\right)-\frac{6}{5}z2\left(\log\left(\sqrt{5}-1\right)-\log(2)\right)\right.\right.\\
& \left.\left.-\frac{6z3}{5}+\frac{2}{3}\left(\log\left(\sqrt{5}-1\right)-\log(2)\right)^3\right)
\end{aligned}$$

[78] See for more details the Appendix A.

$$\times Mellin\left[\left(-\frac{1}{2}\right)^{n} x^{n}-1, \frac{x\left(\sqrt{1-x}\,Hwb[\{fw3\},x]-2\right)}{\sqrt{1-x}(x+2)}\right]$$
$$-Mellin\left[2^{n}x^{n}-1, \frac{x\left(\sqrt{4x+1}\,Hwb[\{fw14,fw14,f0,f0\},x]-4Hwb[\{fw14,f0,f0\},x]\right]}{(2x-1)\sqrt{4x+1}}\right],\{\}\Bigg\}$$

Finally, we compute the asymptotic expansion with:

In[41]:= **asymp3 = AsymptoticsMellint[mel3[[1]], x, n, 7]**

Out[41]= $$\frac{23419\ 2^{n+3}}{625n^{7}}+\frac{557\ 2^{n+3}}{125n^{6}}+\frac{23\ 2^{n+2}}{25n^{5}}+\frac{2^{n+3}}{5n^{4}}$$

Now let us examine some examples of sums that do not fit in the setting of the [16] paper, e.g., that have several binomial coefficients in the numerator in succession, or altogether some generalized binomial sums with multiplicity larger than 1:

Example 2.5.12. *Let us define the following binomal sum*[79]

$$\mathcal{BS}_{\left((1,\frac{1}{2},0),(1,\frac{1}{2},0),(1,\frac{1}{2},0)\right)}(n)=\sum_{i=1}^{n}\left(\frac{1}{2}\right)^{i}\binom{2i}{i}\sum_{j=1}^{i}\left(\frac{1}{2}\right)^{j}\binom{2j}{j}\sum_{k=1}^{j}\left(\frac{1}{2}\right)^{k}\binom{2k}{k}. \quad (2.5.76)$$

We define this sum in `Mathematica`

In[42]:= $$sum4 = GS\left[\left\{\left(\frac{1}{2}\right)^{VarGL}\binom{2\,VarGL}{VarGL},\left(\frac{1}{2}\right)^{VarGL}\binom{2\,VarGL}{VarGL},\left(\frac{1}{2}\right)^{VarGL}\binom{2\,VarGL}{VarGL}\right\},n\right];$$

and then compute its Mellin inverse. We actually use here the option `UseGLCb` *of* `AsymptoticsMellint`[80] *that we set to* `True`*, because we expect that the function will not be able to represent the functions without GCPLs anyway.*

In[43]:= **mel4 = SumToMellin[sum4, C, x, ToGLbBasis → False, UseGLCb → True]**

Out[43]= $$\left\{-\frac{4C_1}{\pi^3}Mellin\left[4^{n}x^{n}-1,\frac{x}{4x-1}GLCb\left[\left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}},\{\}\right\},\left\{\sqrt{\frac{VarGL}{1-VarGL}},\{\}\right\}\right\},x\right]\right]$$
$$+\frac{8}{\pi^3}Mellin\left[8^{n}x^{n}-1,\frac{x}{8x-1}GLCb\left[\left\{\left\{\frac{1}{\sqrt{(1-\mathtt{VarGL})\mathtt{VarGL}}},\{\}\right\},\right.\right.$$
$$\left.\left.\left\{\frac{\mathtt{VarGL}}{(\mathtt{VarGL}-\frac{1}{4})},\left\{\left\{\frac{1}{\sqrt{(1-\mathtt{VarGL})\mathtt{VarGL}}},\{\}\right\},\left\{\frac{1}{\sqrt{(1-\mathtt{VarGL})\mathtt{VarGL}}}\frac{1}{-\frac{1}{2}+\mathtt{VarGL}},\{\}\right\}\right\}\right\},x\right]\right]$$
$$+\frac{2}{\pi^3}\left(C_1^2-C_2\right)Mellin\left[2^{n}x^{n}-1,\frac{\sqrt{x}}{\sqrt{1-x}(2x-1)}\right],$$
$$\left\{C_1\to GLCb\left[\left\{\left\{\frac{1}{\mathtt{VarGL}},\{\}\right\},\left\{\frac{1}{\sqrt{\mathtt{VarGL}(1-\mathtt{VarGL})}}\frac{1}{\mathtt{VarGL}-\frac{1}{2}},\{\}\right\}\right\},0\right],$$

[79] See Definition 2.1.2.

[80] The option is also of course applicable to the all-in-one `AsymptoticsSum` function.

$$\mathtt{C}_2 \to \mathtt{GLCb}\left[\left\{VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{VarGL}{VarGL-\frac{1}{4}}, \left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}, \left\{\frac{\sqrt{\frac{VarGL}{1-VarGL}}}{VarGL-\frac{1}{2}}, \{\}\right\}\right\}\right\}\right\}, 0\right]\right\}\right\}$$

As we see, we obtain a result that involved mostly GCPLs, except for the subdominant part $\tilde{\mathbf{M}}_2[\cdots](n)$ *which depends only on* $\frac{\sqrt{x}}{\sqrt{1-x}(2x-1)}$. *Using this result, we can compute the asymptotic series of our sum:*

In[44]:= **asymp4 = AsymptoticsMellint[mel4[[1]], x, n, 7]**

Out[44]=
$$\frac{8}{\pi^3}\left(\frac{\frac{\pi^{3/2}2^{3n+3}}{21n^{3/2}}+113\pi^{3/2}2^{3n}}{441n^{5/2}}+\frac{116929\pi^{3/2}2^{3n-4}}{9261n^{7/2}}+\frac{3338591\pi^{3/2}2^{3n-7}}{9261n^{9/2}}\right.$$
$$\left.+\frac{10966199551\pi^{3/2}2^{3n-12}}{194481n^{11/2}}+\frac{11536365172963\pi^{3/2}2^{3n-15}}{4084101n^{13/2}}\right)$$

Since no extra constants appear in terms of integral representations, we can straightforwardly apply our `VerifyExpansion` *command to gain numerical confirmation.*

In[45]:= **VerifyExpansion[sum4, asymp4, n]**

Out[45]= {0.0266561, 0.00444268}

Example 2.5.13. *Let us define the following binomial sum*[81]*:*

$$\mathcal{BS}_{\left((1,\frac{1}{2},0),(1,\frac{1}{2},0),(-1,3,0)\right)}(n)=\sum_{i=1}^{n}\left(\frac{1}{2}\right)^i\binom{2i}{i}\sum_{j=1}^{i}\left(\frac{1}{2}\right)^j\binom{2j}{j}\sum_{k=1}^{j}\frac{3^k}{k\binom{2k}{k}}. \tag{2.5.77}$$

We define this sum in `Mathematica` *and then compute its Mellin inverse:*

In[46]:= $sum5 = GS\left[\left\{\left(\frac{1}{2}\right)^{VarGL}\binom{2\,VarGL}{VarGL}, \left(\frac{1}{2}\right)^{VarGL}\binom{2\,VarGL}{VarGL}, \frac{3^{VarGL}}{VarGL\binom{2\,VarGL}{VarGL}}\right\}, \mathbf{n}\right];$

In[47]:= **mel5 = SumToMellin[sum5, C, x, UseGLCb → True]**

Out[47]=
$$\left\{\frac{3}{\pi^2}Mellin\left[3^n x^n-1, \frac{x}{3x-1}\mathtt{GLCb}\left[\left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}, \left\{\frac{VarGL}{VarGL-\frac{2}{3}}, \left\{\left\{\frac{1}{\sqrt{1-VarGL}\left(VarGL-\frac{4}{3}\right)}, \{\}\right\}, \left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}\right\}\right\}, x\right]\right]\right.$$
$$-\frac{4C_1}{\pi^2}Mellin\left[4^n x^n-1, \frac{x}{4x-1}\mathtt{GLCb}\left[\left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}, \left\{\frac{\sqrt{\frac{VarGL}{1-VarGL}}}{VarGL-\frac{1}{2}}, \{\}\right\}\right\}, x\right]\right]$$
$$\frac{2\,(C_1C_2-C_3)}{\pi^2}Mellin\left[2^n x^n-1, \frac{\sqrt{x}}{\sqrt{1-x}(2x-1)}\right],$$
$$\left.\left\{C_1 \to \mathtt{GLCb}\left[\left\{VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{1}{\sqrt{1-VarGL}\left(VarGL-\frac{4}{3}\right)}, \{\}\right\}\right\}\right\}, 0\right],\right.$$

[81] See Definition 2.1.2.

$$\texttt{C}_2 \to \texttt{GLCb}\left[\left\{ VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{\sqrt{\frac{VarGL}{1-VarGL}}}{VarGL-\frac{1}{2}}, \{\}\right\}\right\}\right\}, 0\right],$$

$$\texttt{C}_3 \to \texttt{GLCb}\left[\left\{ VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{VarGL}{VarGL-\frac{2}{3}}, \left\{\left\{\frac{1}{\sqrt{1-VarGL}\left(VarGL-\frac{4}{3}\right)}, \{\}\right\}, \left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}\right\}\right\}\right\}, 0\right]\right\}\right\}$$

We see here that we are lucky and the dominating term, with the 4^n coefficient, does not contain the most complicated GCPL function. We compute the asymptotics of our sum:

In[48]:= **asymp5 = AsymptoticsMellint[mel5[[1]], x, n, 7]**

Out[48]= $$-\frac{4}{\pi^2}\left(\frac{7558409753\pi \texttt{C}_1 2^{2n-15}}{243\texttt{n}^7} + \frac{121324067\pi \texttt{C}_1 2^{2n-12}}{243\texttt{n}^6} + \frac{1591633\pi \texttt{C}_1 2^{2n-10}}{81\texttt{n}^5} + \frac{6703\pi \texttt{C}_1 2^{2n-6}}{27\texttt{n}^4} + \frac{469\pi \texttt{C}_1 2^{2n-4}}{27\texttt{n}^3} + \frac{7\pi \texttt{C}_1 2^{2n-1}}{9\texttt{n}^2} + \frac{\pi \texttt{C}_1 2^{2n+1}}{3\texttt{n}}\right)$$

We see however that the constant C_1 remains in the expansion. We use the function `GLCbToIntegral` *to try to get an integral representation:*

In[49]:= **GLCbToIntegral**$\left[\textbf{GLCb}\left[\left\{ VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{1}{\sqrt{1-VarGL}\left(VarGL-\frac{4}{3}\right)}, \{\}\right\}\right\}\right\}, 0\right]\right]$

Out[49]= $$GLb\left[\left\{\frac{1}{\sqrt{1-VarGL}\left(VarGL-\frac{4}{3}\right)}\right\}, 0\right]$$

In this case, the constant is not a real GCPL constant but a disguised GRPL constant, and because it is quite simple it can even be computed explicitely:

$$C_1 \to -\frac{2\pi}{3}.$$

The final expansion of our function is therefore:

$$\mathcal{BS}_{\left((1,\frac{1}{2},0),(1,\frac{1}{2},0),(-1,3,0)\right)}(n) \underset{n\to+\infty}{=} \frac{2^{2n+4}}{3\sqrt{3}n} + \frac{7\ 2^{2n+2}}{9\sqrt{3}n^2} + \frac{469\ 2^{2n-1}}{27\sqrt{3}n^3} + \frac{6703\ 2^{2n-3}}{27\sqrt{3}n^4} + \frac{1591633\ 2^{2n-7}}{81\sqrt{3}n^5} + \frac{121324067\ 2^{2n-9}}{243\sqrt{3}n^6} + \frac{7558409753\ 2^{2n-12}}{243\sqrt{3}n^7} + \mathcal{O}\left(\frac{4^n}{n^8}\right).$$

We once again can double-check with `VerifyExpansion`[82]*:*

In[50]:= **VerifyExpansion[sum5, asymp5, n]**

[82] Where we considered we replaced in `asymp5` C_1 with its explicit value.

Out[50]= $\{0.0263505, 0.00376436\}$

Example 2.5.14. *Let us define the following generalized binomial sum*[83]*:*

$$\mathcal{BS}_{((-1,2,1),(2,-1,1))}(n) = \sum_{i=1}^{n} \frac{2^i}{i\binom{2i}{i}} \sum_{j=1}^{i} \frac{(-1)^j}{j} \binom{2j}{j}^2. \tag{2.5.78}$$

We define this sum in Mathematica *and then compute its Mellin inverse:*

In[51]:= $sum6 = GS\left[\left\{\frac{\mathbf{2}^{VarGL}}{VarGL\binom{\mathbf{2}\,VarGL}{VarGL}}, \frac{(\mathbf{-1})^{VarGL}\binom{\mathbf{2}\,VarGL}{VarGL}^{\mathbf{2}}}{VarGL}\right\}, \mathbf{n}\right];$

In[52]:= $\mathbf{mel6 = SumToMellin[sum6, C, x, UseGLCb \to True]}$

Out[52]=
$$\left\{\frac{8}{\pi^2} Mellin\left[(-8)^{\mathtt{n}}\mathtt{x}^{\mathtt{n}} - 1, \frac{\mathtt{x}}{8\mathtt{x}+1}\mathtt{GLCb}\left[\left\{\left\{\frac{1}{\sqrt{1 - VarGL}\,VarGL}, \{\}\right\}, \left\{\frac{VarGL}{VarGL + \frac{1}{16}}, \left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}, \left\{\frac{1}{VarGL}, \left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}\right\}\right\}\right\}\right\}\right\}, \mathtt{x}\right]\right]$$
$$-\frac{\mathtt{C}_1}{\pi^2} Mellin\left[2^{-\mathtt{n}}\mathtt{x}^{\mathtt{n}} - 1, \frac{1}{\sqrt{1-\mathtt{x}}(\mathtt{x}-2)}\right],$$
$$\left\{\mathtt{C}_1 \to \mathtt{GLCb}\left[\left\{VarGL, \left\{\left\{\frac{1}{VarGL}, \{\}\right\}, \left\{\frac{VarGL}{VarGL + \frac{1}{16}}, \left\{\left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}, \{\}\right\}, \left\{\frac{1}{VarGL}, \left\{\frac{1}{\sqrt{(1-VarGL)\,VarGL}}\right\}\right\}\right\}\right\}\right\}, 0\right]\right\}\right\}$$

It suffices now to compute the asymptotics, which is guaranteed by Corollary 2.5.1.1:

In[53]:= $\mathbf{asymp6 = AsymptoticsMellint[mel6[[1]], x, n, 7]}$

Out[53]=
$$\frac{8}{\pi^2}\left(\frac{\pi^{3/2}(-1)^{\mathtt{n}}2^{3\mathtt{n}+4}}{153\mathtt{n}^{5/2}} - \frac{637\pi^{3/2}(-1)^{\mathtt{n}}2^{3\mathtt{n}+1}}{23409\mathtt{n}^{7/2}} - \frac{1205159\pi^{3/2}(-1)^{\mathtt{n}}2^{3\mathtt{n}-3}}{3581577\mathtt{n}^{9/2}}\right.$$
$$\left. - \frac{68993759\pi^{3/2}(-1)^{\mathtt{n}}2^{3\mathtt{n}-6}}{60886809\mathtt{n}^{11/2}} + \frac{936852166115\pi^{3/2}(-1)^{\mathtt{n}}2^{3\mathtt{n}-11}}{9315681777\mathtt{n}^{13/2}}\right)$$

Once again we can look at the numerical precision of the approximation:

In[54]:= $\mathbf{VerifyExpansion[sum6, asymp6, n]}$

Out[54]= $\{0.00752506, 0.00150501\}$

[83] See Definition 2.1.2.

3. An overview to solve first order systems of differential equations

Developped in the first half of the XX$^{\text{th}}$ century, quantum field theory (QFT) is one of the most powerful modern theories that describes the world on a very small scale. Designed originally to deal with relativistic quantum mechanics, it came out to be a general framwork for dealing with many-particle problems that could also provide a consistent quantum mechanical description for most of the fundamental forces – electromagnetism, the strong force and the weak one. At the heart of the theory lies the notion of scattering amplitude, i.e., the probability amplitude for scattering from some initial state $|i\rangle$ with m particles to some final state $|f\rangle$ with n particles. In interacting theories, scattering amplitudes are often defined perturbatively and computing them to higher order is necessary to better understand the theory and make more precise experimental predictions.

For involved QFTs such as Yang-Mills theories, these computations can be particularly difficult when we are going to higher numbers of loops, and are usually carried out in the following way: taking the traditional approach, one can write all the Feynman diagrams with their particular color, momentum and Dirac structure and then reduce them all to some master integrals [35–42]. These will depend typically on two parameters: some x that might be a Feynman parameter or a Landau variable on which the center-of-mass energy s, but also the ϵ variable, coming from the dimensional regularisation (or any other regularisation scheme) depends. Besides reducing many Feynman diagrams with a given topology to master integrals, the so-colled "integration by parts" (IBP) method provides in addition a system of first order differential equations in the x variable for the said master integrals [43,56,109,110]. Given such as system, a natural question is how one can solve it order by order in the ϵ parameter.

In some special cases one can bring the system in a particular form [43–47] that enables one to read off solutions straightforwardly. In this chapter, following closely [111], we elaborate two classical methods that allow us to carry the solving order by order in ϵ if such a transformation is not possible. After some preprocessing of the system that allows to reduce it to simpler subproblems (Section 3.2), the first method consists in expanding the system in ϵ and solving the subsequent series of systems by uncoupling them [48] (Section 3.3), wheareas the second one uncouples them first and then proceeds to expand the obtained higher order differential

equation in ϵ to solve it iteratively [50] (Section 3.4), using in particular difference ring and field machinery and uncoupling algorithms [1,112–114]. Assuming in addition that the integrals can be expanded in a power series, one can apply also recurrence solving techniques for the coefficients of the expansion. These methods allowed one to tackle various physical problems such as [22,28,51–60,62]. These methods are mostly designed for systems obeying some constraints, in particular they must be first-order factorizable. In the case it cannot be achieved, a method called the large moment method, introduced in [61–63] and applied for example in [62,64–68]. The setup presented in this chapter allows us to present in the next Chapter 4 our own method of analysis of differential systems that aims at being as general as possible, building upon the elements presented in this chapter.

From now on, we denote the master integrals (MIs) of our system as $\mathcal{I}_i(x,\epsilon)$, and we define the vector containing all of them as $\mathcal{I}(x,\epsilon) := (\mathcal{I}_1(x,\epsilon),\dots,\mathcal{I}_n(x,\epsilon)^\top$, where n is the total number of master integrals we are dealing with as well as the size of our system obtained from IBP [43,56,109,110]. The latter is defined therefore as follows:

$$\frac{\mathrm{d}}{\mathrm{d}x}\mathcal{I}(x,\epsilon) = \mathcal{M}(x,\epsilon)\mathcal{I}(x,\epsilon) + \mathcal{R}(x,\epsilon) \tag{3.0.1}$$

where $\mathcal{M}(x,\epsilon)$ is the matrix of coefficients and $\mathcal{R}(x,\epsilon) := (\mathcal{R}_1(x,\epsilon),\dots,\mathcal{R}_n(x,\epsilon))^\top$ is the inhomogeneous part of the system. Due to the IBP reduction principles, $\mathcal{M}(\epsilon,x) \in \mathcal{M}_n(\mathbb{K}(x,\epsilon))$, i.e., the coefficient matrix is defined over the field of rational functions in ϵ and x over $\mathbb{K}$, where $\mathbb{K}$ is as usual a computable subfield of $\mathbb{R}$ (for instance $\mathbb{Q}$). As for the inhomogeneous part, it might contain simpler Feynman integrals obtained when integration by parts produces integrals that are not part of our $\mathcal{I}(x,\epsilon)$ master integrals — those will be called base case integrals (BI) in the following, and we suppose that some expansion of those is already known, or can be obtained[84].

3.1. Solving hypotheses

We follow the approach of [48]. In order to apply the method proposed below to solve our system, the following assumptions must hold:

- Each element of $\mathcal{R}(x,\epsilon)$ can be Laurent-expanded in ϵ and therefore the solutions $\mathcal{I}(x,\epsilon)$ that we search are also assumed to be expandable in the

[84] For instance, one could apply recursively the IBP method and/or use symbolic summation/integration algorithms (see [62] and the literature therein) in order to obtain new systems or closed form representations.

same way, i.e.,

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

- The coefficient matrix $\mathcal{M}(x,\epsilon)$ has no poles around $\epsilon = 0$, i.e., it has a Taylor expansion of the following form:

$$\mathcal{M}(x,\epsilon) = \sum_{i=0}^{\infty} \mathcal{M}^{(i)}(x)\epsilon^i. \tag{3.1.1}$$

  This form of expansion can actually always be achieved by index-shift. More precisely, suppose that $\mathcal{M}(x,\epsilon)$ has a pole of order k in ϵ, i.e.,

$$\mathcal{M}(x,\epsilon) = \sum_{i=-k}^{\infty} \mathcal{M}^{(i)}(x)\epsilon^i = \frac{\mathcal{M}^{(-k)}(x)}{\epsilon^k} + \cdots + \sum_{i=0}^{\infty} \mathcal{M}^{(i)}(x)\epsilon^i.$$

  Then, since $\mathcal{M}(x,\epsilon)$ only appears in the equation (3.0.1) as the matrix multiplying the MI vector $\mathcal{I}(x,\epsilon)$, we can do the following redefinition of the latter:

$$\begin{aligned}\sum_{i=-k}^{\infty} \mathcal{M}^{(i)}(x)\epsilon^i \mathcal{I}(x,\epsilon) &= \sum_{i=-k}^{\infty} \mathcal{M}^{(i)}(x)\epsilon^{i+k} \left(\frac{\mathcal{I}(x,\epsilon)}{\epsilon^k}\right) \\ &= \underbrace{\tilde{M}(x,\epsilon)}_{=:\sum_{i=0}^{\infty} \mathcal{M}^{(i-k)}(x)\epsilon^i} \times \underbrace{\tilde{\mathcal{I}}(x,\epsilon)}_{=:\frac{\mathcal{I}(x,\epsilon)}{\epsilon^k}}.\end{aligned}$$

  This means that poles of $\mathcal{I}(x,\epsilon)$ will be shifted. But this is not a problem because of the previous assumption. Indeed if $\mathcal{M}(x,\epsilon)$ has poles around $\epsilon = 0$, we index-shift it so that the poles of $\mathcal{I}_i(x,\epsilon)$ are themselves shifted, this is fine, since we suppose they have poles anyway. In addition, after such a shift has been achieved, we suppose that

$$\det \mathcal{M}^{(0)}(x) \neq 0.$$

- The given coefficients $\mathcal{R}_i^{(j)}(x)$ are polynomial expressions in terms of hyperexponential functions and iterated integrals over hyperexponential functions, and therefore we seek for solution coefficients $\mathcal{I}_i^{(j)}(x)$ in the same function space.

Let us now define what is exactly an hyperexponential function.

Definition 3.1.1 (Hyperexponential function)**.** *Let $\mathbb{K}$ be some (computable) field. A non-zero function $f(x)$ over $\mathbb{K}$ is called* ***hyperexponential*** *if there exists some non zero $r(x) \in \mathbb{K}(x)$ such that*[85]

$$\frac{\mathrm{d}_x f(x)}{f(x)} = r(x).$$

We will denote the set of hyperexponential functions over $\mathbb{K}$ with $\mathcal{H}^e(\mathbb{K})$.

Examples: The class of hyperexponential functions is relatively large, it includes for instance:

- all rational functions over $\mathbb{K}$, i.e. $\mathbb{K}(x) \subset \mathcal{H}^e(\mathbb{K})$;
- more generally, all rational functions to some power, i.e.

  $$\{f^q(x) \mid f(x) \in \mathbb{K}(x),\ q \in \mathbb{K}\} \subset \mathcal{H}^e(\mathbb{K}).$$

  In our case and for our computations, this reduces usually to rational functions to some rational power, i.e. $q \in \mathbb{Q}$;
- the exponential function exp itself.

The last point suggests to write another possible definition of an hyperexponential function $f(x)$ as an exponentiated integral over a rational function $r(x)$:

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

From the hyperexponential function definition one might extend naturally to iterated integrals over such functions.

Definition 3.1.2 (Iterated integral over hyperexponential functions)**.** *Let $\mathbb{K}$ be a (computable) field, $l_0, \ldots, l_n \in \mathbb{K}$ and $f_1, \ldots, f_n \in \mathcal{H}^e(\mathbb{K})$. Then the following integral*

$$I(x) = \int_{l_0}^{x} \mathrm{d}x_1 f_1(x_1) \int_{l_1}^{x_1} \mathrm{d}x_2 f_2(x_2) \cdots \int_{l_{n-1}}^{x_{n-1}} \mathrm{d}x_n f_n(x_n) \text{ with } l_0, \ldots, l_n \in \mathbb{K}$$

is called an ***iterated integral over hyperexponential functions*** *(IOH) or also* ***d'Alembertian function****. We will denote the set of IOH over $\mathbb{K}$ with $\mathcal{IH}^e(\mathbb{K})$.*

The set $\mathcal{H}^e(\mathbb{K})$ is also especially nice since it is in particular stable under:

[85] We note as an abreviation $\frac{\mathrm{d}}{\mathrm{d}x} \equiv \mathrm{d}_x$.

- Multiplication: take $f(x), g(x) \in \mathcal{H}^e(\mathbb{K})$ such that there exists $r(x), s(x) \in \mathbb{K}(x)$ such as:
$$\frac{\mathrm{d}_x f}{f} = r, \ \frac{\mathrm{d}_x g}{g} = s.$$
Then
$$\frac{\mathrm{d}_x(fg)}{fg} = \frac{(\mathrm{d}_x f)g}{fg} + \frac{f(\mathrm{d}_x g)}{fg} = \frac{\mathrm{d}_x f}{f} + \frac{\mathrm{d}_x g}{g} = r + s \in \mathbb{K}(x). \quad \square$$

- Inversion: take $f(x) \in \mathcal{H}^e(\mathbb{K})$ with $r(x) \in \mathbb{K}(x)$ such that $\frac{\mathrm{d}_x f}{f} = r$. Then
$$\frac{\mathrm{d}_x\left(\frac{1}{f}\right)}{\left(\frac{1}{f}\right)} = f \times \left(-\frac{\mathrm{d}_x f}{f^2}\right) = -\frac{\mathrm{d}_x f}{f} = -r \in \mathbb{K}(x). \quad \square$$

- Differentiation: take $f(x) \in \mathcal{H}^e(\mathbb{K})$ with $r(x) \in \mathbb{K}(x)$ such that $\frac{\mathrm{d}_x f}{f} = r$. Then
$$\frac{\mathrm{d}_x^2 f}{\mathrm{d}_x f} = \frac{\mathrm{d}_x(rf)}{rf} = \frac{(\mathrm{d}_x r)f}{rf} + \frac{r(\mathrm{d}_x f)}{rf} = \frac{\mathrm{d}_x r}{r} + r \in \mathbb{K}(x). \quad \square$$

Moreover, due to the structure of the shuffle algebra of iterated integrals over hyperexponential functions[86], any polynomial expression out of hyperexponential functions and IOH can always be written as a linear combination of the form

$$f_1(x)I_1(x) + \cdots + f_n(x)I_n(x) \text{ with } h_i(x) \in \mathcal{H}^e(\mathbb{K}), \ I_i(x) \in \mathcal{IH}^e(\mathbb{K}).$$

Remark 3.1.1. *The functions that are not included in this set are for instance some classes of special functions, such as the elliptic functions. We remark further that the harmonic polylogarithm and their extensions in Chapter 2 fill into this class.*

3.2. Triangularization

When applying the IBP method [43, 56, 109, 110], we end up with a system with a rather large size n, usually of order $n \sim 500$ or higher. Nevertheless, from a physical point of view, there is an argument that allows one to reorganize it in a way that is advantageous for us. Indeed, the MI are classified in what is called "sectors", i.e., sets of maximum number of non-vanishing propagators in a single Feynman graph. When some propagator is absent, this defines a sub-sector, and a differential equation for a MI only contains integrals from the same sector or its

[86] See for instance [115].

sub-sectors. What this technically entails at our level is that one can find several sub-systems where the MIs depend only on themselves and on MIs from lower sub-systems (so that they can be considered effectively as BIs in this sub-system). More precisely, one can perform the following pre-processing step:

Step 0: We "triangularize" the system in n smaller sub-systems where the ith subsystem of coupled MIs depend only on the said MIs and on those from the systems 1 through $i-1$.

Remark 3.2.1. *In Section 4.1 of the next Chapter 4 we will give more details, and present and algorithm to find the actual splitting into such subsystems.*

3.3. Solving iteratively the system

Based on [48] we present a method to solve the system order by order in ϵ, using first an expansion in ϵ to get simpler systems that can be solved afterwards.

3.3.1. Defining recursive sub-systems

After triangularization in Section 3.2 we consider now one of the subsystems, i.e., take the MIs $\tilde{\mathcal{I}} = (\tilde{\mathcal{I}}_1(x), ..., \tilde{\mathcal{I}}_n(x))$ ($n \lesssim 15$) with:

$$\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{I}}(x,\epsilon) = \tilde{\mathcal{M}}(x,\epsilon)\tilde{\mathcal{I}}(x,\epsilon) + \tilde{\mathcal{R}}(x,\epsilon). \tag{3.3.1}$$

Given the expansion assumptions of the previous section, we can Laurent-expand the MI, BI and the coefficient matrix[87]:

$$\tilde{\mathcal{M}} = \sum_{j=0}^{\infty} \tilde{M}^{(j)}\epsilon^j, \quad \tilde{\mathcal{R}} = \sum_{j=-k}^{\infty} \mathcal{R}_i^{(j)}\epsilon^j, \quad \tilde{\mathcal{I}} = \sum_{j=-k}^{\infty} \mathcal{I}_i^{(j)}\epsilon^j. \tag{3.3.2}$$

This allows us to state the first step of the method:

Step 1: Plugging (3.3.2) in (3.3.1), we can collect the terms with the same powers of ϵ, which gets us an infinite-dimensional series of differential equation

[87] From now on, for simplicity, we will omit the (x) for the coefficients.

systems (DES):

$$\begin{cases} \mathrm{d}_x\tilde{\mathcal{I}}^{(-l)} &= \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l)} + \tilde{\mathcal{R}}^{(-l)} & \epsilon^{-l} \\ \mathrm{d}_x\tilde{\mathcal{I}}^{(-l+1)} &= \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l+1)} + (\tilde{\mathcal{M}}^{(1)}\tilde{\mathcal{I}}^{(-l)}) + \tilde{\mathcal{R}}^{(-l+1)} & \epsilon^{-l+1} \\ \vdots & \vdots & \vdots \\ \mathrm{d}_x\tilde{\mathcal{I}}^{(k)} &= \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(k)} + \left(\sum_{i=1}^{k+l}\tilde{\mathcal{M}}^{(i)}\tilde{\mathcal{I}}^{(k-i)}\right) + \tilde{\mathcal{R}}^{(k)} & \epsilon^{k} \\ \vdots & \vdots & \vdots \end{cases} \tag{3.3.3}$$

By doing this step, we have eliminated the ϵ dependence from the system, so what remains is a set of recursively dependent systems of first order that depend only on x. Let us see how we can iteratively solve these in a bottom-up approach. We will take for example the first system of the set, i.e.,

$$\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{I}}^{(-l)} = \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(-l)} + \tilde{\mathcal{R}}^{(-l)}. \tag{3.3.4}$$

Remark 3.3.1. *Even though it seems the easiest choice, it is actually perfectly representative of the general method. Indeed, we see that in all the subsystems, the coefficient matrix is always $\mathcal{M}^{(0)}$, and only the inhomogeneous part changes, including each time new contributions from the previous MI coefficients $\tilde{\mathcal{I}}^{(k)}$. Therefore, solving this once will provide us with the general steps and solutions to solve each higher DES.*

3.3.2. Uncoupling the system

In order to solve the system, the first step is to uncouple it, i.e., to transform a first-order system of n equations in n MIs into a single inhomogeneous higher-order differential equation (HODE) in one of the MIs, say $\tilde{\mathcal{I}}_1^{(-l)}$, where the inhomogeneous part depends on the inhomogeneous parts of the system and their derivatives, combined with $n-1$ equations relating linearly over the field of rational functions the other MIs to the first one, its derivatives and the inhomogeneous part of the system. In all those linear relations, the coefficients are from the field $\mathbb{K}(x)$. In order to do the uncoupling step, we use the packages `OreSys` [116], written by Stefan Gerhold and optimised by Carsten Schneider for our problem.

Step 2: Using the `OreSys` package, we uncouple the sub-system:

$$(3.3.4) \overset{\text{OreSys}}{\Longrightarrow} \begin{cases} \boxed{\sum_{k=0}^{m} p_k(x)\mathrm{d}_x^k \tilde{\mathcal{I}}_1^{(-l)}(x) = r(x)} \\ \hookrightarrow \text{HODE in } \tilde{\mathcal{I}}_1^{(-l)}(x),\ p_i(x) \in \mathbb{K}(x) \\ r(x) = \sum_{i=0}^{p}\sum_{j=1}^{m} r_{i,j}(x)\mathrm{d}_x^i \tilde{\mathcal{R}}_j^{(-l)}(x) \\ \hookrightarrow p \in \mathbb{N},\ r_{i,j}(x) \in \mathbb{K}(x) \\ \tilde{\mathcal{I}}_j^{(-l)}(x) = \sum_{i=0}^{m-1} a_{k,i}(x)\mathrm{d}_x^i \tilde{\mathcal{I}}_1^{(-l)}(x) + \rho_k(x) \\ \hookrightarrow j \in \{2,\dots,m\},\ a_{k,i}(x) \in \mathbb{K}(x),\ \rho_k(x) \text{ like } r(x) \end{cases} \tag{3.3.5}$$

Remark 3.3.2. *There exist several possible uncoupling algorithms in the package* `OreSys` *– Gauß, Euclid, Zürcher [112–114] among others – and their efficiency vary depending on the systems.*

3.3.3. Solving the higher-order differential equation for the order k

In order to solve the HODE, we need one crucial assumption: it must be **first-order factorizable**, i.e, there must exist rational functions $\hat{p}_i(x) \in \mathbb{K}(x)$ such that we can write the homogeneous HODE in the following form:

$$(\mathrm{d}_x - \hat{p}_m(x))(\mathrm{d}_x - \hat{p}_{m-1}(x)) \cdots (\mathrm{d}_x - \hat{p}_1(x))y(x) = 0. \tag{3.3.6}$$

Solutions of the HODE respecting such a factorization in a sense that we precise below are called **d'Alembertian solutions** [95][88].

Supposing that we are able to do so, we define then the following hyperexponential functions that are the solutions of the first-order factors of the HODE:

$$\forall k \in \{1,\dots,m\},\ h_k(x) = e^{\int_{\ell_k'}^{x} \mathrm{d}y \hat{p}_k(y)} \Leftrightarrow (\mathrm{d}_x - \tilde{p}_k(x))h_k(x) = 0 \text{ and } \ell_k' \in \mathbb{K}. \tag{3.3.7}$$

The d'Alembertian solutions above are constructed iteratively using the following idea[89]: suppose that we have a simple solution for our equation (3.3.6) – and this is indeed the case, taking $y(x) = h_1(x)$ that cancels the first factor. Now, we define the ansatz $y(x) = h_1(x)\int^x \mathrm{d}x_1\, u(x_1)$, for some $u(x)$ that we want to find. Plugging this in (3.3.6) gives us:

$$\text{LHS } (3.3.6) = (\mathrm{d}_x - \hat{p}_m(x)) \cdots (\mathrm{d}_x - \hat{p}_2(x)) \left[\mathrm{d}_x \left(h_1(x) \int^x \mathrm{d}x_1\, u(x_1)\right)\right.$$

[88] There exist algorithms to check whether the HODE is factorizable and to perform it whenever it is possible [96]. For further details we refer to [117] and references therein.

[89] For more details, see for instance [95].

$$-\hat{p}_1(x)h_1(x)\int \mathrm{d}x_1\, u(x_1)\Big]$$
$$= (\mathrm{d}_x - \hat{p}_m(x))\cdots(\mathrm{d}_x - \hat{p}_2(x))\,[h_1(x)u(x)]$$

Since we know that $h_2(x)$ is a solution (up to a constant) of $(\mathrm{d}_x - \hat{p}_2(x))z(x) = 0$:

$$u(x)h_1(x) = h_2(x) \Leftrightarrow u(x) = \frac{h_2(x)}{h_1(x)},$$

we get the solution

$$y_2(x) = h_1(x)\int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)}.$$

It suffices to apply this procedure recursively to get all the other solutions $y_i(x)$, with $i \in \{3,\cdots,m\}$. One can check that the m homogeneous d'Alembertian solutions are the following ones:

$$y_1(x) = h_1(x),$$

$$y_2(x) = h_1(x)\int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)},\ \ell_0 \in \mathbb{K},$$

$$\vdots$$

$$y_i(x) = h_1(x)\int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)}\cdots\int_{\ell_{i-2}}^{x_{i-2}} \mathrm{d}x_{i-1} \frac{h_i(x_{i-1})}{h_{i-1}(x_{i-1})},\ \ell_{i-2} \in \mathbb{K},$$

$$\vdots$$

$$y_m(x) = h_1(x)\int_{\ell_0}^{x} \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)}\cdots\int_{\ell_{m-2}}^{x_{m-2}} \mathrm{d}x_{m-1} \frac{h_m(x_{m-1})}{h_{m-1}(x_{m-1})},\ \ell_{m-2} \in \mathbb{K}.$$

In particular, one can show [95] that these solutions are linearly independent and the full solution space of (3.3.7) is:

$$\mathcal{S}_h = \left\{ \sum_{i=1}^{m} c_i y_i(x) \,\middle|\, c_i \in \mathbb{K} \right\}.$$

From the algorithmic side, given that the HODE factors in first-order factors, the function `SolveDE` of the package `HarmonicSums` of J. Ablinger [18, 73] will find the full set of solutions described above with all possible simplifications, in particular down to expressions involving HPL and CPL[90].

[90] Where all the HPL/CPL were maximally simplified using identities, shuffle and stuffle algebra of the associated S-sums for instance, see for instance [12] or our Section 2.1.2.

Step 3: Factorize the HODE and solve it using `HarmonicSums`' `SolveDE` function.

Remark 3.3.3. *It might very well happen that the HODE does not fully factor into only first-order factors, and some higher-order factors remain. Then, the system might still be solvable in the same way, provided that we manage to find enough solutions of the higher-order factor. For example, suppose that we have a third order system that factored into the following form:*

$$(\mathrm{d}_x^2 + \hat{p}_{2,1}(x)\mathrm{d}_x + \hat{p}_{2,2}(x))(\mathrm{d}x - \hat{p}_1(x))y(x) = 0. \tag{3.3.8}$$

Then, supposing that one can solve with `SolveDE` [91] *or a different solver, say the standard Mathematica* `DSolve`*, the higher-order factor and get all the solutions, e.g., here two, the solutions will have mostly the same form. More precisely, let us take for example* $\hat{p}_{2,1}(x) = 1$, $\hat{p}_{2,2}(x) = -x$ *and* $\hat{p}_1(x) = \frac{x}{1+x^2}$. *While* `SolveDE` *fails to find a solution to* $,(\mathrm{d}_x^2 + \mathrm{d}_x - x)y(x) = 0$ `DSolve` *finds a two-dimensional solution space made up of Airy functions of the first and second kind:*

```
In[55]:= DSolve[D[y[x], {x, 2}] + D[y[x], x] - x * y[x] == 0, y[x], x]

Out[55]= {{y[x] → c1 e^(-x/2) AiryAi[1/4 + x] + c2 e^(-x/2) AiryBi[1/4 + x]}}
```

So defining the following solutions

$$h_{2,1}(x) = e^{-\frac{x}{2}}\mathrm{Ai}\left(x + \frac{1}{4}\right),\ h_{2,2}(x) = e^{-\frac{x}{2}}\mathrm{Bi}\left(x + \frac{1}{4}\right),$$

the solutions of the whole HODE are therefore[92]

$$y_1(x) = h_1(x) = e^{\int^x \mathrm{d}y\, \frac{x}{1+x^2}} = \sqrt{1+x^2},$$

$$y_2(x) = h_1(x)\int_{\ell_{0,1}}^x \mathrm{d}y\, \frac{h_{2,1}(x)}{h_1(x)} = \sqrt{1+x^2}\int_{\ell_{0,1}}^x \mathrm{d}y\, \frac{e^{-\frac{y}{2}}\mathrm{Ai}\left(y+\frac{1}{4}\right)}{\sqrt{1+y^2}},$$

$$y_3(x) = h_1(x)\int_{\ell_{0,2}}^x \mathrm{d}y\, \frac{h_{2,2}(x)}{h_1(x)} = \sqrt{1+x^2}\int_{\ell_{0,2}}^x \mathrm{d}y\, \frac{e^{-\frac{y}{2}}\mathrm{Bi}\left(y+\frac{1}{4}\right)}{\sqrt{1+y^2}}.$$

In the same way as for the homogeneous solutions, one can actually check that the particular solution of the first equation of (3.3.5) has the following form:

$$g(x) = h_1(x)\int_{\ell_0}^x \mathrm{d}x_1 \frac{h_2(x_1)}{h_1(x_1)} \cdots \int_{\ell_{m-2}}^{x_{m-2}} \mathrm{d}x_{m-1} \frac{h_m(x_{m-1})}{h_{m-1}(x_{m-1})} \int_{\ell_{m-1}}^{x_{m-1}} \mathrm{d}x_m \frac{r(x_m)}{h_m(x_m)}. \tag{3.3.9}$$

[91] The Kovacic's algorithm [93] is actually implemented in `SolveDE` [18] and can already find solutions outside of d'Alembertian ones for second-order factors.

[92] Related to that, a rather general machinery has been developed in [18, 57, 62, 89].

So that combining the particular and homogeneous solutions, the general solution has the following form:

$$\boxed{\tilde{\mathcal{I}}_1^{(-l)}(x) = g(x) + c_1 y_1(x) + \cdots + c_m y_m(x) \text{ with } \forall i \in \{1, \ldots, m\},\ c_i \in \mathbb{K}.} \tag{3.3.10}$$

Finally, using the physical data, e.g. the values of the MI in several particular regimes of x, we can carry out the following step:

Step 4: Fix the general solution constants using the initial conditions.

3.3.4. Solving the system at order k and iterating

Given the $(-l)$-th coefficient of our first MI in (3.3.3), all that remains is to unroll everything and get the other solutions by repeating the above steps iteratively. More precisely, we plug the solution (3.3.10) into the relations (3.3.5) relating the first MI to the others to get all those at order $(-l)$, i.e., for all $k \in \{2, \ldots, m\}$:

$$\begin{aligned}\tilde{\mathcal{I}}_k^{(-l)}(x) &= \sum_{i=0}^{m-1} a_{k,i}(x) \mathrm{d}_x^i \tilde{\mathcal{I}}_1^{(-l)}(x) + \rho_k(x) \\ &= \sum_{i=0}^{m-1} \sum_{j=1}^{m} c_j a_{k,i}(x) \mathrm{d}_x^i y_j(x) + \sum_{i=0}^{m-1} a_{k,i}(x) g(x) + \rho_k(x).\end{aligned}$$

Finally, we can plug the whole vector of solution $\tilde{\mathcal{I}}^{(-l)} = (\tilde{\mathcal{I}}_1^{(-l)}, \ldots, \tilde{\mathcal{I}}_n^{(-l)})$ into

$$\frac{d}{dx}\tilde{\mathcal{I}}^{(k)} = \tilde{\mathcal{M}}^{(0)}\tilde{\mathcal{I}}^{(k)} + \left(\sum_{i=1}^{k+l} \tilde{\mathcal{M}}^{(i)}\tilde{\mathcal{I}}^{(k-i)}\right) + \tilde{\mathcal{R}}^{(k)} \text{ with } k = -l+1. \tag{3.3.11}$$

Step 5: We reiterate the whole process, solving now the system (3.3.11).

Remark 3.3.4. *Let us note that one of the advantages of the method presented above is that we do not have to redo all the steps at each iteration. Indeed, when we do the uncoupling, we can replace the explicit inhomogeneous part $\tilde{\mathcal{R}}^{(-l)}$ by a general vector of undefined functions, e.g.* `Inhomog[(-l)]={R[1][x],...,R[n][x]}` *in Mathematica. The uncoupling is in this case generic and depends only on the matrix coefficient. Since the latter is actually always $\tilde{\mathcal{M}}^{(0)}$ as seen in (3.3.3), the uncoupling does not change and so the homogeneous solutions of the HODE remain the same. The only quantity that changes is the particular solution (3.3.9), since it depends directly on the inhomogeneous part of the system, which changes at each iteration.*

3.4. Solving iteratively the system using the Sigma package

The first method presented above to solve iteratively the system at each order of ϵ concentrates on the series of subsystems obtained for each order of ϵ, that is solved iteratively by uncoupling each of them as first-order systems in one variable. There exists another method, presented in great detail in [49, 50], that exchanges the order of step 1 and step 2, i.e., that uncouples the equation before solving it. Let us present it now.

3.4.1. Further assumptions on the subsystem

We suppose once more that the system has been triangularized before, and we are examining one of the sub-systems of the form (3.3.1) obtained as a consequence:

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

In addition we assume that $\tilde{M}(x,\epsilon)$ is invertible, and that the inhomogeneous part can be expanded as a power series in x, i.e., for each $\tilde{R}_i(x,\epsilon)$ we can write:

$$\tilde{\mathcal{R}}_i(x,\epsilon) = \sum_{n=0}^{\infty} \tilde{R}_i(n,\epsilon)x^n. \tag{3.4.1}$$

Furthermore, we suppose that the Taylor coefficients themselves can be Laurent expanded in ϵ, i.e.,

$$\tilde{R}_i(n,\epsilon) = \sum_{k=l}^{\infty} \tilde{R}_{i,k}(n)\epsilon^k \text{ with } l \in \mathbb{Z}. \tag{3.4.2}$$

More precisely, we suppose that the coefficient $R_{i,k}(n)$ can be given as an expression in terms of indefinite sums over hypergeometric products that we introduce next.

Definition 3.4.1 (hypergeometric sequence)**.** *A sequence $(f(n))_{n\geq 0}$ is called **hypergeometric** if $f(n) \neq 0$ $\forall n \geq l$ for some $l \in \mathbb{N}$ and there exists a rational function $r(x) \in \mathbb{K}(x)$ such that*

$$\forall n \geq l,\ \frac{f(n+1)}{f(n)} = r(n).$$

Remark 3.4.1. *$(f(n))_{n\geq 0}$ with $f(x) \in \mathbb{K}(x)$ without poles is an hypergeometric sequence.*

Often hypergeometric sequences can be represented by hypergeometric products:

Definition 3.4.2 (hypergeometric product)**.** *Let $f(x) \in \mathbb{K}(x)$ and $l, n \in \mathbb{N}$. If $\forall j \geq l,\ f(j) \neq 0$ and its evaluation does not introduce poles, then $\prod_{j=l}^{n} f(j)$ is called an* ***hypergeometric product*** *(HGP).*

Example 3.4.1. *Two classical examples of hypergeometric products are:*

- $n! = \prod_{i=1}^{n} i$,
- $\binom{m}{n} = \prod_{i=1}^{n} \frac{m+1-i}{i}$.
- $\binom{2n}{n} = \prod_{i=1}^{n} \frac{n+i}{i}$

Definition 3.4.3 (indefinite nested sums over HGP [50])**.** *An* ***expression in terms of indefinite nested sums over hypergeometric products (INSH)*** *in k over $\mathbb{K}$ is composed recursively by the three operations $(+, -, \cdot)$ with:*

- *elements from the rational function field $\mathbb{K}(k)$,*
- *hypergeometric products in k over $\mathbb{K}$,*
- *and sums of the form $\sum_{j=l}^{k} f(j)$ with $l \in \mathbb{N}$ where $f(j)$ is an expression of INSH in j over $\mathbb{K}$*[93]*.*

That is to say, after the use of the quasi-shuffle algebra [12], it simplifies down to an expression of the form:

$$S(n) = \sum_{i_1=l_0}^{n} f_1(i_1) \sum_{i_2=l_1}^{i_1} f_2(i_2) \cdots \sum_{i_n=l_{n-1}}^{i_{n-1}} f_n(i_n) \tag{3.4.3}$$

where the $f_1, \ldots, f_n$ are hypergeometric products. If such an expression is a solution to a linear difference equation, it is called a ***d'Alembertian solution*** *[95].*

Remark 3.4.2. *We point out here that expressions of the form (3.4.3) include in particular the classes of harmonic, generalized harmonic, binomial and generalized binomial sums as described in Section 2.1 of Chapter 2.*

3.4.2. Uncoupling the system

We proceed now to the first alternative step[94]:

[93] Here it is assumed that the evaluation of $f(j)|_{j \mapsto m}$ for all $m \in \mathbb{Z}$ with $m \geq l$ does not introduce any poles.

[94] We will denote with a "b" the steps of this second approach.

Step 2.b: Uncouple the equation (3.3.1) using `OreSys`.

We obtain a scalar HODE in x in one of the MI, say $\tilde{\mathcal{I}}(x)$:

$$\boxed{\alpha_0(x,\epsilon)\tilde{\mathcal{I}}_1(x,\epsilon) + \cdots + \alpha_m(x,\epsilon)\mathrm{d}_x^m\tilde{\mathcal{I}}_1(x,\epsilon) = \beta(x,\epsilon)} \tag{3.4.4}$$

where the $\alpha_i(x,\epsilon) \in \mathbb{K}[x,\epsilon]$. If we find those in the rational field $\mathbb{K}(x,\epsilon)$ instead of the ring, it suffices to multiply the whole equation by $\mathrm{lcm}(\mathrm{den}\,\alpha_1,\ldots,\mathrm{den}\,\alpha_n)$[95]. Similarly to the previous case, the inhomogeneous right-hand side (RHS) $r(x,\epsilon)$ of the equation depends only on the inhomogeneous part of the system and its derivatives, and the other MIs on the first MI, its derivatives and the RHS part, i.e.,

$$\beta(x,\epsilon) = \sum_{i=0}^{p}\sum_{j=1}^{m}\beta_{i,j}(x,\epsilon)\mathrm{d}_x^i\tilde{\mathcal{R}}_j(x,\epsilon) \text{ with } p \in \mathbb{N},\ \beta_{i,j}(x,\epsilon) \in \mathbb{K}(x,\epsilon), \tag{3.4.5}$$

$$\begin{aligned}\tilde{\mathcal{I}}_j(x,\epsilon) &= \sum_{i=0}^{m-1} a_{k,i}(x,\epsilon)\mathrm{d}_x^i\tilde{\mathcal{I}}_1(x,\epsilon) + \rho_k(x,\epsilon) \text{ with } j \in \{2,\ldots,m\},\ a_{k,i}(x) \in \mathbb{K}(x,\epsilon),\\ \rho_k(x,\epsilon) &= \sum_{i=0}^{q}\sum_{j=1}^{m}\rho_{k,i,j}(x,\epsilon)\mathrm{d}_x^i\tilde{\mathcal{R}}_j(x,\epsilon) \text{ with } q \in \mathbb{N},\ \rho_{k,i,j}(x,\epsilon) \in \mathbb{K}(x,\epsilon).\end{aligned} \tag{3.4.6}$$

Remark 3.4.3. *It is actually possible at this point to introduce a small refinement [50] that might be very helpful to improve the speed of the algorithm. Let us define*

$$p(x) := \gcd(\alpha_0(x,0),\ldots,\alpha_m(x,0)).$$

Then we can divide the equation (3.4.4) by $p(x)$ and get

$$\sum_{n=0}^{m}\frac{\alpha_n(x,\epsilon)}{p(x)}\tilde{\mathcal{I}}_1(x,\epsilon) = \frac{\beta(x,\epsilon)}{p(x)}.$$

The improvement will be made clear in Remark 3.4.4 below.

3.4.3. Order by order solving

Now, we write the following ansatz for the MI $\tilde{\mathcal{I}}_1(x,\epsilon)$:

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{n=0}^{\infty} I_1(n,\epsilon)x^n, \quad x \in [0;1], \tag{3.4.7}$$

[95] $\mathrm{den}\,\alpha_i$ denotes the denominator of α_i, i.e., if $\alpha_i(x,\epsilon) = \frac{p_i(x,\epsilon)}{q_i(x,\epsilon)}$ with $(p_i,q_i) \in \mathbb{K}[x,\epsilon]\times\mathbb{K}[x,\epsilon]^\star$, then $\mathrm{den}\,\alpha_i(x,\epsilon) = q_i(x,\epsilon)$.

i.e., we assume that it is regular in x. In various applications one has more generally

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{n=0}^{\infty}\sum_{k=0}^{m} I_1^{(0)}(n,k,\epsilon)\log(x)^k x^n$$

around 0 and

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{n=0}^{\infty}\sum_{k=1}^{l} I_1^{(1)}(n,k,\epsilon)\log(1-x)^k x^n$$

around 1. Usually, the convergence radius of either of these series, r, is smaller than one.

This leads us to the next step.

Step 3.b: Plugging the relation (3.4.7) in the HODE (3.4.4), we perform coeffcient comparison with respect to powers of x and write a recurrence of the form

$$a_0(n,\epsilon)I_1(n,\epsilon) + a_1(n,\epsilon)I_1(n+1,\epsilon) + \cdots + a_d(n,\epsilon)I_1(n+d,\epsilon) = b(n,\epsilon) \quad (3.4.8)$$

where $d \in \mathbb{N}$, $a_i(n,\epsilon) \in \mathbb{K}[n,\epsilon]$, $b(n,\epsilon)$ can be expanded in a Laurent series

$$b(n,\epsilon) = \sum_{k=-l}^{\infty} b_k(n)\epsilon^k \text{ with } l \in \mathbb{N}. \quad (3.4.9)$$

In addition, we can compute the coefficients $b_{-l}(n),\ldots,b_r(n) \in \mathbb{K}(n)$, $r \in \mathbb{Z}$ for $n \in \mathbb{N}$ as expressions in terms of indefinite nested sums over hypergeometric products.

Proof. Let us show this for the homogeneous HODE[96], i.e., for $r(x,\epsilon) = 0$ and therefore $b(n,\epsilon) = 0$. For all $i \in \{1,\ldots,m\}$, since $\alpha_i(x,\epsilon) \in \mathbb{K}[x,\epsilon]$ we have

$$\alpha_i(x,\epsilon) = \sum_{r=0}^{p_i} \bar{a}_i(r,\epsilon)x^r \text{ with } p_i = \deg_x(\alpha_i) \in \mathbb{N}.$$

Therefore, using the Cauchy product:

$$\begin{aligned}\alpha_i(x,\epsilon)\mathrm{d}_x^i\tilde{\mathcal{I}}_1(x,\epsilon) &= \left(\sum_{r=0}^{p_i}\bar{a}_i(r,\epsilon)x^r\right)\left(\sum_{n=i}^{\infty}\frac{n!}{(n-i)!}I_1(n,\epsilon)x^{n-i}\right)\\ &= \left(\sum_{r=0}^{p_i}\bar{a}_i(r,\epsilon)x^r\right)\left(\sum_{n=0}^{\infty}\frac{(n+i)!}{n!}I_1(n+i,\epsilon)x^{n}\right)\end{aligned}$$

[96] For more informations on holonomic fonctions and sequences, see for instance [118].

$$= \sum_{n=0}^{\infty} C_i(n,\epsilon)x^n$$

where, since for all $n > p_i$ we have $a_i(n,\epsilon) = 0$, the Cauchy product $C_i(n,\epsilon)$ can be written as[97]

$$C_i(n,\epsilon) := \sum_{k=0}^{\min(n,p_i)} \frac{(n+i-k)!}{(n-k)!}\bar{a}_i(k,\epsilon)I_1(n+i-k,\epsilon).$$

We plug this in (3.4.4) and get:

$$\alpha_0(x,\epsilon)\tilde{\mathcal{I}}_1(x,\epsilon) + \cdots + \alpha_m(x,\epsilon)\mathrm{d}_x^m\tilde{\mathcal{I}}_1(x,\epsilon) = 0 \Rightarrow \sum_{n=0}^{\infty}\left[\sum_{i=0}^{m} C_i(n,\epsilon)\right]x^n = 0,$$

that is to say we have for $n \geq \max_{1\leq i\leq m} p_i$:

$$\boxed{\sum_{i=0}^{m}\sum_{k=0}^{p_i}\frac{(n+i-k)!}{(n-k)!}\bar{a}_i(k,\epsilon)I_1(n+i-k,\epsilon) = 0}. \tag{3.4.10}$$

We define now

$$M := \max_{1\leq i\leq m} p_i.$$

The coefficient in front of $I_1(n+j,\epsilon)$ for $j \in \{-M,\ldots,-1,0,1,\ldots,m\}$ is obtained by setting $i-k = j \Leftrightarrow i = k+j$ and collecting the relevant $\bar{a}_i(k,\epsilon)$ coefficients, i.e., by defining

$$\tilde{a}_j(n,\epsilon) := \sum_{r=\max(0,-j)}^{m-j}\frac{(n+j)!}{(n-r)!}\bar{a}_{r+j}(r,\epsilon). \tag{3.4.11}$$

We get a difference equation of the following form:

$$\tilde{a}_{-M}(n,\epsilon)I_1(n-M,\epsilon) + \cdots + a_m(n,\epsilon)I_1(n+m,\epsilon) = 0. \tag{3.4.12}$$

Shifting everything by M and defining the following quantities:

$$\boxed{a_j(n,\epsilon) := \tilde{a}_{j-M}(n,\epsilon) = \sum_{r=\max(0,M-j)}^{m+M-j}\frac{(n+j-M)!}{(n-r)!}\bar{a}_{r+j-M}(r,\epsilon)} \tag{3.4.13}$$

we finally get the expected difference equation:

$$a_0(n,\epsilon)I_1(n,\epsilon) + \cdots + a_d(n,\epsilon)I_1(n,\epsilon) = 0; \tag{3.4.14}$$

it is of order $d \leq m+M$, since some of the $a_j(n,\epsilon)$ might be equal to 0 or cancel among themselves. □

[97] This operation is defined in the ring of formal power series. If one wants to stay in the analytic world, one might need to check the absolute convergence of the $\mathrm{d}_x^i\tilde{\mathcal{I}}_1(x,\epsilon)$ series in order to perform the Cauchy product on it and $\alpha_i(x,\epsilon)$.

Remark 3.4.4. *With the refinement introduced above, consisting in dividing* (3.4.4) *by the GCD of the coefficients, there is actually an even stronger bound on* d*, that is to say* $d \leq m + M - \deg p$.

Remark 3.4.5. *We note that this improvement has a direct influence on solving the systems in closed form*[98] *or applying the large moment method; see Section 3.4.6.*

At this point, we obtained a difference equation of higher order in n that still depends on ϵ. In order to solve it, we need to get rid of the latter.

Step 4.b: We solve the difference equation (3.4.14) order by order in ϵ, i.e. we search for a Laurent expansion for $I_1(n,\epsilon)$ of the form

$$I_1(n,\epsilon) = \sum_{k=-l}^{\infty} I_{1,k}(n)\epsilon^k \tag{3.4.15}$$

where $l \in \mathbb{N}$ and the $I_{1,k}(n)$ can be given in terms of indefinite nested sums over hypergeometric products.

In order to solve the difference equation (3.4.14) recursively, we suppose first that all the $a_i(x,0) \in \mathbb{K}[x,\epsilon]$ are not zero at the same time, which can always be achieved. Indeed, if this is the case, we can write them all in the form:

$$a_i(x,\epsilon) = \epsilon^{u_i}\bar{a}_i(x,\epsilon) \text{ where } u_i \in \mathbb{N},\ \bar{a}_i(x,\epsilon) \in \mathbb{K}[x,\epsilon],\ \bar{a}_i(x,0) \neq 0.$$

Then by setting $u := \operatorname{lcm}(u_1,\ldots,u_d)$ we divide (3.4.14) by ϵ^u so that the equation becomes:

$$\bar{a}_0(n,\epsilon)I_1(n,\epsilon) + \cdots + \bar{a}_d(n,\epsilon)I_d(n,\epsilon) = \bar{b}(n,\epsilon)$$

where the lowest pole of $b(n,\epsilon)$ around $\epsilon = 0$ is shifted by u:

$$\bar{b}(n,\epsilon) := \sum_{k=-l}^{\infty} b_k(n)\epsilon^{k-u} = \sum_{k=-(l+u)}^{\infty} b_{k+u}(n)\epsilon^k.$$

Now, we can define a maximal $d' \in \mathbb{N}$, $d' < d$ such that $a_{d'}(x,0) \neq 0$ as well as a $M \in \mathbb{N}$ such that for all $n \geq M$, $a_{d'}(n,0) \neq 0$. The idea is to algorithmically construct when possible an expansion of the form (3.4.15) using `Sigma`[99], with given initial values $I_{1,j}(i) = c_{i,j}$ for $j \in \{l,\ldots,r\}$ and $i \in \{0,\ldots,\max(d',M)-1\}$. The general steps of the method are as follows:

[98] See Section 3.3.4

[99] See for more detail [119].

1. We plug the expansion (3.4.15) in (3.4.14) :

$$\begin{aligned}
&a_0(n,\epsilon)\left[I_{1,-l}(n)\epsilon^{-l} + I_{-l+1,1}(n)\epsilon^{-l+1} + \cdots\right]\\
+&a_1(n,\epsilon)\left[I_{1,-l}(n+1)\epsilon^{-l} + I_{-l+1,1}(n+1)\epsilon^{-l+1} + \cdots\right]\\
+&\cdots\\
+&a_{d'}(n,\epsilon)\left[I_{1,-l}(n+d)\epsilon^{-l} + I_{-l+1,1}(n+d')\epsilon^{-l+1} + \cdots\right]\\
&= b_{-l}(n)\epsilon^{-l} + b_{-l+1}(n)\epsilon^{-l+1}
\end{aligned}$$

   and then collect terms:

$$\begin{aligned}
&\left[a_0(n,\epsilon)I_{1,-l}(n) + \cdots + a_{d'}(n,\epsilon)I_{1,-l}(n+d') - b_l(n)\right]\epsilon^{-l}\\
+&\left[a_0(n,\epsilon)I_{1,-l+1}(n) + \cdots + a_{d'}(n,\epsilon)I_{1,-l+1}(n+d') - b_{-l+1}(n)\right]\epsilon^{-l+1}\\
+&\cdots = 0.
\end{aligned}$$

2. We multiply everything by ϵ^l and then send $\epsilon \to 0$. Since $a_i(n,0) \neq 0$, the only term that remains is therefore

$$a_0(n,0)I_{1,-l}(n) + \cdots + a_{d'}(n,0)I_{-l,1}(n+d') = b_l(n).$$

3. Now we use `Sigma` to solve[100] the obtained recurrence equation and to combine the solutions such that the initial conditions are fulfilled. In the case that `SolveRecurrence` fails to find a solution, then we stop. Otherwise, we obtain an expression of $I_{1,-l}(n)$ in terms of INSH.

4. We define then

$$I_1'(n,\epsilon) := I_1(n,\epsilon) - I_{1,-l}(n)\epsilon^{-l} = \sum_{k=-l+1}^{\infty} I_{1,k}(n)\epsilon^k \tag{3.4.16}$$

   and plug this into (3.4.14):

$$a_0(n,\epsilon)\left[I_{1,l}(n)\epsilon^l + I_1'(n,\epsilon)\right] + \cdots + a_{d'}(n,\epsilon)\left[I_{1,l}(n+d')\epsilon^l + I_1'(n+d')\right] = b_l(n,\epsilon).$$

5. Collecting the ϵ^l terms and getting them on the other side, we get an equation of a form similar to (3.4.14) for $I_1'(n,\epsilon)$ with an updated RHS:

$$a_0(n,\epsilon)I_1'(n,\epsilon) + \cdots + a_{d'}(n,\epsilon)I_1'(n+d',\epsilon) = b'(n,\epsilon) \tag{3.4.17}$$

[100] The underlying algorithm can be considered as the discrete version described in Section 3.3.3. In particular, we have to assume that the recurrence factorizes into first-order factors (see [95]).

where

$$b'(n,\epsilon) := b(n,\epsilon) - [a_0(n,\epsilon)I_{1,-l}(n) + \cdots + a_{d'}(n,\epsilon)I_{1,-l}(n+d',\epsilon)]\,\epsilon^{-l}.$$

Note that the coefficients of the pole of order l in $b'(n,\epsilon)$ cancel among themselves and $b'(n,\epsilon)$ has only a pole of order $l-1$ around $\epsilon = 0$.

6. We loop this procedure to get the next term $I_{1,-l+1}(n)$.

3.4.4. Solving for the other master integrals and iterating

Step 5.b: Once we get a solution for $\tilde{\mathcal{I}}_1(x,\epsilon)$ up to some order l' in ϵ, we plug it in (3.4.6) in order to get the other MIs up to the required order in ϵ.

Remark 3.4.6. *It might well be that due to the structure of the equations (3.4.6), there are some simplifications or some coefficients $a_{k,i}(x,\epsilon)$ that have poles in ϵ. What is actually necessary to do is some additional preprocessing in order to find the order $k \in \mathbb{N}$ of the maximal pole in each of those expressions for the $\tilde{\mathcal{I}}_j(x,\epsilon)$ and compute $\tilde{\mathcal{I}}_1(x,\epsilon)$ up to the required order+k.*

3.4.5. Alternative application of the method

The second method presented above relies, compared to the first one, on the setting of recurrences and `Sigma`, whereas the first one was dealing with differential equations through extensive use of the `SolveDE` function of the `HarmonicSums` package. It is actually possible to remain in this setting also for the second method, and this consists in taking the following twist[101]: after the Step 2.b, i.e., the uncoupling through `OreSys` of the whole subsystem, instead of expanding $\tilde{\mathcal{I}}_1(x,\epsilon)$ in x, we expand in ϵ, i.e.,

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{n=0}^{\infty} I_1(n,\epsilon)x^n \to \tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{k=-l}^{\infty} \tilde{\mathcal{I}}_{1,k}(x)\epsilon^k. \tag{3.4.18}$$

Remark 3.4.7. *Actually, to be once again perfectly rigourous and if one does not want to stay within the formal Laurent setting, one might need to use some discrete version of the Fubini theorem to check whether this alternative expansion is equivalent to the first one, i.e., if one is allowed to interchange freely the ϵ and x sums:*

$$\tilde{\mathcal{I}}_1(x,\epsilon) = \sum_{n=0}^{\infty}\sum_{k=-l}^{\infty} I_1(n)\epsilon^k x^n = \sum_{k=-l}^{\infty}\sum_{n=0}^{\infty} I_1(n)\epsilon^k x^n.$$

[101] Compare also [89].

Now, expanding the $\alpha_i(x,\epsilon)$ and $\beta(x,\epsilon)$ of (3.4.4) in ϵ instead of x, we can proceed exactly in the spirit of the ideas of the proof for the step 3.b as well as those of step 4.b: plug all the ϵ expansions in the HODE (3.4.4), identify terms with same order of ϵ and then send ϵ to 0 to get a HODE purely in x of the following form for the first coefficient $\tilde{\mathcal{I}}_1(x,\epsilon)$:

$$a_0(x,0)\tilde{\mathcal{I}}_1^{(-l)}(x) + \cdots + a_d(x,0)\mathrm{d}_x^d\tilde{\mathcal{I}}_1^{(-l)}(x) = \beta_{-l}(x), \tag{3.4.19}$$

where $d \leq m$ and the $a_i(x,0) \in \mathbb{K}[x]$ are obtained from linear combinations of the ϵ-expansion coefficients of the $\alpha_i(x,\epsilon)$, in the spirit of (3.4.13).
Here we can apply directly `SolveDE` in order to get the homogeneous and particular solutions of the HODE, given as always that it is first-order factorizable, and using the initial conditions, we solve completely for $\tilde{\mathcal{I}}_{1,-l}(x)$. The rest of the computation proceeds in the same way [50], i.e., plug the solution in the initial HODE (3.4.4) and repeat until the required order is reached. Finally, we insert the obtained solution in the uncoupling equation relating the $\tilde{\mathcal{I}}_j(x,\epsilon)$ to $\tilde{\mathcal{I}}_1(x,\epsilon)$.

Similarly, one may proceed differently after the uncoupling given in (3.3.5). Namely one may transform the linear differential equations to a recurrence as elaborated in Section 3.4.3 (but being free of ϵ) and solve the obtained recurrence by `Sigma`. This tactic in connection with the large moment method in Section 3.4.6 has been considered in [62].

3.4.6. The large moments method

If one fails to solve the linear recurrences arising in (3.4.3), or alternatively the attached differential equation in (3.4.4), one may focus on computing the values $I_{i,k}(n)$ in

$$I_I(n,\epsilon) = \sum_{k=-l}^{\infty} I_{i,k}(n)\epsilon^k \tag{3.4.20}$$

for $n = 0, 1, \ldots, \lambda$ with large λ ($\lambda \sim 10000$) by using the derived recurrences. This method, introduced in [61] have been recently applied to many physical problems such as [28, 63, 64, 66–68, 120–127].

4. Optimising the analysis of differential systems

As introduced in Chapter 3, we consider a differential system of first order in x, called $\mathcal{S}$, in $n \in \mathbb{N}$ master integrals (MIs from now on) that we call $\mathcal{I}_1, \ldots, \mathcal{I}_n$ and $m \in \mathbb{N}$ base case integrals (BIs from now on) in the homogeneous part, i.e., in the right hand side (RHS from now on) $\mathcal{J}_1, \ldots, \mathcal{J}_m$. All integrals are functions of the differentiation variable x as well as the spacetime dimension D that we parametrize according to usual physics notations as

$$D := 4 + \epsilon \tag{4.0.1}$$

or equivalently, we express the integrals as a function of ϵ. We can therefore write the system $\mathcal{S}$ in a compact form as

$$\frac{\mathrm{d}}{\mathrm{d}x}\mathcal{I}(x,\epsilon) = \mathcal{M}(x,\epsilon) \cdot \mathcal{I}(x,\epsilon) + \mathcal{R}(x,\epsilon) \tag{4.0.2}$$

where the different quantites are:

- the MI vector (MIV)
$$\mathcal{I}(x,\epsilon) = (\mathcal{I}_1(x,\epsilon), \ldots, \mathcal{I}_n(x,\epsilon))^\intercal ;$$
- the RHS vector (RHSV)
$$\mathcal{R}(x,\epsilon) = (\mathcal{R}_1(x,\epsilon), \ldots, \mathcal{R}_n(x,\epsilon))^\intercal ;$$
- the coefficient matrix (CM)
$$\mathcal{M}(x,\epsilon) \in \mathcal{M}_{n\times n}(\mathbb{K}(x,\epsilon))$$
with $\mathbb{K}$ a computable field as usual (frequently $\mathbb{K} \subseteq \mathbb{R}$).

The RHSV, corresponding to the inhomogeneous part of the system, is defined more precisely as

$$\forall i \in \{1, \ldots, n\},\ \mathcal{R}_i(x,\epsilon) = \sum_{k=1}^{n} \varphi_{i,k}(x,\epsilon)\mathcal{J}_k(x,\epsilon), \quad \varphi_{i,k}(x,\epsilon) \in \mathbb{K}(x,\epsilon) \tag{4.0.3}$$

More compactly, denoting with $\langle a_1, \ldots, a_n\rangle_{\mathbb{K}}$ the space generated linearly by the elements $a_1, \ldots, a_n$ over some field $\mathbb{K}$, we have that

$$\mathcal{R}(x,\epsilon) \in \left(\langle \mathcal{J}_1(x,\epsilon), \ldots, \mathcal{J}_m(x,\epsilon)\rangle_{\mathbb{K}(x,\epsilon)}\right)^n$$

Remark 4.0.1. *The solving algorithms of Section 3.4 first uncouple the system and afterwards look for solutions for the coefficients of the ϵ-expansion. Throughout this chapter, this is the approach we concentrate on, because our optimisation method intrinsically relies on the ϵ-order analysis in the differential equations and relations obtained as a resulf of uncoupling.*

Aim: Find among the possible uncoupling strategies the best version so that orders to which the integrals under consideration have to be expanded are minimal.

We note that the available package `SolveCoupledSystems` [72] by Schneider had refined some analysis in this direction that were essential for concrete calculations of Feynman integrals in closed form or to evaluate a large number of moments of them. In this chapter we will enhance the analysis to be ready for even more challenging calculations.

4.1. Preprocessing the system

Because we consider input systems that are quite big, usually 100 — 500 equations, the first step is to preprocess the system, more precisely by **triangularizing** or **clustering** it. The idea is to find a list of subsystems of $\mathcal{S}$ that can be solved in a recursive way from the simplest one onwards[102]. We call this list of systems the set of irreducible systems defined as follows:

Definition 4.1.1 (Irreducible systems)**.** *Let $\mathcal{S}$ defined as (4.0.2). The **set of irreducible systems** of $\mathcal{S}$ is a list $\{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$ such that the following holds:*

- *$\{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$ is a set partition of $\mathcal{S}$.*
- *For $1 \leq i \leq \ell$, $\mathcal{S}_i$ is a system whose RHS depends on the integrals of*

$$\mathcal{S}_1 \cup \cdots \cup \mathcal{S}_{i-1}$$

and during all these choices, $|\mathcal{S}_i|$ is minimal for $1 \leq i \leq \ell$.

We write in short:

$$\mathcal{S}_1 \prec \mathcal{S}_2 \prec \cdots \prec \mathcal{S}_n \prec \mathcal{S} \tag{4.1.1}$$

*where $\mathcal{S}_i \prec \mathcal{S}_j$ means that $\mathcal{S}_i$ is a "smaller" subsystem than $\mathcal{S}_j$ in the sense above. The $\mathcal{S}_i$ are called **irreducible systems**. In particular $\mathcal{S}$ is called an irreducible system if $\{\mathcal{S}\}$ is an irreducible subsystem of $\mathcal{S}$.*

[102] Note that a preculstering is available if IBP methods are used to derive those systems.

The algorithm that produces a set of irreducible systems $\{\mathcal{S}_1, \ldots, \mathcal{S}_\ell\}$ is pretty straightforward. By construction, one can also prove the unicity of such as decomposition,. The algorithm goes as follows:

Algorithm for computing the set of irreducible systems:

1. For each of the equations of the system $\mathcal{S}$, define a rule that associates to the differentiated MI the set of those MIs on which it explicitely depends. More precisely, given the equation for the MIS $\mathcal{I}_k$, $k \in \{1, \ldots, n\}$ of the system $\mathcal{S}$, that has the general form

$$\frac{\mathrm{d}}{\mathrm{d}x}\mathcal{I}_k = \sum_{p=1}^{q} f_p(x,\epsilon)\mathcal{I}_{k_p}(x,\epsilon) + \mathcal{R}_k(x,\epsilon) \quad q \leq n,$$

$$\forall i, j \in \{1, \ldots, q\},\ f_i(x,\epsilon) \in (\mathbb{K}(x,\epsilon))^\star,\ k_i \in \{1, \ldots, n\},\ k_i \neq k_j,$$

   the rule would be[a]:

$$\mathcal{I}_k \longrightarrow \{\mathcal{I}_{k_1}, \ldots, \mathcal{I}_{k_q}\}$$

2. We apply those rules recursively on themselves to find the full dependence of the MIs on other MIs[b]. Given the algorithm has been tailored to match the Wolfram language syntax, by "applying rules" we mean here the following: given a collection of objects $A, A_1, \ldots, A_i, \ldots, A_n, B_1, \ldots, B_m$, $A \neq A_i$, $A \neq i$ for all i, j, and two rules $\mathbf{R_1} : A \longrightarrow \{A_1, \ldots, A_i, \ldots, A_n\}$ and $\mathbf{R_2} : A_i \longrightarrow \{B_1, \ldots, B_m\}$, applying $\mathbf{R_1}$ and/or $\mathbf{R_2}$ to A means here mapping:

$$(\mathbf{R_1})(A) = \{A_1, \ldots, A_i, \ldots, A_n\}, \quad (\mathbf{R_2})(A) = A$$

$$(\mathbf{R_1}, \mathbf{R_2})(A) = \{A_1, \ldots, A_i, \ldots, A_n\} \cup \{B_1, \ldots, B_m\}$$

3. It suffices now to collect the MIs that have the same dependence with respect to the full rules: the set of equations in the MIs with the same full dependence will be an irreducible subsystem of $\mathcal{S}$.

[a] Given that we request all rational functions $f_i(x,\epsilon)$ to be non-zero, there is an explicit dependence of $\mathcal{I}_k$ on all of the $\mathcal{I}_{k_i}$, $i \in \{1, \ldots, q\}$

[b] Note that often $\mathbb{R}_\mathbb{k} = 0$. Otherwise, the arising BIs are exactly those that do not arise on the left-hand side.

Remark 4.1.1. *Each time we apply the rules in Step 2, we either add at least one new MI, or we do not add anything which means that we exhausted all of them and can stop. Therefore the algorithm finshes in finite time, at most after n steps.*

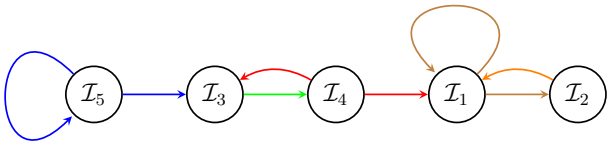
Example 4.1.1. *In order to see how the algorithm works let us take the following toy-system:*

$$\begin{cases} \frac{1}{x^2-1}\mathcal{I}_5'(x) &= \frac{x(x+2)}{x-2}\mathcal{I}_5(x) + x\mathcal{I}_3(x) + \mathcal{R}_5(x) \\ (x+1)\mathcal{I}_4'(x) &= -\mathcal{I}_3(x) + \frac{\mathcal{I}_1(x)}{x} + \mathcal{R}_4(x) \\ \frac{x(x-2)}{3}\mathcal{I}_3'(x) &= x\mathcal{I}_4(x) + \mathcal{R}_3(x) \\ \mathcal{I}_2'(x) &= x^3\mathcal{I}_1(x) + \mathcal{R}_2(x) \\ 2\mathcal{I}_1'(x) &= x\mathcal{I}_1(x) - \frac{8}{x-1}I_2(x) + \mathcal{R}_1(x), \end{cases}$$

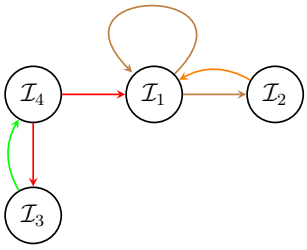
where the RHS is not specified further. As in the first step, we write all the corresponding rules:

$$\mathcal{I}_5 \longrightarrow \{\mathcal{I}_3, \mathcal{I}_5\},\ \mathcal{I}_4 \longrightarrow \{\mathcal{I}_1, \mathcal{I}_3\},\ \mathcal{I}_3 \longrightarrow \{\mathcal{I}_4\},\ \mathcal{I}_2 \longrightarrow \{\mathcal{I}_1\},\ \mathcal{I}_1 \longrightarrow \{\mathcal{I}_1, \mathcal{I}_2\}. \quad (4.1.2)$$

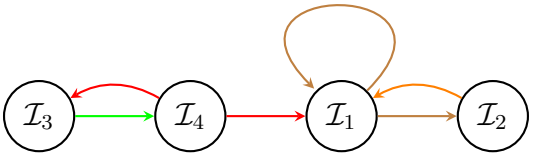
As in the second step, we now apply the rules recursively, to get the full dependence rule. This process can be represented diagramatically as:



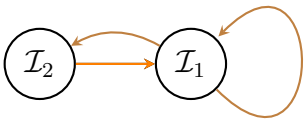
$$\mathcal{I}_5 \to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5\}$$



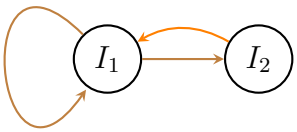
$$\mathcal{I}_4 \to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}$$



$$\mathcal{I}_3 \to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}$$



$$\mathcal{I}_2 \to \{\mathcal{I}_1, \mathcal{I}_2\}$$



$$\mathcal{I}_1 \to \{\mathcal{I}_1, \mathcal{I}_2\}$$

The full rules are therefore the following:

$$\boxed{\mathcal{I}_5 \to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5\}}$$

$$\boxed{\begin{aligned}\mathcal{I}_4 &\to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}\\ \mathcal{I}_3 &\to \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}\end{aligned}}$$

$$\boxed{\begin{aligned}\mathcal{I}_2 &\to \{\mathcal{I}_1, \mathcal{I}_2\}\\ \mathcal{I}_1 &\to \{\mathcal{I}_1, \mathcal{I}_2\}\end{aligned}}$$

So now it suffices to cluster (hence the name) the equations by identical full dependence — we realise that we get three irreducible systems:

$$\mathcal{S}_1 := \{\mathcal{I}_1, \mathcal{I}_2\},\ \mathcal{S}_2 := \{\mathcal{I}_3, \mathcal{I}_4\},\ \mathcal{S}_3 := \{\mathcal{I}_5\}.$$

By construction of the rules, we also get a natural order in which we need to solve the systems: first $\mathcal{S}_1$ as it is homogeneous, then $\mathcal{S}_2$ as we will need the expressions for $\mathcal{I}_1, \mathcal{I}_2$ in the homogeneous part and finally $\mathcal{S}_3$ which is actually can be considered as just one unique differential equation in $\mathcal{I}_5$ with all the other MIs being in the inhomogeneous term.

Remark 4.1.2. *During the clustering process, we might have equations that are of the form*

$$\frac{\mathrm{d}}{\mathrm{d}x}\mathcal{I}_j = 0.$$

*This actually means that $\mathcal{I}_j$ depends only on ϵ, i.e., $\mathcal{I}_j \to \{\}$. More precisely, $\mathcal{I}_j$ can be considered in this case as a BIs: we call it **induced BI**, or IBI in short. When detected by our algorithm, it is automatically added to the BIs set.*

4.2. How to proceed

First we will focus on one irreducible system (see Sections 4.3 and 4.4). Afterwards we will utilise the recursive nature of the full system and combine the subresults of the irreducible subsystems to obtain a full analysis of the combined system in Section 4.5.

The analysis of an irreducible subsystem is split further into two steps. First, we focus on optimising the ϵ-order expansion, looking at the BI integrals only in Section 4.3. Finally, we will take also into account the required ϵ-order expansion of the MIs to be solved in the given irreducible systems in Section 4.4. Combining both aspects will lead to the required information for the irreducible subsystem itself.

4.3. Selecting the equations to solve for an irreducible system

4.3.1. Criterion for optimal uncoupling

Once the preprocessing clustering step is done, we can concentrate on individual irreducible subsystems, as a simpler subtask. From now on, we note all the quantities of the subsystems with a tilde (e.g., $\tilde{\mathcal{I}}$). Because these subsystems can be solved completely up to a certain order in ϵ given the BIs and the MIs also up to a certain order in ϵ — we will make that clear in an instant — we can analyse them individually. Naturally, the first question that comes to our mind is about the meaning of "one uncoupling is 'better' than another". In other words, we should find a comparison point between different uncouplings.

Let us consider a toy example to motivate this challenge. Suppose we have a system in three MIs $\tilde{\mathcal{I}}_{i=1,2,3}(x,\epsilon)$ with one BI $\tilde{\mathcal{J}}(x,\epsilon)$, that we uncoupled in $\tilde{\mathcal{I}}_1$ so that we get the following HODE for $\tilde{\mathcal{I}}_1$ and the following expression for $\tilde{\mathcal{I}}_2$:

$$\tilde{\mathcal{I}}_2(x,\epsilon) = F(\{\tilde{\mathcal{I}}_1(x,\epsilon), \frac{\mathrm{d}^i}{\mathrm{d}x^i}\tilde{\mathcal{I}}_1(x,\epsilon)\}) + \frac{1}{\epsilon^2}\tilde{\rho}_{2,1}(x,\epsilon)\tilde{\mathcal{J}}(x,\epsilon) + \epsilon\tilde{\rho}_{2,2}(x,\epsilon)\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}(x,\epsilon)$$

where $F(\{\tilde{\mathcal{I}}_1, \frac{\mathrm{d}^i}{\mathrm{d}x^i}\tilde{\mathcal{I}}_1\})$ is a linear expression over $\tilde{\mathcal{I}}_1$ and its derivatives $\frac{\mathrm{d}^i}{\mathrm{d}x^i}\tilde{\mathcal{I}}_1$ over

$\mathbb{K}(x,\epsilon)$ and

$$\tilde{\rho}_{2,i\in\{1,2\}}(x,\epsilon)\in\mathbb{K}(x,\epsilon)\text{ with }\epsilon\nmid\operatorname{den}\left(\tilde{\rho}_{2,i\in\{1,2\}}(x,\epsilon)\right)$$

or, in simple terms, the $\tilde{\rho}_{2,i}(x,\epsilon)$ being rational functions being regular at $\epsilon=0$. Now, suppose we want for instance to solve the system up to the 1st order in ϵ. This means we want to get all the $\tilde{\mathcal{I}}_i$ up to order 1. Then we have to find out up to which order $n\geq l$ we need, or more precisely *it is sufficient*, to give an expansion of the BI $\tilde{\mathcal{J}}(x,\epsilon)=\sum_{n=l}^{\infty}\tilde{\mathcal{J}}_n(x)\epsilon^n$. Let us put aside for the moment the contribution from $F(\{\tilde{\mathcal{I}}_1,\frac{\mathrm{d}^i}{\mathrm{d}x^i}\tilde{\mathcal{I}}_1\})$[103]. We have two terms involving $\tilde{\mathcal{J}}(x,\epsilon)$:

- $\frac{1}{\epsilon^2}\tilde{\rho}_{2,1}(x,\epsilon)\tilde{\mathcal{J}}(x,\epsilon)$ — for this term, around $\epsilon\to 0$ we get

$$\begin{aligned}\frac{1}{\epsilon^2}\tilde{\rho}_{2,1}(x,\epsilon)\tilde{\mathcal{J}}(x,\epsilon)&\sim\frac{1}{\epsilon^2}\tilde{\mathcal{J}}(x,\epsilon)\\&=\frac{1}{\epsilon^2}\left(\cdots+\tilde{\mathcal{J}}_0(x)+\epsilon\tilde{\mathcal{J}}_1(x)+\epsilon^2\tilde{\mathcal{J}}_2(x)+\epsilon^3\tilde{\mathcal{J}}_3(x)+\cdots\right)\\&=\cdots+\frac{1}{\epsilon^2}\tilde{\mathcal{J}}_0(x)+\frac{1}{\epsilon}\tilde{\mathcal{J}}_1(x)+\tilde{\mathcal{J}}_2(x)+\boxed{\epsilon\tilde{\mathcal{J}}_3(x)}+\cdots\end{aligned}$$

- $\epsilon\tilde{\rho}_{2,2}(x,\epsilon)\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}(x,\epsilon)$ — for this term we get at worst[104]

$$\begin{aligned}\epsilon\tilde{\rho}_{2,2}(x,\epsilon)\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}(x,\epsilon)&\sim\epsilon\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}(x,\epsilon)\\&=\epsilon\left(\cdots+\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_0(x)+\epsilon\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_1(x)+\epsilon^2\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_2(x)+\cdots\right)\\&=\cdots+\boxed{\epsilon\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_0(x)}+\epsilon^2\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_1(x)+\epsilon^3\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_2(x)+\cdots\end{aligned}$$

In this case we see that to satisfy both constraints, we have to determine $\tilde{\mathcal{J}}(x,\epsilon)$ up to order 3. In other words, to compare two uncouplings, we compare the $\epsilon-$orders for the BIs appearing in the inhomogeneous part of the HODE and in the linear factors (LFs): **the less the expansion orders are, the better is the uncoupling**.

Remark 4.3.1. *Let us note that we can actually distinguish between "absolute" and "relative" ϵ-orders, so we should always be precise which one is being chosen. More precisely, the "absolute" order is the usual* ***order up to which we should expand a particular function****. For example, in the example above we have*

[103] We will come back to it in the full algorithm below in Section 4.4.

[104] Because we can get $\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{J}}_i(x)=0$ for the first $i=l,l+1,\ldots$, and in this case the ϵ order is further reduced.

found that we need to expand the $\tilde{\mathcal{J}}(x,\epsilon)$ up to (absolute) order 3 in ϵ around $\epsilon = 0$, under the assumption that the final result should be expanded up to the linear term. It will also be useful to consider the ***shift induced by the presence of ϵ monomials in from of BIs or MIs****, which we call the "relative" order. In our example, the relative order is obtained simply by looking at the orders p_i of the ϵ coefficients in front of $\tilde{\mathcal{J}}(x,\epsilon)$ and its derivatives, and then taking $-\min\{p_i\}$, here $-\min\{1,-2\} = 2$. The advantage of the "absolute" order is that it is immediately the result but it needs more external data to be computed (in particular, up to which order we need to solve the system) and also gives a unique result taylored for a specific input data of orders, while the relative order only gives the relative "shift" in orders. Nevertheless, it is more useful because by keeping the initial input as symbolic variables and keeping track of the differences, it is possible to return at the end a (relatively simple) function that allows one almost immediately to obtain optimal solutions for any initial orders*[105]*. Another technical advantage of the relative order is due to the fact that we might not initially know where the expansion starts for the BIs, because we do not consider Taylor series expansions but rather Laurent series expansions. In particular, it is quite frequent for the expansions coming from particle physics that the expansion starts at order -3 and not 0. In this case it is better to keep everything relative until the very end, as otherwise we have to properly take care of situations where we get negative orders.*

4.3.2. Uncoupling the system and selecting the best equations — a naive approach

As motivated above, we stick to the scheme where we keep the full ϵ and x dependence when we uncouple the system.

Remark 4.3.2. *We focus for the moment on the Zürcher uncoupling [112, 114] scheme using Stefan Gerhold's* `OreSys` *Mathematica package [116].*

Let us examine first the simplest approach, which consists simply in uncoupling the system in the first variable $\tilde{\mathcal{I}}_1$. The default output will be a HODE in the first MI and $n-1$ relations to obtain the other MIs from the first. more precisely, the default algorithm introduces n dummy variables W_i and returns a list of the form, where we use the notation $\{\mathrm{d}_x^j A_i\} = \{\mathrm{d}_x A_i, \mathrm{d}_x^2 A_i, \mathrm{d}_x^3 A_i, \ldots\}$:

Zürcher output:

- $\boxed{1}$: $\mathrm{HODE}\left(\underline{W_1, \{\mathrm{d}_x^j W_1\}}; \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}\right)$
  This is a differential equation in the variable W_1, where the inhomogeneous

[105] We will expand on this more in Section 4.4.

right-hand side will depend linearly on the $\tilde{\mathcal{R}}_i$ and their derivatives. This is the fundamental equation that we always need to solve. The important point here is that the `OreSys` uncoupling function introduces artifical dummy variables $W_1, \ldots, W_n$, such that the $\tilde{\mathcal{I}}_i$ variables are obtained as linear combinations of those given in the literature of [12].

- $\boxed{2}$: $\left\{ \begin{array}{rcl} W_2 & = & F_2(W_1, \{\mathrm{d}_x^j W_1\}; \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}) \\ W_3 & = & F_3(W_1, \{\mathrm{d}_x^j W_1\}; W_2, \{\mathrm{d}_x^j W_2\}; \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}) \\ \vdots & & \vdots \\ W_n & = & F_n(W_1, \{\mathrm{d}_x^j W_1\}; \cdots; W_{n-1}, \{\mathrm{d}_x^j W_{n-1}\}; \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}) \end{array} \right.$

  From the equation $\boxed{1}$ above, once we obtain W_1 we can immediately obtain all the remaining $W_2, \ldots, W_n$ as linear combinations $F_{p=2,\ldots,n}$ of W_1, its derivatives, as well as $\tilde{\mathcal{R}}_i$ and their derivatives

- $\boxed{3}$: $\left\{ \begin{array}{rcl} \tilde{\mathcal{I}}_1 & = & G_1(\{W_i, \mathrm{d}_x^j W_i\}) \\ \vdots & & \vdots \\ \tilde{\mathcal{I}}_n & = & G_n(\{W_i, \mathrm{d}_x^j W_i\}) \end{array} \right.$

  Finally, the last set of relations allows us to recover the original variables $\tilde{\mathcal{I}}$ knowing the W_i.

Remark 4.3.3. *In those cases when $\tilde{\mathcal{I}} = W_1$, which appeared to be the most frequent during testing, it is possible to perform a direct elimination of the W_i variables in $\boxed{1}$ and $\boxed{2}$. This is what we will always be considering, and we motivate further this choice at the end of Section 4.3.4 in the Remark 4.3.7.*

Remark 4.3.4. *It is important to note complexity wise that even in the case where $\tilde{\mathcal{I}}_1 = W_1$, eliminating the W_i variables from the uncoupling is rather costly in big systems, and could in principle be time consuming.*

Definition 4.3.1. *We will denote the full set of relations for a Zürcher uncoupling in $\tilde{\mathcal{I}}_i$ as*

$$\mathcal{U}_Z^{(i,n)} := \left(\tilde{\mathcal{I}}_i^{(H,n)}, \tilde{\mathcal{I}}_1^{(i,n)}, \tilde{\mathcal{I}}_2^{(i,n)}, \ldots, \tilde{\mathcal{I}}_{i-1}^{(i,n)}, \tilde{\mathcal{I}}_{i+1}^{(i,n)}, \ldots, \tilde{\mathcal{I}}_n^{(i,n)}\right) \tag{4.3.1}$$

In $\mathcal{U}_Z^{(i,n)}$, the Z stands for Zürcher, n for the order n of the HODE and the index i indicating in which variable we are uncoupling. A HODE of order n in $\tilde{\mathcal{I}}_i$ is denoted with $\tilde{\mathcal{I}}_i^{(H,n)}$ and a LF for $\tilde{\mathcal{I}}_j$ depending on the HODE of order n in $\tilde{\mathcal{I}}_i$ with $\tilde{\mathcal{I}}_j^{(i,n)}$. More precisely, we denote:

$$\mathcal{I}_p^{(H,n)} : \mathit{HODE}\left(\underline{\tilde{\mathcal{I}}_p, \{\mathrm{d}_x^j \tilde{\mathcal{I}}_p\}}; \{\tilde{\mathcal{J}}_i, \mathrm{d}_x^j \tilde{\mathcal{J}}_i\}\right), \tag{4.3.2}$$

$$\forall q \in \{2, \ldots, n\}, \quad \tilde{\mathcal{I}}_q^{(p,n)} : \tilde{\mathcal{I}}_q = F_q(\tilde{\mathcal{I}}_p, \{\mathrm{d}_x^j \tilde{\mathcal{I}}_p\}; \{\tilde{\mathcal{J}}_i, \mathrm{d}_x^j \tilde{\mathcal{J}}_i\}) \tag{4.3.3}$$

where the $\mathcal{J}_i$ are the BIs.

Using Zürcher's method straightforwardly minimises the number of HODEs, as there will be only one per uncoupling in the Zürcher scheme, but might give less optimal orders for the ϵ-expansions, at the very least because we actually do not even have an actual choice, but only what the Zürcher uncoupling algorithm returns us for the full uncoupling of the system in $\tilde{\mathcal{I}}$. It is pretty clear that to remedy to this problem, we need to relax the constraint on the minimal number of HODEs to get better ϵ-orders. The simplest idea to achieve that consists in doing several uncouplings, in particular in each of the $\tilde{\mathcal{I}}_i$ variables, i.e., using the notation from Remark 4.3.1. We want to compute:

$$\left(\mathcal{U}_Z^{(1,n)}, \mathcal{U}_Z^{(2,n)}, \ldots, \mathcal{U}_Z^{(n,n)}\right).$$

The price to pay is that one needs initial values also for other MIs to combine solutions during the system solving. Let us first sketch out the general procedure and then expand on the steps:

1. Uncouple the irreducible system in all the possible ways by circular permutation of the variables $\tilde{\mathcal{I}}_i$.

2. We select the best HODEs according to the ϵ-order of the factors appearing in front of the BIs in it.

3. We select the best possible equations in a recursive way, given the HODEs and the LFs that we have already selected. It might happen that several "equally good" choices are possible at some point. In this case, we simply take all of them and create an equal number of branches.

4. We select the best choices among all of the possibilities at the end, and return them.

We consider an explicit example before properly formalising the general steps we have just described above:

Example 4.3.1. *Suppose we are given a system of size $n = 3$, with MIs $\tilde{\mathcal{I}}_{i=1,2,3}(x, \epsilon)$ and 4 BIs $\tilde{\mathcal{J}}_{j=1,\ldots,4}(x, \epsilon)$. From now on, a tuple $(\tilde{\mathcal{J}}_i, m_i)$ means that in the HODE/one of the LFs we encounter a factor of the form*

$$\epsilon^{m_i} \frac{\alpha_i(x,\epsilon)}{\beta_i(x,\epsilon)} \tilde{\mathcal{J}}_i(x,\epsilon) \text{ with } \alpha_i(x,\epsilon), \beta_i(x,\epsilon) \in \mathbb{K}[x,\epsilon] \text{ and } \epsilon \nmid \alpha_i(x,\epsilon), \beta_i(x,\epsilon)$$

Furthermore, we consider the relative order approach, as introduced in Remark 4.3.1.

Suppose we have managed already to uncouple the system cyclically[106]*, i.e., to perform three uncouplings in terms of the three unknown integrals* $\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3$*. Then we read the* ϵ *orders of the subintegrals* $\tilde{\mathcal{J}}_1, \ldots, \tilde{\mathcal{J}}_4$ *in all the uncouplings*[107]*:*

Uncouplings	*MIs*	$\tilde{\mathcal{J}}_1$	$\tilde{\mathcal{J}}_2$	$\tilde{\mathcal{J}}_3$	$\tilde{\mathcal{J}}_4$
$\mathcal{U}_Z^{(1,3)}$	$\tilde{\mathcal{I}}_1^{(H,3)}$	*0*	*1*	*0*	*0*
	$\tilde{\mathcal{I}}_2^{(1,3)}$	*0*	*0*	*0*	*0*
	$\tilde{\mathcal{I}}_3^{(1,3)}$	*-1*	*0*	*-1*	*-1*
$\mathcal{U}_Z^{(2,3)}$	$\tilde{\mathcal{I}}_2^{(H,3)}$	*-1*	*0*	*1*	*-1*
	$\tilde{\mathcal{I}}_1^{(2,3)}$	*-2*	*-1*	*0*	*-2*
	$\tilde{\mathcal{I}}_3^{(2,3)}$	*0*	*0*	*0*	*0*
$\mathcal{U}_Z^{(3,3)}$	$\tilde{\mathcal{I}}_3^{(H,3)}$	*0*	*1*	*0*	*0*
	$\tilde{\mathcal{I}}_1^{(3,3)}$	*0*	*0*	*0*	*0*
	$\tilde{\mathcal{I}}_2^{(3,3)}$	*0*	*1*	*0*	*0*

(4.3.4)

First we need to select which is the best HODE that we should use. We use the following observation: the HODE dominates the ϵ*-orders and thus should be minimized first. Based on that, we proceed with the other unknowns to keep the orders as small as possible. To tackle the selection of the optimal HODE we proceed as follows: given a set of* p *equations, we define* p *non-negative integers* $n_1, \ldots n_p$ *and we initialise them all to* 0*. Then, for each BI in succesion we look at the* ϵ*-order and then for the equation where it is maximal, we take the corresponding* n_i *and increment them by one. At the end of the process, we select the equations corresponding to the maximal* n_i*. If there was only one optimal equation, we can take it and then perform the selection of the best equation for the next variable. If there are several equations, we simply take all of them and create a separate case for each, and then keep on selecting the next variable depending on each of them — we will then select which is the best overall system at the very end. Let us explicitely detail the procedure above to select the optimal HODE(s):*

1. *There are three HODE* $\tilde{\mathcal{I}}_1^{(H,3)}, \tilde{\mathcal{I}}_2^{(H,3)}, \tilde{\mathcal{I}}_3^{(H,3)}$*, so we initialize*

$$n_1 = 0, n_2 = 0, n_3 = 0.$$

[106] In particular, what the precise system looked like is not of interest for us.

[107] Here the number that appears in the column of $\tilde{\mathcal{J}}_i$ is the associated m_i appearing in the equation indicated at the head of the row.

2. *Let us look at the ϵ-orders for $\tilde{\mathcal{J}}_1$ for each of the equations:*

$$\left.\begin{array}{lcl} \tilde{\mathcal{I}}_1^{(H,3)} & : & 0 \\ \tilde{\mathcal{I}}_2^{(H,3)} & : & -1 \\ \tilde{\mathcal{I}}_3^{(H,3)} & : & 0 \end{array}\right\} \Rightarrow n_1 = 1, n_2 = 0, n_3 = 1.$$

n_1, n_3 are increased, as the best, i.e., the maximal ϵ-value is 0 and appears in $\tilde{\mathcal{I}}_1^{(H)}, \tilde{\mathcal{I}}_3^{(H)}$.

3. *Let us look at the ϵ-orders for $\tilde{\mathcal{J}}_2$ for each of the equations:*

$$\left.\begin{array}{lcl} \tilde{\mathcal{I}}_1^{(H,3)} & : & 1 \\ \tilde{\mathcal{I}}_2^{(H,3)} & : & 0 \\ \tilde{\mathcal{I}}_3^{(H,3)} & : & 1 \end{array}\right\} \Rightarrow n_1 = 2, n_2 = 0, n_3 = 2.$$

Again n_1, n_3 are increased, as the best, i.e., the maximal ϵ-value is 1 and appears in $\tilde{\mathcal{I}}_1^{(H)}, \tilde{\mathcal{I}}_3^{(H)}$.

4. *Let us look at the ϵ-orders for $\tilde{\mathcal{J}}_3$ for each of the equations:*

$$\left.\begin{array}{lcl} \tilde{\mathcal{I}}_1^{(H,3)} & : & 0 \\ \tilde{\mathcal{I}}_2^{(H,3)} & : & 1 \\ \tilde{\mathcal{I}}_3^{(H,3)} & : & 0 \end{array}\right\} \Rightarrow n_1 = 2, n_2 = 1, n_3 = 2.$$

This time only n_2 is increased, as the best, i.e., the maximal ϵ-value is 1 and appears only in $\tilde{\mathcal{I}}_2^{(H)}$.

5. *Let us look at the ϵ-orders for $\tilde{\mathcal{J}}_4$ for each of the equations:*

$$\left.\begin{array}{lcl} \tilde{\mathcal{I}}_1^{(H,3)} & : & 0 \\ \tilde{\mathcal{I}}_2^{(H,3)} & : & -1 \\ \tilde{\mathcal{I}}_3^{(H,3)} & : & 0 \end{array}\right\} \Rightarrow n_1 = 3, n_2 = 1, n_3 = 3.$$

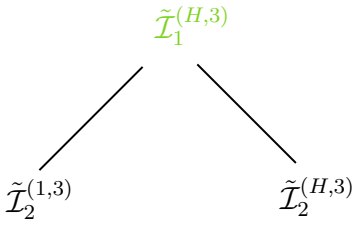
Thus n_1, n_3 are increased, as the best, i.e., the maximal ϵ-value is 3 and appears in $\tilde{\mathcal{I}}_1^{(H)}, \tilde{\mathcal{I}}_3^{(H)}$.

6. *In this case there are two maximal identical counting variables n_1 and n_3. So we have to select both $\tilde{\mathcal{I}}_1^{(H)}$ and $\tilde{\mathcal{I}}_3^{(H)}$.*

We represent the selection process also as follows:

	$\tilde{\mathcal{J}}_1$	$\tilde{\mathcal{J}}_2$	$\tilde{\mathcal{J}}_3$	$\tilde{\mathcal{J}}_4$
$\tilde{\mathcal{I}}_1^{(H,3)}$	0	1	0	0
$\tilde{\mathcal{I}}_2^{(H,3)}$	-1	0	1	-1
$\tilde{\mathcal{I}}_3^{(H,3)}$	0	1	0	0
$n_1^{(H,3)}$	1	2	2	(3)
$n_2^{(H,3)}$	0	0	1	1
$n_3^{(H,3)}$	1	2	2	(3)

Here the columns from left to right demonstrate the updates while processing the different BIs. The maximal values in the last column determine which HODE are preferable. Let us concentrate first on the case where we choose $\tilde{\mathcal{I}}_1^{(H,3)}$*:*

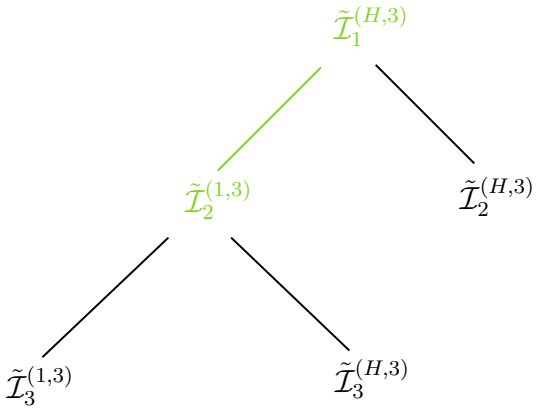


Now we have two integrals and the connected equations left. We proceed as previously (i.e., we define two variables that we increment according to the best choice). The two possible choices are done according to the first equation that has been selected, in particular because we assume that now we know[108] $\tilde{\mathcal{I}}_1$*, we can either immediately obtain* $\tilde{\mathcal{I}}_2$ *from* $\tilde{\mathcal{I}}_1$ *using* $\tilde{\mathcal{I}}_2^{(1,3)}$*, or we can simply solve the differential equation for* $\tilde{\mathcal{I}}_2$ *obtained when uncoupling in* $\tilde{\mathcal{I}}_2$*:*

	$\tilde{\mathcal{J}}_1$	$\tilde{\mathcal{J}}_2$	$\tilde{\mathcal{J}}_3$	$\tilde{\mathcal{J}}_4$
$\tilde{\mathcal{I}}_2^{(1,3)}$	0	0	0	0
$\tilde{\mathcal{I}}_2^{(H,3)}$	-1	0	1	-1
$n_2^{(1,3)}$	1	2	2	(3)
$n_2^{(H,3)}$	0	1	2	2

In this instant, we skip the $\tilde{\mathcal{I}}_2^{(H,3)}$ *path and we are left to tackle* $\tilde{\mathcal{I}}_2^{(1,3)}$*. Choosing this branch of the tree, we arrive at the final choice for* $\tilde{\mathcal{I}}_3$*. Once more, we can use either HODE in* $\tilde{\mathcal{I}}_3$ *or LF expressing* $\tilde{\mathcal{I}}_3$ *as a function of* $\tilde{\mathcal{I}}_1$*. In short, we focus on the left subtree of*

[108] When we say "we know" a certain MI, it is always implied "up to a certain order in ϵ".

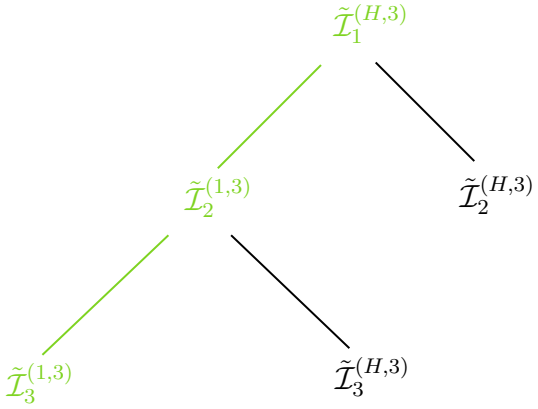


The ϵ-order comparison gives us:

	$\tilde{\mathcal{J}}_1$	$\tilde{\mathcal{J}}_2$	$\tilde{\mathcal{J}}_3$	$\tilde{\mathcal{J}}_4$
$\tilde{\mathcal{I}}_3^{(1,3)}$	*-1*	*0*	*-1*	*-1*
$\tilde{\mathcal{I}}_3^{(H,3)}$	*0*	*1*	*0*	*0*
$n_3^{(1,3)}$	*0*	*0*	*0*	*0*
$n_3^{(H,3)}$	*1*	*2*	*3*	*(4)*

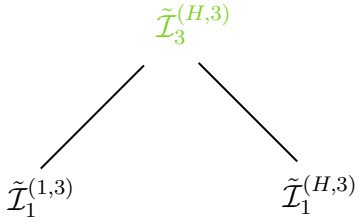
Hence using the HODE $\tilde{\mathcal{I}}_3^{(H,3)}$ is the best choice. Summarising, the first possible solution is obtained by taking the following three equations:

$$S_1: \quad \tilde{\mathcal{I}}_1^{(H,3)},\ \tilde{\mathcal{I}}_2^{(1,3)},\ \tilde{\mathcal{I}}_3^{(H,3)}$$

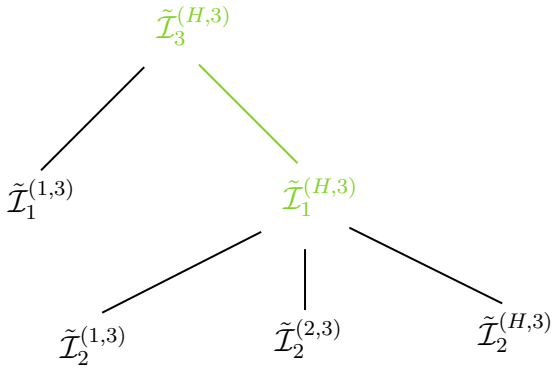


The other possible choice is taking $HODE(\tilde{\mathcal{I}}_3)$. We have now to proceed with this second tree of possibilities, taking it as the first equation. We now need to select the best equation for $\tilde{\mathcal{I}}_1$[109]*:*

[109] We always proceed cyclically for the moment, i.e. if the best original equation allows us to obtain $\tilde{\mathcal{I}}_j$, then we search next for $\tilde{\mathcal{I}}_{j+1}$, then $\tilde{\mathcal{I}}_{j+2}$, where $j \in \{1, \ldots, n\}$ and we take $(j \bmod n) + 1$ if needed.



which turns out to be $\tilde{\mathcal{I}}_1^{(H,3)}$ *here. The final equation we need is the one for* $\tilde{\mathcal{I}}_2$*. We have three possible choices for the last step: the first possible choice is that once again, knowing that we got* $\tilde{\mathcal{I}}_1$ *by solving* $HODE(\tilde{\mathcal{I}}_1)$*, we can use the relation* $\tilde{\mathcal{I}}_2^{(1,3)}$*. But because we also know* $\tilde{\mathcal{I}}_2$ *now, we can use the relation* $\tilde{\mathcal{I}}_2^{(2,3)}$*, that we know from uncoupling in* $\tilde{\mathcal{I}}_2$ *— that is to say, we use the information from other paths. Finally, as usual, we could simply solve* $HODE(\tilde{\mathcal{I}}_3)$*.*



Here are now the ϵ*-order comparisons:*

	$\tilde{\mathcal{J}}_1$	$\tilde{\mathcal{J}}_2$	$\tilde{\mathcal{J}}_3$	$\tilde{\mathcal{J}}_4$
$\tilde{\mathcal{I}}_2^{(1)}$	0	0	0	0
$\tilde{\mathcal{I}}_2^{(3)}$	0	1	0	0
$\tilde{\mathcal{I}}_2^{(H)}$	-1	0	1	-1
$n_2^{(1)}$	1	1	1	2
$n_2^{(3)}$	1	2	2	(3)
$n_2^{(H)}$	0	0	1	1

Summarisng, the best tactic for the second attempt is the solution:

$$S_2: \quad \tilde{\mathcal{I}}_1^{(H,3)},\ \tilde{\mathcal{I}}_2^{(3,3)},\ \tilde{\mathcal{I}}_3^{(H,3)}.$$

By doing the steps that we have just done, we basically did the step 3 of the sketch of the algorithm we described just before — taking as input the uncouplings and ϵ*-orders for each of the relations and equations, we managed to select what seems to be the optimal solutions. In this example both of the solutions* S_1 *and* S_2 *are mostly the same and the only difference is whether we should use* $\tilde{\mathcal{I}}_2^{(3)}$ *or* $\tilde{\mathcal{I}}_2^{(1)}$*.*

Now we proceed with step 4 where we need to find a way to decide which of the tactics is preferable. The idea is as follows:

- For each "good" uncoupling, let us denote with $m_{i,j}$ the $\epsilon-$order for the BI $\tilde{\mathcal{J}}_j(x,\epsilon)$, $1 \leq j \leq m$ appearing in the expression for the MI $\tilde{\mathcal{I}}_i(x,\epsilon)$, $1 \leq i \leq n$, and write the order matrix:

$$M = (m_{i,j})_{1\leq i\leq n, 1\leq j\leq m}.$$

- We define two functions:
  - A function that sums all the elements of a $m \times n$ matrix:

$$\sigma : \begin{cases} \mathcal{M}_{n,m}(\mathbb{Z}) & \longrightarrow & \mathbb{Z} \\ M = (m_{i,j})_{\substack{1\leq i\leq n \\ 1\leq j\leq m}} & \longmapsto & \sum_{\substack{1\leq i\leq n \\ 1\leq j\leq m}} m_{i,j} \end{cases} \tag{4.3.5}$$

  - The Frobenius norm of an $m \times n$ matrix:

$$\|\bullet\| : \begin{cases} \mathcal{M}_{n,m}(\mathbb{Z}) & \longrightarrow & \mathbb{R}_+ \\ M & \longmapsto & \sqrt{\mathrm{Tr}(M^\top M)} \end{cases} \tag{4.3.6}$$

    where $\mathrm{Tr}\, M$ denotes the trace of M, and $M^\top$ is the transpose of the matrix M.

- We obtain the "best" solution by finding the element that maximizes the sum of orders σ and in the case of equality minimizes their variance $\|\bullet\|$, or equivalently maximizes $-\|\bullet\|$.

Toy example: Let us illustrate the reason for these criterions when we select a good solution with an example. Suppose we are given the following matrices:

$$\tilde{M}_1 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \quad \tilde{M}_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & -2 & 0 & 0 \end{pmatrix}.$$

The first important aspect is of course the total number of coefficients we will need to compute for the BIs, which we get by summing up the orders of the elements of the matrices — in the example above, we obtain $\sigma(\tilde{M}_1) = \sigma(\tilde{M}_2) = 3$. The problem is that the second matrix has values whose standard deviation is higher and some of the coefficients are actually lower than 0 (e.g. $\left(\tilde{M}_2\right)_{3,2} = -2$), i.e., we actually need to expand certain BIs to higher orders. In order to avoid this, we have to use the Frobenius norm $\|\bullet\|$ to distinguish between both cases. Because $\left\|\tilde{M}_2\right\| = \sqrt{21} > \sqrt{3} = \left\|\tilde{M}_1\right\|$, there is more deviation for the values with the larger norm, and therefore we finally select $\tilde{M}_1$, i.e., the matrix with the minimal norm in absolute value.

Example 4.3.1 (continued)**.** *In our particular case, we denote with M_1 the order matrix for the first solution S_1 and with M_2 the order matrix for the second solution S_2. We indicate on the right side of each row which is the associated equation, and on top of each column the BIs:*

$$M_1 = \begin{array}{c} \begin{matrix} \tilde{\mathcal{J}}_1 & \tilde{\mathcal{J}}_2 & \tilde{\mathcal{J}}_3 & \tilde{\mathcal{J}}_4 \end{matrix} \\ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} \tilde{\mathcal{I}}_1^{(H,3)} \\ \tilde{\mathcal{I}}_2^{(1,3)} \\ \tilde{\mathcal{I}}_3^{(H,3)} \end{matrix}, \quad M_2 = \begin{array}{c} \begin{matrix} \tilde{\mathcal{J}}_1 & \tilde{\mathcal{J}}_2 & \tilde{\mathcal{J}}_3 & \tilde{\mathcal{J}}_4 \end{matrix} \\ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \end{array} \begin{matrix} \tilde{\mathcal{I}}_1^{(H,3)} \\ \tilde{\mathcal{I}}_2^{(3,3)} \\ \tilde{\mathcal{I}}_3^{(H,3)} \end{matrix}$$

We have $\sigma(M_1) = 2 < \sigma(M_2)$ so we can immediately pick M_2 as the solution and more precisely we obtain finally that the best uncoupling is

$$\tilde{\mathcal{I}}_1^{(H,3)},\ \tilde{\mathcal{I}}_2^{(1,3)},\ \tilde{\mathcal{I}}_3^{(H,3)}.$$

4.3.2.1. Simple linear selection method

With this example above, we can now formalise properly the steps above in the following method, that we call the **simple linear selection method**:

Input: An irreducible differential system $\mathcal{S}$ of order n with k BIs $\tilde{\mathcal{J}}_i$:

$$\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{I}}(x,\epsilon) = \mathcal{M}(x,\epsilon)\cdot\tilde{\mathcal{I}}(x,\epsilon) + \tilde{\mathcal{R}}(x,\epsilon), \tag{4.3.7}$$

where

$$\forall i \in \{1,\dots,n\},\ \mathcal{R}_i(x,\epsilon) = \sum_{p=1}^{k} \varphi_{i,p}(x,\epsilon)\tilde{\mathcal{J}}_p(x,\epsilon), \quad \varphi_{i,p}(x,\epsilon) \in \mathbb{K}(x,\epsilon). \tag{4.3.8}$$

Output: A preferable set of HODEs and LFs.

Simple linear selection method:

1. Uncouple the irreducible system in all the possible ways by circular permutation of the variables $\tilde{\mathcal{I}}_i$. Given that the system is of size n, we get

$$n^2 = \underbrace{n}_{\text{HODEs}} + \underbrace{n(n-1)}_{\text{LFs}}$$

equations. More precisely we uncouple the n systems

$$\begin{aligned}
\{\underline{\tilde{\mathcal{I}}_1}, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \dots, \tilde{\mathcal{I}}_{n-2}, \tilde{\mathcal{I}}_{n-1}, \tilde{\mathcal{I}}_n\} &\text{ in } \tilde{\mathcal{I}}_1 \\
\{\underline{\tilde{\mathcal{I}}_2}, \tilde{\mathcal{I}}_3, \tilde{\mathcal{I}}_4, \dots, \tilde{\mathcal{I}}_{n-1}, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_1\} &\text{ in } \tilde{\mathcal{I}}_2 \\
\{\underline{\tilde{\mathcal{I}}_3}, \tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5, \dots, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2\} &\text{ in } \tilde{\mathcal{I}}_3 \qquad (4.3.9) \\
&\vdots \qquad (4.3.10) \\
\{\underline{\tilde{\mathcal{I}}_n}, \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \dots, \tilde{\mathcal{I}}_{n-3}, \tilde{\mathcal{I}}_{n-2}, \tilde{\mathcal{I}}_{n-1}\} &\text{ in } \tilde{\mathcal{I}}_n.
\end{aligned}$$

We select now those for which we can "invert" the $\boxed{3}$ relations of Zürcher's output on page 196, i.e., for those where we can get rid of the dummy W_i variables and obtain uncouplings in $\tilde{\mathcal{I}}_p$, $p \in \{1, \dots, n\}$, of the form described in Remark 4.3.1:

$$\mathcal{U}_Z^{(p,n)} : \left\{ \underbrace{\tilde{\mathcal{I}}_p^{(H,n)}}_{\text{HODE}}, \underbrace{\tilde{\mathcal{I}}_{p+1}^{(p,n)}, \dots, \tilde{\mathcal{I}}_n^{(p,n)}, \tilde{\mathcal{I}}_1^{(p,n)}, \dots, \tilde{\mathcal{I}}_{p-1}^{(1,n)}}_{\text{LFs}} \right\} \qquad (4.3.11)$$

where we remind that we denote:

$$\mathcal{I}_p^{(H)} : \text{HODE}\left(\underline{\tilde{\mathcal{I}}_p}, \{\mathrm{d}_x^j \tilde{\mathcal{I}}_p\}; \{\tilde{\mathcal{J}}_i, \mathrm{d}_x^j \tilde{\mathcal{J}}_i\}\right), \qquad (4.3.12)$$

$$\forall q \in \{2, \dots, n\}, \quad \tilde{\mathcal{I}}_q^{(p)} : \tilde{\mathcal{I}}_q = F_q(\tilde{\mathcal{I}}_p, \{\mathrm{d}_x^j \tilde{\mathcal{I}}_p\}; \{\tilde{\mathcal{J}}_i, \mathrm{d}_x^j \tilde{\mathcal{J}}_i\}). \qquad (4.3.13)$$

Note that there are rare cases where the inversing only works by a change of basis. We skip here these exotic cases for simplicity; see also Remark 4.3.7.

In total, supposing we managed to eliminate the W_i dependence in all the uncouplings, we should obtain the following set of all available relations:

$$\left\{ \begin{array}{c}
\left\{\tilde{\mathcal{I}}_1^{(H,n)}, \tilde{\mathcal{I}}_2^{(1,n)}, \tilde{\mathcal{I}}_3^{(1,n)}, \dots, \tilde{\mathcal{I}}_{n-2}^{(1,n)}, \tilde{\mathcal{I}}_{n-1}^{(1,n)}\right\} \\
\left\{\tilde{\mathcal{I}}_2^{(H,n)}, \tilde{\mathcal{I}}_3^{(2,n)}, \dots, \tilde{\mathcal{I}}_{n-1}^{(2,n)}, \tilde{\mathcal{I}}_n^{(2,n)}, \tilde{\mathcal{I}}_1^{(2,n)}\right\} \\
\vdots \\
\left\{\tilde{\mathcal{I}}_p^{(H,n)}, \tilde{\mathcal{I}}_{p+1}^{(p,n)}, \dots, \tilde{\mathcal{I}}_n^{(p,n)}, \tilde{\mathcal{I}}_1^{(p,n)}, \dots, \tilde{\mathcal{I}}_{p-1}^{(1,n)}\right\} \\
\vdots \\
\left\{\tilde{\mathcal{I}}_n^{(H)}, \tilde{\mathcal{I}}_1^{(n,n)}, \tilde{\mathcal{I}}_2^{(n,n)}, \dots, \tilde{\mathcal{I}}_{n-2}^{(n,n)}, \tilde{\mathcal{I}}_{n-1}^{(n,n)}\right\}
\end{array} \right\}. \qquad (4.3.14)$$

2. Each relation from (4.3.14) is of the form

$$\forall i,j \in \{1,\dots,n\},\quad \tilde{\mathcal{I}}_j^{(i,n)} : \tilde{\mathcal{I}}_j = \bar{F}_j(\tilde{\mathcal{I}}_i, \{\mathrm{d}_x^p \tilde{\mathcal{I}}_i\}) + \sum_{q=0}^{Q^{(i,j)}} \sum_{l=1}^{k} \epsilon^{m_{l,q}^{(i,j)}} \frac{\alpha_{l,q}^{(i,j)}(x,\epsilon)}{\beta_{l,q}^{(i,j)}(x,\epsilon)} \mathrm{d}_x^q \tilde{\mathcal{J}}_l \tag{4.3.15}$$

$$\forall i \in \{1,\dots,n\},\quad \tilde{\mathcal{I}}_i^{(H,n)} : \bar{F}_H(\tilde{\mathcal{I}}_i, \{\mathrm{d}_x^p \tilde{\mathcal{I}}_i\}) = \sum_{q=0}^{Q^{(H,j)}} \sum_{l=1}^{k} \epsilon^{m_{l,q}^{(H,i)}} \frac{\alpha_{l,q}^{(H,i)}(x,\epsilon)}{\beta_{l,q}^{(H,i)}(x,\epsilon)} \mathrm{d}_x^q \tilde{\mathcal{J}}_l \tag{4.3.16}$$

where for $i,j \in \{H,1,2,\dots,n\}$, $\bar{F}_i(\tilde{\mathcal{I}}_j, \{\mathrm{d}_x^p \tilde{\mathcal{I}}_j\})$ are linear combinations of the $\tilde{\mathcal{I}}_j, \mathrm{d}_x^p \tilde{\mathcal{I}}_j$ and for all $i \in \{H,1,2,\dots,n\}$, $j \in \{1,2,\dots,n\}$ and $q \in \{0,\dots,Q^{(i,j)}\}$:

$$\alpha_{q,l}^{(i,j)}(x,\epsilon), \beta_{q,l}^{(i,j)}(x,\epsilon) \in \mathbb{K}[x,\epsilon] \text{ and } \epsilon \nmid \alpha_{l,q}^{(i,j)}(x,\epsilon), \beta_{l,q}^{(i,j)}(x,\epsilon).$$

It means in particular that for each relation $\tilde{\mathcal{I}}_j^{(i,n)}$ and to each $\tilde{\mathcal{J}}_l$, we can associate the relative ϵ order $m_{l,q}^{(i,j)}$. More precisely, because we are considering derivatives of $\tilde{\mathcal{J}}_l$, we have to take the "worst" contribution of ϵ into account which is the one that will be minimal, i.e.,

$$\bar{m}_l^{(i,j)} := \min_{0 \le q \le Q^{(i,j)}} \left(m_{l,q}^{(i,j)} \right).$$

We can therefore build the following ϵ-order lists for all $i \in \{H,1,\dots,n\}$, $j \in \{1,\dots,n\}$:

$$\mathcal{O}(\tilde{\mathcal{I}}_j^{(i,n)}) := \left(\bar{m}_1^{(i,j)}, \dots, \bar{m}_l^{(i,j)}, \dots, \bar{m}_k^{(i,j)} \right). \tag{4.3.17}$$

Now for a given set of ϵ-order lists $\left\{ \mathcal{O}\left(\tilde{\mathcal{I}}_{j_1}^{(i_1,n)}\right), \dots, \mathcal{O}\left(\tilde{\mathcal{I}}_{j_q}^{(i_q,n)}\right) \right\}$, $q \in \mathbb{N}$, we define the following procedure to select an optimal equation with its ϵ-order:

Equation selection procedure $\mathcal{F}$

- Define and initialise the variables:

$$n_{j_1}^{(i_1)} = 0, \ldots, n_{j_q}^{(i_q)} = 0.$$

- Compare the $\bar{m}_1^{(i_1,j_1)}, \ldots, \bar{m}_1^{(i_q,j_q)}$. For all the maximal values $\bar{m}_1^{(i_\ell,j_\ell)}$, increment $n_{j_\ell}^{(i_\ell)} :\leftarrow n_{j_\ell}^{(i_\ell)} + 1$.
- Do this in order for all the $l \in \{2, \ldots, k\}$, $\bar{m}_l^{(i_1,j_1)}$.
- Compare the $n_{j_\ell}^{(i_\ell)}$, $\ell \in \{1, \ldots, q\}$ values at the end and for all values ℓ such that $n_{j_\ell}^{(i_\ell)}$ is maximal, return the corresponding ℓ.

We use the equation selection $\mathcal{F}\left(\mathcal{O}\left(\tilde{\mathcal{I}}_1^{(H,n)}\right), \mathcal{O}\left(\tilde{\mathcal{I}}_2^{(H,n)}\right), \ldots, \mathcal{O}\left(\tilde{\mathcal{I}}_n^{(H,n)}\right)\right)$, i.e., we select the optimal equation from all the HODEs. Suppose that we obtain the following list of $s \leq n$ optimal HODEs

$$(\ell_1 \ldots, \ell_s)\,. \tag{4.3.18}$$

3. For each of the best equations $\tilde{\mathcal{I}}_{\ell_p}^{(H,n)}$, we can relabel the relations so that

$$\tilde{\mathcal{I}}_{\ell_p}^{(H,n)} \longrightarrow \tilde{\mathcal{I}}_1^{(H,n)},$$

and then without loss of generality we consider the case where the best HODE is for $\tilde{\mathcal{I}}_1^{(H,n)}$. We can immediately build the full tree of selections, which will have the following form:

$$\begin{array}{lccccccccc}
R_1: & & & & & \tilde{\mathcal{I}}_1^{(H,n)} & & & & \\
R_2: & & & & \tilde{\mathcal{I}}_2^{(H,n)} & & & \tilde{\mathcal{I}}_2^{(1,n)} & & \\
R_3: & & & \tilde{\mathcal{I}}_3^{(H,n)} & & & \tilde{\mathcal{I}}_3^{(1,n)} & & \tilde{\mathcal{I}}_3^{(2,n)} & \\
 & & \iddots & & \vdots & & \vdots & & \ddots & \\
R_n: & \tilde{\mathcal{I}}_n^{(H,n)} & \tilde{\mathcal{I}}_n^{(1,n)} & \cdots & \cdots & \cdots & \cdots & \cdots & \tilde{\mathcal{I}}_n^{(n-2,n)} & \tilde{\mathcal{I}}_n^{(n-1,n)}
\end{array}$$

The list of best uncouplings $\tilde{\mathcal{B}}$ contains in the initialisation phase one element:

$$\tilde{\mathcal{B}} := \left(\left(\tilde{\mathcal{I}}_1^{(H,n)}\right)\right).$$

4. To each row R_i we can associate

$$\underbrace{\left(\tilde{\mathcal{I}}_i^{(H,n)}, \tilde{\mathcal{I}}_i^{(1,n)}, \tilde{\mathcal{I}}_i^{(2,n)}, \dots, \tilde{\mathcal{I}}_i^{(i-1,n)}\right)}_{R_i} \mapsto \underbrace{\left(\mathcal{O}\left(\tilde{\mathcal{I}}_i^{(H,n)}\right), \tilde{\mathcal{O}}\left(\tilde{\mathcal{I}}_i^{(1,n)}\right), \tilde{\mathcal{O}}\left(\tilde{\mathcal{I}}_i^{(2,n)}\right), \dots, \tilde{\mathcal{O}}\left(\tilde{\mathcal{I}}_i^{(i-1,n)}\right)\right)}_{=:\bar{R}_i},$$

and we apply now the equation selection procedure $\mathcal{F}$ on every row (or layer) $\bar{R}_i$, $i \in \{2, \dots, n\}$, to select the best path. At each layer j, computing $\mathcal{F}(R_j)$ yields $q \geq 1$ possible optimal solutions $S = \{\tilde{\mathcal{I}}_j^{(i_1,n)}, \dots, \tilde{\mathcal{I}}_j^{(i_q,n)}\}$. Given that after the step $j-1$, $\tilde{\mathcal{B}}$ is of the form

$$\tilde{\mathcal{B}} = (\dots, (\tilde{\mathcal{I}}_1^{(H,n)}, \tilde{\mathcal{I}}_2^{(k_p^2,n)}, \dots, \tilde{\mathcal{I}}_{j-1}^{(k_p^{j-1},n)}), \dots)$$

for $p \in \mathbb{N}^\star$ and $k_p^r \leq n$, we then enlarge $\tilde{\mathcal{B}}$ in the following way:

$$\tilde{\mathcal{B}} :\leftarrow (\dots, (\tilde{\mathcal{I}}_1^{(H,n)}, \dots, \tilde{\mathcal{I}}_{j-1}^{(k_p^{j-1},n)}, \tilde{\mathcal{I}}_j^{(i_1,n)}), \dots, (\tilde{\mathcal{I}}_1^{(H,n)}, \dots, \tilde{\mathcal{I}}_{j-1}^{(i_p^{j-1},n)}, \tilde{\mathcal{I}}_j^{(i_q,n)}) \dots)$$

After iterating this on all the rows, $\tilde{\mathcal{B}}$ is the list of all the possible candidates for the preferable set of equations.

5. It just remains to distringuish the best solutions among them, i.e., to reduce their number as much as possible. For that, to each sublist $\mathcal{L}_p$ of $\tilde{\mathcal{B}}$ we can associate a matrix of ϵ orders:

$$\tilde{\mathcal{L}}_p = \left(\tilde{\mathcal{I}}_{j_{p,1}}^{(i_{p,1},n)}, \dots, \tilde{\mathcal{I}}_{j_{p,n}}^{(i_{p,n},n)}\right) \mapsto \tilde{\mathcal{M}}_p := \begin{pmatrix} \mathcal{O}\left(\tilde{\mathcal{I}}_{j_{p,1}}^{(i_{p,1},n)}\right) \\ \vdots \\ \mathcal{O}\left(\tilde{\mathcal{I}}_{j_{p,n}}^{(i_{p,n},n)}\right) \end{pmatrix}$$

where $\mathcal{O}\left(\tilde{\mathcal{I}}_{j_{p,j}}^{(i_{p,j},n)}\right)$ builds a row of the matrix. Now apply first σ on all the matrices $\tilde{\mathcal{M}}_p$ and take those where $\sigma\left(\tilde{\mathcal{M}}_p\right)$ is maximal, and then apply $-\|\bullet\|$ on the remaining matrices and take those where $-\left\|\tilde{\mathcal{M}}_p\right\|$ is maximal. Those elements will correspond to the optimal solutions.

4.3.3. Refining the uncoupling scheme

In the section above, we focused on the most simple scheme, i.e., the circular uncoupling in all of the variables. Nevertheless, this approach is still rather rough,

at least for two reasons. First, all the uncouplings are of order n, as we just permute the variables, and therefore, by Zürcher's scheme specifications, we will get at worst n HODEs of order n. The second issue is that at the moment we select the different equations "in sequence" for the $\tilde{\mathcal{I}}_i$, i.e., if the best HODE turned out to be $\tilde{\mathcal{I}}_p^{(H,n)}$, it comes from the uncoupling

$$\left\{\tilde{\mathcal{I}}_p^{(H,n)}, \tilde{\mathcal{I}}_{p+1}^{(p,n)}, \ldots, \tilde{\mathcal{I}}_n^{(p,n)}, \tilde{\mathcal{I}}_1^{(p,n)}, \ldots, \tilde{\mathcal{I}}_{p-1}^{(1,n)}\right\},$$

and we will then select the best equation only for $\tilde{\mathcal{I}}_{p+1}$, then for $\tilde{\mathcal{I}}_{p+2}$, etc[110]. In the following we will focus on various refinements.

4.3.3.1. Initial conditions optimizing procedure

While the cyclic uncoupling improves the $\epsilon-$orders by allowing several choices, it does so at the expense of the orders of HODE that we have to solve. In order to improve this situation, we emphasize the following: the equations are selected in such a way that they can be solved using only the previous MIs[111] if they are HODEs, or simply plugging in the previous MIs if they are LFs. In particular, once one gets an equation for a MI $\tilde{\mathcal{I}}_i$, be it a HODE or a LF, one can actually consider that it *becomes a BI* for all the MIs we have not selected yet. This means we can consider an MI that is fixed in the choice tree as a BI and to uncouple for the next variables in this configuration, which reduces the degree of the HODE by 1 for each such fixed MI. Let us now describe more precisely this improved version of the simple linear selection method. We consider once again an irreducible system $\mathcal{S}$ of order n with k BIs $\tilde{\mathcal{J}}_i$. We also consider that we have done the first two steps of the simple linear selection method, i.e.:

1. We uncouple first cyclically $\mathcal{S}$ in n MIs $\tilde{\mathcal{I}}_i$, $i \in \{1, \ldots, n\}$ and m BIs $\tilde{\mathcal{J}}_j$, $j \in \{1, \ldots, m\}$ so that we get

$$n^2 = \underbrace{n}_{\text{HODE}} + \underbrace{n(n-1)}_{\text{LFs}} \quad \text{equations.}$$

   As in the Step 1 of the simple linear selection method, we obtain the same set of uncouplings of the form (4.3.9).

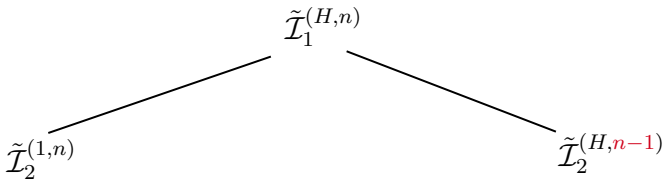
2. We choose the best HODE by comparing the $\epsilon-$orders for the BIs in the RHS, i.e., using once again the selection function $\mathcal{F}$.

[110] As we pointed out in the previous subsection, what happens actually is that whatever the best HODE is, we will relabel the $\tilde{\mathcal{I}}_i$ such that $\tilde{\mathcal{I}}_p^{(H,n)}$ becomes $\tilde{\mathcal{I}}_1^{(H,n)}$ for technical reasons, but it does not change the point of the argument.

[111] And as usual the BIs.

What will change is the third step, i.e., writing down the selection tree:

- Suppose that the best HODE is $\tilde{\mathcal{I}}_i^{(H,n)}$. We have to find now the best equations among
$$\tilde{\mathcal{I}}_1, \dots, \tilde{\mathcal{I}}_{i-1}, \tilde{\mathcal{I}}_{i+1}, \dots, \tilde{\mathcal{I}}_n$$
in this exact order. Suppose without loss of generality now that $i = 1$. The next equation we need to find is for $\tilde{\mathcal{I}}_2$. Thus, we uncouple the subsystem $\tilde{\mathcal{S}}_1$ in the unknowns $\tilde{\mathcal{I}}_2, \dots, \tilde{\mathcal{I}}_n$, where $\tilde{\mathcal{I}}_1$ is seen as a BI. In short, the MIs and BIs are changed from the original system $\mathcal{S}$ to $\tilde{\mathcal{S}}_1$ in the following way:
$$\begin{array}{cccc} & \mathcal{S} & & \tilde{\mathcal{S}}_1 \\ \text{MIs:} & \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \dots, \tilde{\mathcal{I}}_n & \longrightarrow & \tilde{\mathcal{I}}_2, \dots, \tilde{\mathcal{I}}_n \\ \text{BIs:} & \tilde{\mathcal{J}}_1, \dots, \tilde{\mathcal{J}}_m & \longrightarrow & \tilde{\mathcal{I}}_1, \tilde{\mathcal{J}}_2, \dots, \tilde{\mathcal{J}}_m \end{array}$$
We have now in total
$$n^2 + n + 1 = \underbrace{n}_{\text{HODE}^{(n)}} + \underbrace{n(n-1)}_{\text{LFs}^{(n)}} + \underbrace{1}_{\text{HODE}^{(n-1)}} + \underbrace{(n-2)}_{\text{LFs}^{(n-1)}} \quad \text{relations}$$
where by $\text{HODE}^{(j)}$ we mean HODE of degree j, and $\text{LF}^{(j)}$ meaning LF coming from an uncoupling of degree j. We have therefore the following tree of choices:

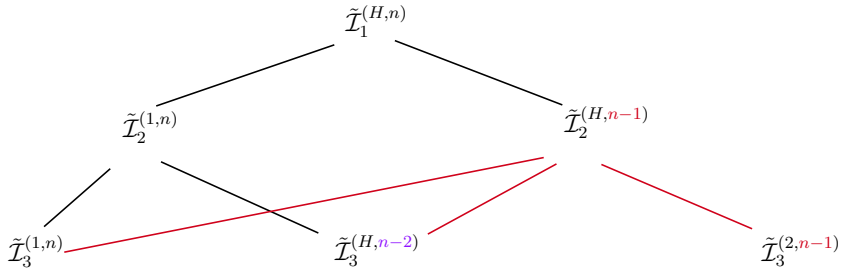


More precisely, compared to the original algorithm, we replaced the $\tilde{\mathcal{I}}_2^{(H,n)}$ by $\tilde{\mathcal{I}}_2^{(H,n-1)}$, since this allows to strictly reduce the degree of the HODE and hence the number of initial values for solving. We choose the best equation in the tree using the same method as described before.

- We reiterate the operation for $\tilde{\mathcal{I}}_3$. More precisely, since we have obtained $\tilde{\mathcal{I}}_1$ and $\tilde{\mathcal{I}}_2$, we uncouple the new subsystem $\tilde{\mathcal{S}}_2$ in the unknowns $\tilde{\mathcal{I}}_3, \dots, \tilde{\mathcal{I}}_n$ where:
$$\begin{array}{cccc} & \tilde{\mathcal{S}}_1 & & \tilde{\mathcal{S}}_2 \\ \text{MIs:} & \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \dots, \tilde{\mathcal{I}}_n & \longrightarrow & \tilde{\mathcal{I}}_3, \dots, \tilde{\mathcal{I}}_n \\ \text{BIs:} & \tilde{\mathcal{I}}_1, \tilde{\mathcal{J}}_1, \dots, \tilde{\mathcal{J}}_m & \longrightarrow & \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \tilde{\mathcal{J}}_2, \dots, \tilde{\mathcal{J}}_m. \end{array}$$
We obtain

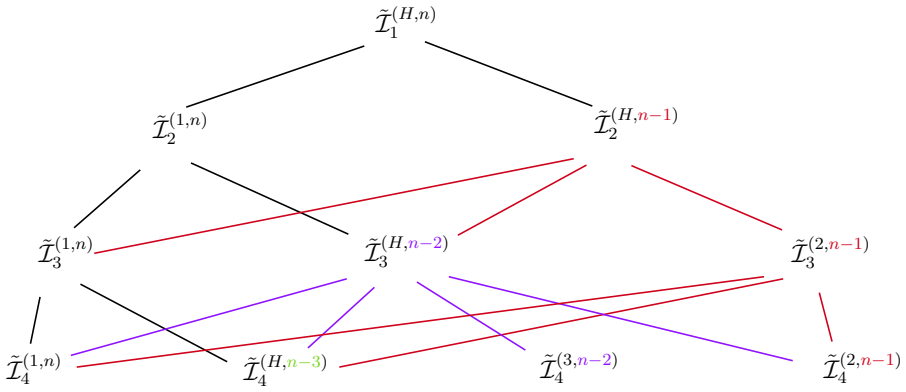
$$\begin{aligned} & n^2 + (n-1) + (n-2) \\ = & \underbrace{n}_{\text{HODE}^{(n)}} + \underbrace{n(n-1)}_{\text{LFs}^{(n)}} + \underbrace{1}_{\text{HODE}^{(n-1)}} + \underbrace{(n-2)}_{\text{LFs}^{(n-1)}} + \underbrace{1}_{\text{HODE}^{(n-2)}} + \underbrace{(n-3)}_{\text{LFs}^{(n-2)}} \quad \text{relations.} \end{aligned} \tag{4.3.19}$$

- The tree takes the following form:



  We choose the best equation as usually and then we go on to $\tilde{\mathcal{I}}_4$.

- The choise tree we get for $\tilde{\mathcal{I}}_4$ is



- We reiterate this procedure until we choose all the equations.

We can now rewrite the Step 3 from Section 4.3.2.1 as:

3. For each of the best equations $\tilde{\mathcal{I}}_{\ell_p}^{(H,n)}$, we can relabel the relations so that

$$\tilde{\mathcal{I}}_{\ell_p}^{(H,n)} \longrightarrow \tilde{\mathcal{I}}_1^{(H,n)},$$

   and then without loss of generality we consider the case where the best HODE is for $\tilde{\mathcal{I}}_1^{(H,n)}$ and we relabel the system accordingly. We define now

the subsystems $\tilde{S}_j$, $j \in \{1, \dots, n-1\}$ as

$$\tilde{S}_j : \frac{\mathrm{d}}{\mathrm{d}x}\tilde{\tilde{\mathcal{I}}}^{(j)}(x,\epsilon) = \tilde{\mathcal{M}}^{(j)}(x,\epsilon) \cdot \tilde{\mathcal{I}}^{(j)}(x,\epsilon) + \tilde{\mathcal{R}}^{(j)}(x,\epsilon) \tag{4.3.20}$$

where

$$\tilde{\mathcal{I}}^{(j)}(x,\epsilon) = \left(\tilde{\mathcal{I}}_{j+1}(x,\epsilon), \dots, \tilde{\mathcal{I}}_n(x,\epsilon)\right)^\intercal,$$
$$\tilde{\mathcal{R}}^{(j)}(x,\epsilon) = \left(\tilde{\mathcal{R}}_{j+1}(x,\epsilon), \dots, \tilde{\mathcal{R}}_n(x,\epsilon)\right)^\intercal.$$

If we denote the coefficient matrix as

$$\tilde{\mathcal{M}}(x,\epsilon) = (m_{p,q})_{p,q\in\{1,\dots,n\}} \in \mathcal{M}_{n\times n}(\mathbb{K}(x,\epsilon)),$$

then the submatrix $\mathcal{M}^{(j)}$ is defined as

$$\tilde{\mathcal{M}}^{(j)}(x,\epsilon) = (m_{p+j,q+j})_{p,q\in\{1,\dots,n-j\}} \in \mathcal{M}_{(n-j)\times(n-j)}(\mathbb{K}(x,\epsilon)).$$

Finally the RHS subvector contains the $\tilde{\mathcal{J}}_i$, $i \in \{1, \dots, k\}$ and the $\tilde{\mathcal{I}}_i$, $i \in \{1, \dots, j\}$:

$$\forall i \in \{1, \dots, n\},\ \mathcal{R}_i^{(j)}(x,\epsilon) = \sum_{p=1}^{k} \varphi_{i,p}(x,\epsilon)\tilde{\mathcal{J}}_p(x,\epsilon) + \sum_{\ell=1}^{j} \bar{\varphi}_{i,\ell}(x,\epsilon)\tilde{\mathcal{I}}_\ell(x,\epsilon), \tag{4.3.21}$$

where

$$\varphi_{i,p}(x,\epsilon), \bar{\varphi}_{i,\ell}(x,\epsilon) \in \mathbb{K}(x,\epsilon).$$

We uncouple now all systems $\tilde{\mathcal{S}}_1, \tilde{\mathcal{S}}_2, \dots, \tilde{\mathcal{S}}_{n-1}$ in their first variables, i.e., for all $j \in \{1, \dots, n-1\}$, we uncouple $\tilde{\mathcal{S}}_j$ in $\tilde{\mathcal{I}}_{j+1}$ and obtain the uncouplings

$$\left\{\tilde{\mathcal{I}}_{j+1}^{(H,n-j)}, \tilde{\mathcal{I}}_1^{(j+1,n-j)}, \tilde{\mathcal{I}}_2^{(j+1,n-j)}, \dots, \tilde{\mathcal{I}}_j^{(j+1,n-j)}, \tilde{\mathcal{I}}_{j+2}^{(j+1,n-j)}, \dots, \tilde{\mathcal{I}}_n^{(j+1,n-j)}\right\}. \tag{4.3.22}$$

Now, with all the new obtained relations, if we define the corrected uncoupling as

$$\tilde{\mathcal{U}}_Z^{(i,n)} := \left\{\tilde{\mathcal{I}}_i^{(H,n-i+1)}, \tilde{\mathcal{I}}_1^{(i,n)}, \tilde{\mathcal{I}}_2^{(i,n)}, \dots, \tilde{\mathcal{I}}_{i-1}^{(i,n)}, \tilde{\mathcal{I}}_{i+1}^{(i,n-i+1)}, \dots, \tilde{\mathcal{I}}_{n-1}^{(i,n-i+1)}, \tilde{\mathcal{I}}_n^{(i,n-i+1)}\right\}, \tag{4.3.23}$$

we simply replace the $\mathcal{U}$ uncouplings from (4.3.14) by their $\tilde{\mathcal{U}}$ counterpart, i.e., we do:

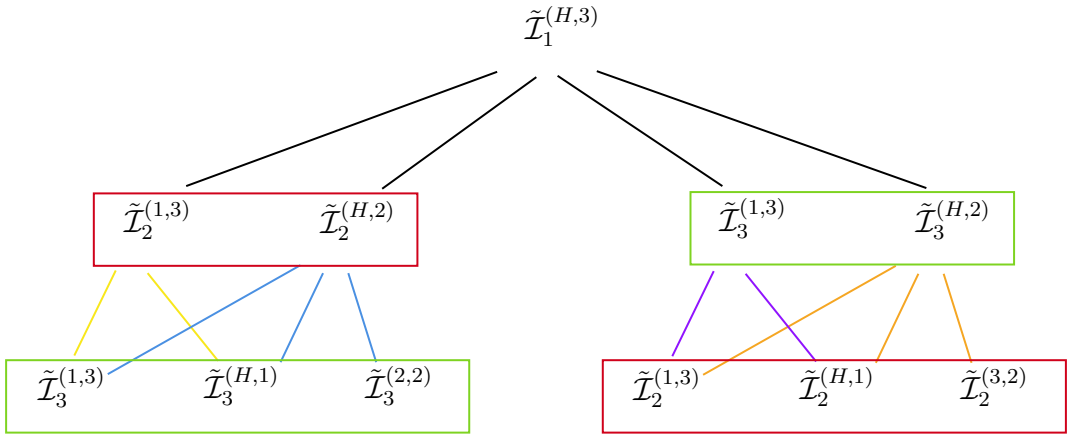
$$\left\{\mathcal{U}_Z^{(1,n)}, \mathcal{U}_Z^{(2,n)}, \dots, \mathcal{U}_Z^{(n-1,n)}, \mathcal{U}_Z^{(n,n)}\right\} \to \left\{\mathcal{U}_Z^{(1,n)}, \tilde{\mathcal{U}}_Z^{(2,n)}, \dots, \tilde{\mathcal{U}}_Z^{(n-1,n)}, \tilde{\mathcal{U}}_Z^{(n,n)}\right\}.$$

The selection tree now takes the following form, where we note in red the difference with the original tree:

$$
\begin{array}{lccccccccc}
R_1: & & & & \tilde{\mathcal{I}}_1^{(H,n)} & & & & & \\
R_2: & & & \tilde{\mathcal{I}}_2^{(H,n-1)} & & \tilde{\mathcal{I}}_2^{(1,n)} & & & & \\
R_3: & & \tilde{\mathcal{I}}_3^{(H,n-2)} & & \tilde{\mathcal{I}}_3^{(1,n)} & & \tilde{\mathcal{I}}_3^{(2,n-1)} & & & \\
 & \iddots & & \vdots & & \vdots & & \ddots & & \\
R_n: & \tilde{\mathcal{I}}_n^{(H,1)} & \tilde{\mathcal{I}}_n^{(1,n)} & \cdots & \cdots & \cdots & \cdots & \cdots & \tilde{\mathcal{I}}_n^{(n-2,3)} & \tilde{\mathcal{I}}_n^{(n-1,2)}
\end{array}
$$

4.3.3.2. Extending the tree of choices

We can improve the method from above as follows. Until now, we kept on choosing sequentially, e.g., first the best equation for $\tilde{\mathcal{I}}_2$, then $\tilde{\mathcal{I}}_3$, and so on, only one MI at a time. Nevertheless, we can do the following: after selecting the best HODE, say $\tilde{\mathcal{I}}_1^{(H,n)}$, instead of choosing the best equation for $\tilde{\mathcal{I}}_2$ ($\tilde{\mathcal{I}}_2^{(H,n-1)}$ or $\tilde{\mathcal{I}}_2^{(1,n)}$) we can choose among all those possible equations (HODEs and LFs) for all the remaining MIs, i.e., $\tilde{\mathcal{I}}_2, \ldots, \tilde{\mathcal{I}}_n$. Depending on the variable $\tilde{\mathcal{I}}_i$ chosen and whether it is a HODE or not, we eliminate it and proceed with $\tilde{\mathcal{I}}_2, \ldots, \tilde{\mathcal{I}}_{i-1}, \tilde{\mathcal{I}}_{i+1}, \ldots, \tilde{\mathcal{I}}_n$, etc. For example, for a system of order 3, this gives us the following tree of choices:



In order to write down the most general possible case for a system in n variables, where the first variable we choose is $\tilde{\mathcal{I}}_1$ (which we can always achieve by relabeling the variable), we have to define a new convention. We define the following notation:

Definition 4.3.2. *The set of equations accessible at the m^{th} step to obtain $\tilde{\mathcal{I}}_i$, using $\tilde{\mathcal{I}}_1^{(1,n)}$, the HODE of order $n-m$ in $\tilde{\mathcal{I}}_i$ and the LFs for $\tilde{\mathcal{I}}_{j_l}$, $l = 1, \ldots, p$, is denoted with $\mathcal{S}_{i;j_1,\ldots,j_p}^{(n,m)}$. More precisely, we have:*

$$\mathcal{S}_{i;j_1,\ldots,j_p}^{(n,m)} := \left\{\tilde{\mathcal{I}}_i^{(1,n)}, \tilde{\mathcal{I}}_i^{(H,n-m)}, \tilde{\mathcal{I}}_i^{(j_1,n-m+1)}, \ldots, \tilde{\mathcal{I}}_i^{(j_p,n-1)}\right\} \tag{4.3.24}$$

$$\mathcal{S}_{;}^{(n,0)} := \left\{\tilde{\mathcal{I}}_1^{(H,n)}\right\}, \quad \mathcal{S}_{i;}^{(n,1)} := \left\{\tilde{\mathcal{I}}_i^{(1,n)}, \tilde{\mathcal{I}}_i^{(H,n-1)}\right\} \tag{4.3.25}$$

E.g., using this notation, we can write the previous tree in the following very compact form:

$$\begin{array}{cc} & \mathcal{S}_{;}^{(3,0)} \\ \mathcal{S}_{2;}^{(3,1)} & \mathcal{S}_{3;}^{(3,1)} \\ \mathcal{S}_{3;2}^{(3,2)} & \mathcal{S}_{2;3}^{(3,2)} \end{array}$$

Let us formalise again the way we enhance the simple linear selection method:

3. For each of the best equations $\tilde{\mathcal{I}}_{\ell_p}^{(H,n)}$, we can relabel the relations so that

$$\tilde{\mathcal{I}}_{\ell_p}^{(H,n)} \longrightarrow \tilde{\mathcal{I}}_1^{(H,n)},$$

and then without loss of generality we consider the case where the best HODE is for $\tilde{\mathcal{I}}_1^{(H,n)}$ and we relabel the system accordingly. We define now the subsystems $\tilde{S}_{j;k}$, $j \in \{1,\dots,n-1\}$, $k \in \{0,\dots,n-j-1\}$ as

$$\tilde{S}_{j;k} : \frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{I}}^{(j;k)}(x,\epsilon) = \tilde{\mathcal{M}}^{(j;k)}(x,\epsilon)\cdot\tilde{\mathcal{I}}^{(j;k)}(x,\epsilon) + \tilde{\mathcal{R}}^{(j;k)}(x,\epsilon) \tag{4.3.26}$$

where

$$\tilde{\mathcal{I}}^{(j;k)}(x,\epsilon) = \left(\tilde{\mathcal{I}}_{j+k+1}(x,\epsilon),\dots,\tilde{\mathcal{I}}_n(x,\epsilon),\tilde{\mathcal{I}}_{j+1}(x,\epsilon),\dots,\tilde{\mathcal{I}}_{j+k}(x,\epsilon)\right)^{\intercal},$$

$$\tilde{\mathcal{I}}^{(j;0)}(x,\epsilon) = \tilde{\mathcal{I}}^{(j)}(x,\epsilon)$$

$$\tilde{\mathcal{R}}^{(j;k)}(x,\epsilon) = \left(\tilde{\mathcal{R}}_{j+k+1}(x,\epsilon),\dots,\tilde{\mathcal{R}}_n(x,\epsilon),\tilde{\mathcal{R}}_{j+1}(x,\epsilon),\dots,\tilde{\mathcal{R}}_{j+k}(x,\epsilon)\right)^{\intercal}.$$

$$\tilde{\mathcal{R}}^{(j;0)}(x,\epsilon) = \tilde{\mathcal{R}}^{(j)}(x,\epsilon)$$

$$\tilde{\mathcal{M}}^{(j;k)}(x,\epsilon) = \left(m_{p+\bar{k}+j,q+\bar{k}+j}\right)_{p,q\in\{1,\dots,n-j\}} \in \mathcal{M}_{(n-j)\times(n-j)}(\mathbb{K}(x,\epsilon)),$$

$$\tilde{\mathcal{M}}^{(j;0)}(x,\epsilon) = \tilde{\mathcal{M}}^{(j)}(x,\epsilon),$$

where $\bar{k} = k \mod (n-j)$. Finally the RHS subvector contains the $\tilde{\mathcal{J}}_i$, $i \in \{1,\dots,k\}$ and the $\tilde{\mathcal{I}}_i$, $i \in \{1,\dots,j\}$:

$$\forall i \in \{1,\dots,n\},\ \mathcal{R}_i^{(j;k)}(x,\epsilon) = \sum_{p=1}^{k}\varphi_{i+\bar{k},p}(x,\epsilon)\tilde{\mathcal{J}}_p(x,\epsilon) + \sum_{\ell=1}^{j}\bar{\varphi}_{i+\bar{k},\ell}(x,\epsilon)\tilde{\mathcal{I}}_\ell(x,\epsilon), \tag{4.3.27}$$

where

$$\varphi_{i,p}(x,\epsilon), \bar{\varphi}_{i,\ell}(x,\epsilon) \in \mathbb{K}(x,\epsilon) \text{ and } \bar{k} = k \mod (n-j).$$

In summary, the notation above simply stands for circular permutations of the system $\tilde{\mathcal{S}}_j$. For all $k \in \{0,\ldots,n-j-1\}$, we uncouple $\tilde{\mathcal{S}}_{1;k}, \tilde{\mathcal{S}}_{2;k}, \ldots, \tilde{\mathcal{S}}_{n-1;k}$ in their first variable, i.e., for all $j \in \{1,\ldots,n-1\}$ we uncouple $\tilde{\mathcal{S}}_{j;k}$ in $\tilde{\mathcal{I}}_{j+k+1}$ and with the new obtained relations, we can define the corrected uncouplings as:

$$\tilde{\mathcal{U}}_Z^{(i,n;k)} := \left\{\tilde{\mathcal{I}}_{i+k}^{(H,n-i+1)}, \tilde{\mathcal{I}}_1^{(i,n)}, \tilde{\mathcal{I}}_2^{(i,n)}, \ldots, \tilde{\mathcal{I}}_{i+k-1}^{(i,n)}, \tilde{\mathcal{I}}_{i+k+1}^{(i,n-i+1)}, \ldots, \tilde{\mathcal{I}}_{n-1}^{(i,n-i+1)}, \tilde{\mathcal{I}}_n^{(i,n-i+1)}\right\}, \tag{4.3.28}$$

Therefore, we generate not only the usual HODE order correction for the order
$(\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \ldots, \tilde{\mathcal{I}}_n)$, i.e.,

$$\left\{\mathcal{U}_Z^{(1,n)}, \tilde{\mathcal{U}}_Z^{(2,n)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n)}, \tilde{\mathcal{U}}_Z^{(n,n)}\right\}$$

but also for the remaining $n-2$ cyclic permutations

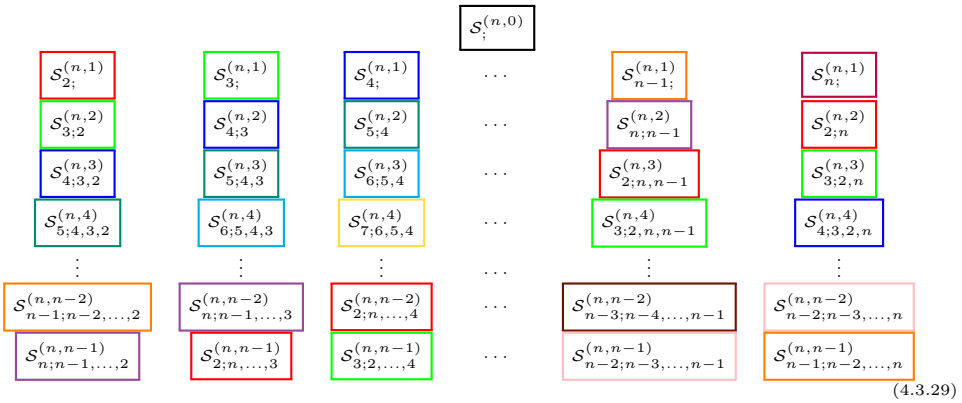
$$\left(\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_3, \tilde{\mathcal{I}}_4, \ldots, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_2\right); \left(\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5, \ldots, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3\right); \cdots; \left(\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_2, \ldots, \tilde{\mathcal{I}}_{n-2}, \tilde{\mathcal{I}}_{n-1}\right).$$

This means in particular that the full correction to the initial uncoupling is extended to

$$\begin{aligned}\left\{\mathcal{U}_Z^{(1,n)}, \mathcal{U}_Z^{(2,n)}, \ldots, \mathcal{U}_Z^{(n-1,n)}, \mathcal{U}_Z^{(n,n)}\right\} \to \\ \left\{\mathcal{U}_Z^{(1,n)}, \tilde{\mathcal{U}}_Z^{(2,n)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n)}, \tilde{\mathcal{U}}_Z^{(n,n)},\right. \\ \tilde{\mathcal{U}}_Z^{(2,n;1)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;1)}, \tilde{\mathcal{U}}_Z^{(n,n;1)}, \\ \tilde{\mathcal{U}}_Z^{(2,n;2)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;2)}, \tilde{\mathcal{U}}_Z^{(n,n;2)}, \\ \vdots \\ \left.\tilde{\mathcal{U}}_Z^{(2,n;n-1)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;n-1)}, \tilde{\mathcal{U}}_Z^{(n,n;n-1)}\right\}.\end{aligned}$$

Using the Definition 4.3.2, the selection tree generalizes to the following

compact form:



(4.3.29)

The list of best uncouplings $\tilde{\mathcal{B}}$ contains at the moment one element:

$$\tilde{\mathcal{B}} := \left(\left(\tilde{\mathcal{I}}_1^{(H,n)}\right)\right). \tag{4.3.30}$$

We define also another set, that we call the HODE set:

$$\tilde{\mathcal{H}} := \left(\tilde{\mathcal{I}}_1^{(H,n)}\right).$$

Now each row R_i, $i \in \{2,\dots,n\}$, becomes of the form:

$$R_i = \bigcup_{j=0}^{n-2} \mathcal{S}^{(n,i-1)}_{f(i+j);f(i+j-1),f(i+j-2),\dots,f(j)},$$

where

$$f(p) := \begin{cases} p \mod n & \text{if } p < n \\ n & \text{if } p = n \\ (p \mod n) + 1 & \text{if } p > n \end{cases}.$$

4. If we map the function $\tilde{\mathcal{O}}$ to all the elements of each row R_i which are of the form $\tilde{\mathcal{I}}_k^{(i,j)}$, we obtain a vector of ϵ-orders that we call $\bar{R}_i$ as in the Section 4.3.2.1. We apply now the equation selection procedure $\mathcal{F}$ on every row (or layer) $\bar{R}_i$, $i \in \{2,\dots,n\}$, to select the best path. At each layer j, computing $\mathcal{F}(R_j)$ will yield $q \geq 1$ possible optimal solutions $S = \{\tilde{\mathcal{I}}_j^{(i_1,l_1)},\dots,\tilde{\mathcal{I}}_j^{(i_q,l_q)}\}$. Given that after the step $j-1$, $\tilde{\mathcal{B}}$ is of the form

$$\tilde{\mathcal{B}} = (\dots,(\tilde{\mathcal{I}}_1^{(H,n)},\tilde{\mathcal{I}}_2^{(k_p^2,t_p^2)},\dots,\tilde{\mathcal{I}}_{j-1}^{(k_p^{j-1},t_p^{j-1})}),\dots)$$

for $p \in \mathbb{N}^\star$ and $k_p^r, t_p^s \leq n$, we then enlarge $\tilde{\mathcal{B}}$ in the following way:

$$\tilde{\mathcal{B}} :\leftarrow (\ldots, (\tilde{\mathcal{I}}_1^{(H,n)}, \ldots, \tilde{\mathcal{I}}_{j-1}^{(k_p^{j-1}, t_p^{j-1})}, \tilde{\mathcal{I}}_j^{(i_1,l_1)}), \ldots, (\tilde{\mathcal{I}}_1^{(H,n)}, \ldots, \tilde{\mathcal{I}}_{j-1}^{(i_p^{j-1}, t_p^{j-1})}, \tilde{\mathcal{I}}_j^{(i_q,l_q)}) \ldots)$$

As we carry out the selection above, at the same time, for each row R_i we also do the following for each of the optimal solutions:

- If the optimal equation is a HODE of order $q \in \{1, \ldots, n-1\}$, e.g. $\tilde{\mathcal{I}}_j^{(H,q)}$, then we add it as well as its dependent LFs to $\tilde{\mathcal{H}}$:

  $$\tilde{\mathcal{H}} :\leftarrow \tilde{\mathcal{H}} \cup \{\tilde{\mathcal{I}}_j^{(H,q)}\}$$

  and then erase all the other HODEs of order j — as neither them nor their LFs will be used further — as well as all the LFs to compute $\tilde{\mathcal{I}}_j$ from the selection tree.
- If the optimal equation is a LF, then we erase all the other HODEs of order j — as neither them nor their LFs will be used further — as well as all the LFs to compute $\tilde{\mathcal{I}}_j$ from the selection tree. In short, we do the same as in the other cas except for adding the $\tilde{\mathcal{I}}_j^{(H,n-1)}$ to $\tilde{\mathcal{H}}$.

After iterating this on all the rows, $\tilde{\mathcal{B}}$ is the list of all the possible candidates for a preferable set of equations.

4.3.3.3. Adding Gauß uncouplings

One last possible extension for the choices is to use another uncoupling scheme, in particular the Gauß uncoupling scheme [116]. Different to the Zürcher scheme, Gauß uncoupling has a different structure that does not produce only one HODE and $n-1$ LFs but n HODEs of different orders. The structure is as follows:

Gauß output:

$$\begin{cases} \text{HODE}\left(\{\underline{\tilde{\mathcal{I}}_{i_1}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_1}}\}; \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}\right) \\ \text{HODE}\left(\{\underline{\tilde{\mathcal{I}}_{i_2}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_2}}\}; \{\tilde{\mathcal{I}}_{i_1}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_1}\}, \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}\right) \\ \qquad\qquad \vdots \\ \text{HODE}\left(\{\underline{\tilde{\mathcal{I}}_{i_n}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_n}}\}; \{\tilde{\mathcal{I}}_{i_{n-1}}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_{n-1}}\}, \cdots, \{\tilde{\mathcal{I}}_{i_1}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_1}\}, \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}\right) \end{cases} \tag{4.3.31}$$

where for all $q \in \{1, \dots, n\}$

$$\text{HODE}\left(\{\underline{\tilde{\mathcal{I}}_{i_q}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_q}}\}; \{\tilde{\mathcal{I}}_{i_{q-1}}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_{q-1}}\}, \cdots, \{\tilde{\mathcal{I}}_{i_1}, \mathrm{d}_x^j \tilde{\mathcal{I}}_{i_1}\}, \{\tilde{\mathcal{R}}_i, \mathrm{d}_x^j \tilde{\mathcal{R}}_i\}\right)$$

means a inhomogeneous differential equation in $\tilde{\mathcal{I}}_{i_q}$ that depends on the $\tilde{\mathcal{I}}_{i_p}$ and their derivatives $\mathrm{d}_x^j \tilde{\mathcal{I}}_{i_p}$, $p \in \{1, \dots, j\}$, as well as on the $\tilde{\mathcal{R}}_i$ and their derivatives $\mathrm{d}_x^j \tilde{\mathcal{R}}_{i_p}$.

Remark 4.3.5. *Note that Gauß uncoupling is more sensible to the order of the equations, in contrast to Zürcher's method where the important element is the variable in which we uncouple. Therefore, to obtain the best choice, one has to uncouple in all the possible ways which yields $n!$ uncouplings for a system of size n. In practice though, one has to put a time limit because otherwise this single step will already take too much time to be useful.*

In order to incorporate the Gauß uncoupling scheme in the linear selection method, we do the following:

1. Following Remark 4.3.5, one performs as many Gauss uncouplings as possible. We obtain $q \leq n!$ sets of equations of the form (4.3.31).
2. We compare the uncouplings. If some of them are identical, which may happen, we delete the duplicates.
3. Given that the Gauß uncouplings have a much more rigid structure — due to the dependences, we do not split the uncoupling sets given in (4.3.31) and will simply add them to the $\tilde{\mathcal{B}}$ set defined for example in (4.3.30), at the end of the step 4.

Remark 4.3.6. *The Gauß uncoupling scheme is interesting as sometimes it produces better ϵ-order results. At the same time, it also serves as a safeguard for the situation where in the Zürcher uncoupling we cannot eliminate the dummy variable W_1. As long as we will perform at least one Gauß uncoupling, such exotic cases containing the dummy variables cannot arise.*

Remark 4.3.7. *In Remark 4.3.3, we explained that in some situations we have a slight bottleneck in the Zürcher scheme. It stems from the fact that sometimes there might not be a a way to get rid of the W variables because the first relation in [3] on page 196 is*

$$\tilde{\mathcal{I}}_1 = G_1(\{W_i, \mathrm{d}_x^j W_i\}),$$

with G_1 not being the identity, i.e., $\tilde{\mathcal{I}}_1 = W_1$, which would allow to fully eliminate all the W_i variables. This entails the fact that we cannot avoid introducing W_1 as

an extra MI and therefore having an extra HODE that we are forced to solve to get it. Because now we know that increasing the number of HODEs usually makes the system difficult to solve, as seen in Section 4.3.4, we decided to avoid such configurations.

4.3.4. Recurrences as selection scheme

Throughout this section we adopted the differential equation paradigm, that is to say we uncoupled a differential system into HODE and LFs and then we only compared the ϵ-orders appearing in those to select an "optimal" equation. However, there is a possible alternative — while minimising the ϵ-orders is a valid criterion, one should not neglect two other elements: the order of the HODEs, as well as the order of the associated recurrences. For the former case, it is rather clear that if we have fewer terms in the ϵ expansion to give but at the same time much more differential equations with higher orders to solve, the trade off might not be interesting. For the latter case, remember from the Step 3.b from Section 3.4.3 that whenever we have a differential equation of the form

$$\alpha_0(x,\epsilon)\tilde{\mathcal{I}}_1(x,\epsilon) + \cdots + \alpha_m(x,\epsilon)\mathrm{d}_x^m\tilde{\mathcal{I}}_1(x,\epsilon) = \beta(x,\epsilon), \tag{4.3.32}$$

one can obtain, using the formal series ansatz and coefficient comparison[112], an associated difference equation of the form[113]

$$a_0(n,\epsilon)I_1(n,\epsilon) + a_1(n,\epsilon)I_1(n+1,\epsilon) + \cdots + a_d(n,\epsilon)I_1(n+d,\epsilon) = b(n,\epsilon). \tag{4.3.33}$$

Note that the order m of the DE in (4.3.32) equals exactly the dimension of the coupled system, i.e., the number of unknowns. But the order d (4.3.33) depends on the degree of the polynomials $a_i(x,\epsilon)$ with respect to λ.
Thus it might happen that while the differential equation we obtain is of relatively small order, the order of the recurrence might become really high. Therefore, a configuration where the ϵ-orders and the HODEs orders are small but the result is still not completley optimal because the recurrences are big and might take a lot of time to solve. In particular if the order is higher, e.g., of order 100, it might be difficult to obtain the required initial values from the physical problem. Therefore, one can extend the procedure $\mathcal{F}$ to not only compute the cumulative ϵ-orders $\mathcal{O}\left(\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}\right)$ but also the HODE and recurrence orders. We call this extended procedure $\bar{\mathcal{F}}$ for an equation $\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}$:

[112] See also [49].
[113] In `HarmonicSums`, the function that does this operation is `DEToRE`.

Equation selection procedure $\bar{\mathcal{F}}$

- ϵ-order: Compute $n_\epsilon := \mathcal{F}\left(\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}\right)$.
- Differential equaton order: if $\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}$ is a HODE then take n_d; = order $\left(\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}\right)$ and if $\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}$ is a LF then take $n_d := 0$.
- Recurrence order: If $\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}$ is a HODE, using the formal power series ansatz and coefficient comparison, compute the recurrence $\bar{\tilde{\mathcal{I}}}_{j_\ell}^{(i_\ell)}$ associated to $\tilde{\mathcal{I}}_{j_\ell}^{(i_\ell)}$ and set $n_r :=$ order $\left(\bar{\tilde{\mathcal{I}}}_{j_\ell}^{(i_\ell)}\right)$. If $\bar{\tilde{\mathcal{I}}}_{j_\ell}^{(i_\ell)}$ is a LF then set $n_r := 0$.
- Return the triplet $\{-n_\epsilon, n_d, n_r\}$.

As opposed to the original $\mathcal{F}$ procedure, we have to minimise not only an integer but a triplet of integers. There does not exist a unique optimal scalar function for that but one can take the simplest possible "equal weight" form which would be

$$f_1(\{-n_\epsilon, n_d, n_r\}) := -n_\epsilon + n_d + n_r. \tag{4.3.34}$$

In the implementation, one could also consider that depending on the solving schemes we can also have the following weight functions:

- Only ϵ-order: $f_2(\{-n_\epsilon, n_d, n_r\}) := -n_\epsilon$
- Only HODE order: $f_3(\{-n_\epsilon, n_d, n_r\}) := n_d$
- Only recurence order: $f_4(\{-n_\epsilon, n_d, n_r\}) := n_r$

We include also the possibility for the user to also define themselves the weight function f and give it as an optional argument to the main function.

We remark that this refinement plays also an important role for the large moment method[114] to derive large moments using the recurrences as given in (4.3.33).

4.4. Computing the corrections

After having clustered the whole system in Section 4.1, we have described in Section 4.1 a method to obtain an optimal set of equations to solve, using relations obtained from different Zürcher uncouplings, selected in such a way that we minimise the ϵ-orders, combined with Gauß uncouplings. What remains is to solve the system up to the user-required order in the MIs:

[114] See Section 3.4.6.

- Compute for each subsystem the order up to which we need to give the BIs.
- Using these results, compute the absolute ϵ-order correction for each BI in a recursive way for the whole system[115] such that the MIs can be sufficiently high expanded.

In other words, we need to compute the relative ϵ-order corrections locally for each subsystem so that we can compute the absolute ϵ-order corrections in the full system. In order to understand how we extract the different $\epsilon-$order corrections, let us consider a toy example:

Example 4.4.1. *Let us take a system of order $n = 4$ with 4 MIs $\tilde{\mathcal{I}}_{i=1,\dots,4}$ and 2 BIs $\tilde{\mathcal{J}}_{i=1,2}$. After the procedure described in Section 4.3, we assume that we are given the following set of optimal equations*[116]*:*

$$\begin{cases} \text{HODE}(\tilde{\mathcal{I}}_1) &= \epsilon^{\alpha_1}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_1}\tilde{\mathcal{J}}_2 \\ \text{HODE}(\tilde{\mathcal{I}}_4) &= \epsilon^{\alpha_2}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_2}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_2}\tilde{\mathcal{I}}_1 \\ \tilde{\mathcal{I}}_3 &= \epsilon^{\alpha_3}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_3}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_3}\tilde{\mathcal{I}}_1 + \epsilon^{\delta_3}\tilde{\mathcal{I}}_4 \\ \tilde{\mathcal{I}}_2 &= \epsilon^{\alpha_4}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_4}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_4}\tilde{\mathcal{I}}_1 + \epsilon^{\delta_4}\tilde{\mathcal{I}}_4 + \epsilon^{\eta_4}\tilde{\mathcal{I}}_3 \end{cases} . \tag{4.4.1}$$

As we can see, due to the selection procedure where we decrease the degree of the uncouplings by considering previous MIs as BIs, we obtain a triangular dependence — the HODE in $\tilde{\mathcal{I}}_4$ can be solved only once the HODE in $\tilde{\mathcal{I}}_1$ is solved, then to obtain $\tilde{\mathcal{I}}_3$ we have to plug in $\tilde{\mathcal{I}}_1$ and $\tilde{\mathcal{I}}_4$, etc.[117]*. What we aim to do is to know up to which order in ϵ we should give $\tilde{\mathcal{J}}_1$, $\tilde{\mathcal{J}}_2$ in order to be able to compute the MIs up to a given order. In full generality, we write the following initialisation:*

$$(\tilde{\mathcal{I}}_1, a), (\tilde{\mathcal{I}}_2, b), (\tilde{\mathcal{I}}_3, c), (\tilde{\mathcal{I}}_4, d) \tag{4.4.2}$$

This means we suppose we want to compute $\tilde{\mathcal{I}}_1$ up to (symbolic and absolute) order a, $\tilde{\mathcal{I}}_2$ up to b, $\tilde{\mathcal{I}}_3$ up to c and $\tilde{\mathcal{I}}_4$ up to d. If we look now at the first equation from (4.4.3), in order to be able to reach a for $\tilde{\mathcal{I}}_1$, we need $\tilde{\mathcal{J}}_1$ to be expanded up to order $a - \alpha_1$: if $\alpha_1 > 0$, this would mean that we only need fewer orders, and if $\alpha_1 < 0$, like in Section 4.3.1, we will need to go higher than a. We need to improve this a bit: indeed, if $a > a - \alpha_1$, we still want to get $\tilde{\mathcal{I}}_1$ up to order a so actually we need to go up to a, so we can simply take $\max(a, a - \alpha_1)$. But now when we look at the second equation from (4.4.3), we realise that because $\tilde{\mathcal{I}}_1$ appears there, in order to be able to get $\tilde{\mathcal{I}}_4$ up to the order d, we need to have $\tilde{\mathcal{I}}_1$ up to order

[115] Where the words "relative" and "absolute" are used in the sense of Remark 4.3.1.

[116] For simplicity, we fully remove the dependence in x, and consider only the ϵ part in front of each of the MIs or BIs appearing in the RHS of the HODEs or LFs relations

[117] We note again that due to the selection procedure from Section 4.3, the system is naturally sorted and equations can be solved from top to bottom.

$\max(a, d, d-\gamma_2)$. *However this means in the first equation this was also the order up to which we required* $\tilde{\mathcal{I}}_1$*, which in turn means that the actual order up to which we need to give* $\tilde{\mathcal{J}}_1$ *is* $\max(\max(a, d, d-\gamma_2), \max(a, d, d-\gamma_2)-\alpha_1)$*. Based on that we observe that there is a need for retroactive correction, and therefore we have to* ***update recursively*** *the* ϵ*-orders, from the last to the first equation. If* A_i *is a letter denoting the order we required for a given MI at the* i^{th} *step, we note with* A_{i+1} *the updated order at the* $(i+1)^{\text{th}}$ *step. We do this explicitely for* $\tilde{\mathcal{J}}_1$ *in this example:*

1. *Updating with the* 4^{th} *equation* ($\tilde{\mathcal{I}}_2 = \epsilon^{\alpha_4}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_4}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_4}\tilde{\mathcal{I}}_1 + \epsilon^{\delta_4}\tilde{\mathcal{I}}_4 + \epsilon^{\eta_4}\tilde{\mathcal{I}}_3$) *gives*

$$\begin{aligned} &(\tilde{\mathcal{I}}_1, a_1 = \max(a, b-\gamma_4)) \qquad (\tilde{\mathcal{I}}_2, b_1 = \boxed{b} - \alpha_4)\\ &(\tilde{\mathcal{I}}_4, d_1 = \max(d, b-\delta_4))\\ &(\tilde{\mathcal{I}}_3, c_1 = \max(c, b-\eta_4)). \end{aligned}$$

   The left column of corrections represents the retroactive corrections we need to do to the MIs orders ($\gamma_4, \delta_4, \eta_4$ *are powers of* ϵ *appearing in front of the MIs) while the right one is representing the correction to* $\tilde{\mathcal{J}}_1$ *itself* (α_4 *is the* ϵ*-power appearing in front of the* $\tilde{\mathcal{J}}_1$*). The boxed number represents the required correction to the MI appearing on the RHS, here* $\tilde{\mathcal{I}}_2$ *— in this case, given it is the last equation and* $\tilde{\mathcal{I}}_2$ *does not appear in any equation above the original order* b *is already correct. The obtained corrected orders at this point are:*

$$(\tilde{\mathcal{I}}_1, a_1),\ (\tilde{\mathcal{I}}_2, b_1),\ (\tilde{\mathcal{I}}_3, c_1),\ (\tilde{\mathcal{I}}_4, d_1)$$

2. *Updating with the* 3^{rd} *equation* ($\tilde{\mathcal{I}}_3 = \epsilon^{\alpha_3}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_3}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_3}\tilde{\mathcal{I}}_1 + \epsilon^{\delta_3}\tilde{\mathcal{I}}_4$) *gives*

$$\begin{aligned} &(\tilde{\mathcal{I}}_1, a_2 = \max(a_1, c_1-\gamma_3)) \qquad (\tilde{\mathcal{I}}_3, c_2 = \boxed{c_1} - \alpha_3)\\ &(\tilde{\mathcal{I}}_4, d_2 = \max(d_1, c_1-\delta_3)). \end{aligned}$$

   Here we see that because $\tilde{\mathcal{I}}_3$ *appeared in the LF for* $\tilde{\mathcal{I}}_2$ *and obtained a correction* $c \to c_1$*, we take* c_1 *as the corrected* $\tilde{\mathcal{I}}_3$ $\epsilon-$*order. The obtained corrected orders at this point are:*

$$(\tilde{\mathcal{I}}_1, a_2),\ (\tilde{\mathcal{I}}_2, b_1),\ (\tilde{\mathcal{I}}_3, c_2),\ (\tilde{\mathcal{I}}_4, d_2).$$

3. *Updating with the* 2^{nd} *equation* ($\text{HODE}(\tilde{\mathcal{I}}_4) = \epsilon^{\alpha_2}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_2}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_2}\tilde{\mathcal{I}}_1$) *gives*

$$(\tilde{\mathcal{I}}_1, a_3 = \max(a_2, d_2-\gamma_2)) \qquad (\tilde{\mathcal{I}}_4, d_3 = \boxed{d_2} - \alpha_2).$$

   The obtained corrected orders at this point are:

$$(\tilde{\mathcal{I}}_1, a_3),\ (\tilde{\mathcal{I}}_2, b_1),\ (\tilde{\mathcal{I}}_3, c_2),\ (\tilde{\mathcal{I}}_4, d_3).$$

4. *Updating with the* 1st *equation* ($\mathrm{HODE}(\tilde{\mathcal{I}}_1) = \epsilon^{\alpha_1}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_1}\tilde{\mathcal{J}}_2$) *yields*

$$\setminus \qquad\qquad (\tilde{\mathcal{I}}_1, a_4 = \boxed{a_3} - \alpha_1).$$

The obtained corrected orders at this point are:

$$(\tilde{\mathcal{I}}_1, a_4),\ (\tilde{\mathcal{I}}_2, b_1),\ (\tilde{\mathcal{I}}_3, c_2),\ (\tilde{\mathcal{I}}_4, d_3).$$

The full correction for $\tilde{\mathcal{J}}_1$ *is therefore:*

$$\max(a_4, b_1, c_2, d_3).$$

Moreover, the corrected MI $\epsilon-$*orders are:*

$$(\tilde{\mathcal{I}}_1, a_3),\ (\tilde{\mathcal{I}}_2, b),\ (\tilde{\mathcal{I}}_3, c_1),\ (\tilde{\mathcal{I}}_4, d_2).$$

Remark 4.4.1. *The example above generalizes very easily if there are derivatives of the MIs/BIs, e.g., if the second equation of the example* (4.4.3) *was:*

$$\mathrm{HODE}(\tilde{\mathcal{I}}_4) = \epsilon^{\alpha_2}\tilde{\mathcal{J}}_1 + \epsilon^{\beta_2}\tilde{\mathcal{J}}_2 + \epsilon^{\gamma_2}\tilde{\mathcal{I}}_1 + \epsilon^{\gamma_2^{(1)}}\mathrm{d}_x\tilde{\mathcal{I}}_1.$$

Because we have now $\tilde{\mathcal{I}}_1$ *and its derivative, we will need to satisfy both of these conditions. Therefore at step 3 the correction will be:*

$$(\tilde{\mathcal{I}}_1, a_3 = \max(a_2, d_2 - \gamma_2, d_2 - \gamma_2^{(1)}).$$

4.4.1. ϵ-order correction

Let us now formalise the procedure described above.

Input: Let $n \in \mathbb{N}$ be the size of the subsystem, containing $p \in \mathbb{N}^\star$ MIs $\tilde{\mathcal{I}}_i$ and $q \in \mathbb{N}^\star$ BIs $\tilde{\mathcal{J}}_j$. Let σ be a permutation of $(1, \ldots, n)$. Furthermore, we are given

the following set of optimal equations:

$$\begin{cases} \tilde{\mathcal{I}}^{(t_1,n)}_{\sigma(1)} &= \sum_{u=0}^{U^{(t_1,\sigma(1))}} \sum_{v=1}^{q} \epsilon^{m^{(t_1,\sigma(1))}_{u,v}} \frac{\alpha^{(t_1,\sigma(1))}_{v,u}}{\beta^{(t_1,\sigma(1))}_{v,u}} \mathrm{d}^u_x \tilde{\mathcal{J}}_v \\ \tilde{\mathcal{I}}^{(t_2,m_1)}_{\sigma(2)} &= \sum_{w=0}^{W^{(t_2,\sigma(2))}} \epsilon^{m^{(t_2,\sigma(2))}_{w}} \frac{\alpha^{(t_2,\sigma(2))}_{w}}{\beta^{(t_2,\sigma(2))}_{w}} \mathrm{d}^w_x \tilde{\mathcal{I}}_{\sigma(1)} \\ &\quad + \sum_{u=0}^{U^{(t_2,\sigma(2))}} \sum_{v=1}^{q} \epsilon^{m^{(t_2,\sigma(2))}_{u,v}} \frac{\alpha^{(t_2,\sigma(2))}_{v,u}}{\beta^{(t_2,\sigma(2))}_{v,u}} \mathrm{d}^u_x \tilde{\mathcal{J}}_v \\ \tilde{\mathcal{I}}^{(t_3,m_2)}_{\sigma(3)} &= \sum_{i=1}^{2} \sum_{w=0}^{W^{(t_3,\sigma(3))}} \epsilon^{m^{(t_3,\sigma(3))}_{w}} \frac{\alpha^{(t_3,\sigma(3))}_{w}}{\beta^{(t_3,\sigma(3))}_{w}} \mathrm{d}^w_x \tilde{\mathcal{I}}_{\sigma(i)} \\ &\quad + \sum_{u=0}^{U^{(t_3,\sigma(3))}} \sum_{v=1}^{q} \epsilon^{m^{(t_3,\sigma(3))}_{u,v}} \frac{\alpha^{(t_3,\sigma(3))}_{v,u}}{\beta^{(t_3,\sigma(3))}_{v,u}} \mathrm{d}^u_x \tilde{\mathcal{J}}_v \\ \vdots \quad & \vdots \qquad\qquad \vdots \\ \tilde{\mathcal{I}}^{(t_n,m_{n-1})}_{\sigma(n)} &= \sum_{i=1}^{n} \sum_{w=0}^{W^{(t_n,\sigma(n))}} \epsilon^{m^{(t_n,\sigma(n))}_{w}} \frac{\alpha^{(t_n,\sigma(n))}_{w}}{\beta^{(t_n,\sigma(n))}_{w}} \mathrm{d}^w_x \tilde{\mathcal{I}}_{\sigma(i)} \\ &\quad + \sum_{u=0}^{U^{(t_n,\sigma(n))}} \sum_{v=1}^{q} \epsilon^{m^{(t_n,\sigma(n))}_{u,v}} \frac{\alpha^{(t_n,\sigma(n))}_{v,u}}{\beta^{(t_n,\sigma(n))}_{v,u}} \mathrm{d}^u_x \tilde{\mathcal{J}}_v \end{cases} \tag{4.4.3}$$

where we have $m_i \in \{0, \dots, n-1\}$, $t_i \in \{1, \dots, n\} \cup \{H\}$, $U^{(t_i,\sigma(i))}, W^{(t_i,\sigma(i))} \in \mathbb{N}$, $m^{t_i,\sigma(i)}_{u,v}, m^{t_i,\sigma(i)}_{w} \in \mathbb{N}$ and finally $\alpha^{(t_i,\sigma(i))}_{u,v}, \alpha^{(t_i,\sigma(i))}_{w}, \beta^{(t_i,\sigma(i))}_{u,v}, \beta^{(t_i,\sigma(i))}_{w} \in \mathbb{K}[x, \epsilon]$ such that ϵ does not divide any of the α, β polynomials.
Output: Tuples $\left(\tilde{\mathcal{J}}_i, C_i\right)$ and $\left(\tilde{\mathcal{I}}_i, \bar{C}_i\right)$, where C_i and $\bar{C}_i$ are the ϵ-corrections of the $\tilde{\mathcal{J}}_i, \tilde{\mathcal{I}}_i$ respectively

Algorithm: We first write the initialisation:

$$(\tilde{\mathcal{I}}_{\sigma(1)}, a_1),\ (\tilde{\mathcal{I}}_{\sigma(2)}, a_2),\ (\tilde{\mathcal{I}}_{\sigma(3)}, a_3),\ \dots,\ (\tilde{\mathcal{I}}_{\sigma(n)}, a_n). \tag{4.4.4}$$

for variables $a_1, \dots, a_n$. Later these variables will be specialised to concrete values[a]. Now for each of the $\tilde{\mathcal{J}}_i$ we perform recursively the correction of the orders, starting with the n^{th} equation until the first one, such that at each step $k \in \{1, \dots, n\}$ we define:

- For each of the $\tilde{\mathcal{I}}_{\sigma(i)}$, $i \in \{1, \dots, k-1\}$:

  $$a_{i,n-k+1} := \max(a_{i,n-k}, a_{k,n-k} - m^{t_k,\sigma(k)}_0, a_{k,n-k} - m^{t_k,\sigma(k)}_1, \dots, a_{k,n-k} - m^{t_k,\sigma(k)}_{W^{(t_k,\sigma(k))}})$$

  where $a_{i,0} = a_i$.

- For $\tilde{\mathcal{I}}_{\sigma(k)}$ itself, we define the correction for each of the $\tilde{\mathcal{J}}_l$, $l \in \{1, \dots, q\}$:

  $$a^l_{k,n-k} := \max(a_{k,n-k-1} - \epsilon^{(t_k,\sigma(k))}_{0,l}, a_{k,n-k-1} - \epsilon^{(t_k,\sigma(k))}_{1,l}, \dots, a_{k,n-k-1} - \epsilon^{(t_k,\sigma(k))}_{U^{t_k,\sigma(k)},l}).$$

After n steps, the final correction for each of the $\tilde{\mathcal{J}}_l$, $l \in \{1, \ldots, q\}$ becomes:

$$C_l(a_1, \ldots, a_n) := \max_{1 \leq k \leq n} \{a^l_{k,n-k}\}.$$

The correction for the $\tilde{\mathcal{I}}_{\sigma(i)}$, $i \in \{1, \ldots, p\}$ is:

$$\bar{C}_i(a_1, \ldots, a_n) := a_{i,n-i-1}.$$

[a] See also Remark 4.4.3.

Remark 4.4.2. *We note here that the final correction C_l and $\bar{C}_l$ are expressions in the variables a_i that can be considered as functions in the unknown required orders a_i. As a consequence we do not have to redo calculations if the desired orders change, but simply calculate the values by evaluating the functions.*

Remark 4.4.3. *Technically speaking, to implement this "corrections as function" in Mathematica we use "function-rules". More precisely, when performing the operation described in the algorithm above, we return a rule of the form*

$$\{\{\tilde{\mathcal{I}}_{\sigma(1)}, a_{1_}\}, \ldots, \{\tilde{\mathcal{I}}_{\sigma(n)}, a_{n_}\}\} \to \{\{\{\tilde{\mathcal{I}}_{\sigma(1)}, \bar{C}_1\}, \ldots, \{\tilde{\mathcal{I}}_{\sigma(n)}, \bar{C}_n\}\}, \{\{\{\tilde{\mathcal{J}}_1, C_1\}, \ldots, \{\tilde{\mathcal{J}}_q, C_q\}\}\}\}.$$

For example, for the following toy system

$$\begin{aligned} HODE(\tilde{\mathcal{I}}_1) &= -2\epsilon^8(1 - 2\epsilon^2)\mathcal{J} + \epsilon^{27}\mathcal{J}' \\ \tilde{\mathcal{I}}_2 &= -\frac{\epsilon^{14}}{2}\tilde{\mathcal{I}}_1 + \frac{\epsilon^{19}}{2}\tilde{\mathcal{I}}_1' - 3\epsilon^{19}\mathcal{J} \end{aligned}$$

after computation of the correction the rule would be:

$$\begin{aligned} &\{\{\tilde{\mathcal{I}}_1, a_\}, \{\tilde{\mathcal{I}}_2, b_\}\} \\ &\quad \longrightarrow \{\{\{\tilde{\mathcal{I}}_1, \max(a, b-14)\}, \{\tilde{\mathcal{I}}_2, b\}\}, \{\{\{\mathcal{J}, \max\left[\max(a, b-14) - 8, b-19\right]\}\}\}. \end{aligned}$$

While the underscore _ appearing for the symbolic orders a and b on the left hand side is necessary for Mathematica, we will often omit it here for simplicity.

4.4.2. Equation selection method and correction computing

In Section 4.3.2.1 we have presented the simple linear selection method and in the associated subsections we describe several improved schemes. Nevertheless, up to now they all relied on the possible equations we could select, but when given a certain set of equations they all consistently used the equation selection procedure $\mathcal{F}$. Now with Section 4.4.1 we can actually either follow the usual path and use simply $\mathcal{F}$ or apply explicitely the ϵ-order correction procedure together with $\mathcal{F}$.

For that we need to correct some steps as $\mathcal{F}$ is working "horizontally" — i.e., we compare equations for a collection of MIs $\tilde{\mathcal{I}}_i$ at a given step — whereas the ϵ-order correction procedure is working "vertically" — i.e. we are given a set of equations for all the MIs and then we compute the correction for them. Let us describe in this subsection this combined approach.

Input: An irreducible differential system $\mathcal{S}$ of order n with k BIs $\tilde{\mathcal{J}}_i$:

$$\frac{\mathrm{d}}{\mathrm{d}x}\tilde{\mathcal{I}}(x,\epsilon) = \mathcal{M}(x,\epsilon)\cdot\tilde{\mathcal{I}}(x,\epsilon) + \tilde{\mathcal{R}}(x,\epsilon), \tag{4.4.5}$$

where

$$\forall i \in \{1,\dots,n\},\ \mathcal{R}_i(x,\epsilon) = \sum_{p=1}^{k} \varphi_{i,p}(x,\epsilon)\tilde{\mathcal{J}}_p(x,\epsilon), \quad \varphi_{i,p}(x,\epsilon) \in \mathbb{K}(x,\epsilon). \tag{4.4.6}$$

Output: An optimal set of HODEs and LFs.

Corrected ϵ-order selection method:

1. Uncouple the irreducible system in all the possible ways by circular permutation of the variables $\tilde{\mathcal{I}}_i$. We get

$$n^2 = \underbrace{n}_{\text{HODEs}} + \underbrace{n(n-1)}_{\text{LFs}}$$

equations. More precisely we uncouple the system

$$\begin{aligned}
\{\underline{\tilde{\mathcal{I}}_1}, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \dots, \tilde{\mathcal{I}}_{n-2}, \tilde{\mathcal{I}}_{n-1}, \tilde{\mathcal{I}}_n\} &\text{ in } \tilde{\mathcal{I}}_1\\
\{\underline{\tilde{\mathcal{I}}_2}, \tilde{\mathcal{I}}_3, \tilde{\mathcal{I}}_4, \dots, \tilde{\mathcal{I}}_{n-1}, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_1\} &\text{ in } \tilde{\mathcal{I}}_2\\
\{\underline{\tilde{\mathcal{I}}_3}, \tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5, \dots, \tilde{\mathcal{I}}_n, \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2\} &\text{ in } \tilde{\mathcal{I}}_3\\
&\vdots\\
\{\underline{\tilde{\mathcal{I}}_n}, \tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \dots, \tilde{\mathcal{I}}_{n-3}, \tilde{\mathcal{I}}_{n-2}, \tilde{\mathcal{I}}_{n-1}\} &\text{ in } \tilde{\mathcal{I}}_n.
\end{aligned}$$

We select now those for which we can "invert" the $\boxed{3}$ relations[a], i.e., for those where we can get rid of the dummy W_i variables and obtain

uncouplings in $\tilde{\mathcal{I}}_p$, $p \in \{1, \ldots, n\}$, of the form described in Remark 4.3.1:

$$\mathcal{U}_Z^{(p,n)} : \left\{ \underbrace{\tilde{\mathcal{I}}_p^{(H,n)}}_{\text{HODE}}, \underbrace{\tilde{\mathcal{I}}_{p+1}^{(p,n)}, \ldots, \tilde{\mathcal{I}}_n^{(p,n)}, \tilde{\mathcal{I}}_1^{(p,n)}, \ldots, \tilde{\mathcal{I}}_{p-1}^{(1,n)}}_{\text{LFs}} \right\}. \tag{4.4.7}$$

2. Following the most extended correction from Section 4.3.3.2, we perform uncouplings of subsystems to replace:

$$\begin{aligned}\left\{\mathcal{U}_Z^{(1,n)}, \mathcal{U}_Z^{(2,n)}, \ \ldots, \mathcal{U}_Z^{(n-1,n)}, \mathcal{U}_Z^{(n,n)}\right\} \to \\ \left\{\mathcal{U}_Z^{(1,n)}, \tilde{\mathcal{U}}_Z^{(2,n)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n)}, \tilde{\mathcal{U}}_Z^{(n,n)}, \right. \\ \tilde{\mathcal{U}}_Z^{(2,n;1)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;1)}, \tilde{\mathcal{U}}_Z^{(n,n;1)}, \\ \tilde{\mathcal{U}}_Z^{(2,n;2)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;2)}, \tilde{\mathcal{U}}_Z^{(n,n;2)}, \\ \vdots \\ \left. \tilde{\mathcal{U}}_Z^{(2,n;n-1)}, \ldots, \tilde{\mathcal{U}}_Z^{(n-1,n;n-1)}, \tilde{\mathcal{U}}_Z^{(n,n;n-1)} \right\}.\end{aligned}$$

3. As described in Section 4.3.3.3 we also add all possible Gauss uncouplings.

4. We expand the tree through all possible paths to obtain all the candidates of the form

$$\mathcal{S}_i = \{\tilde{\mathcal{I}}_{\sigma_i(1)}^{i_1,j_1}, \ldots, \tilde{\mathcal{I}}_{\sigma_i(n)}^{i_1,j_1}\}$$

where $\sigma_1 \in \mathfrak{S}(n)$.

5. Apply the ϵ-order correction procedure to each of the $\mathcal{S}_i$ as outlined in Section 4.4.1.

6. Replace the symbolic orders a_i in the correction functions C, $\bar{C}$ by an explicit integer, for example for all i, we take $a_i = 0$.

7. In order to be able to select the optimal solving strategy with the best order, we do as in Step 5 of Section 4.3.2.1, i.e., we associate order matrices to each set $\mathcal{S}_i$ and then apply (4.3.5) and (4.3.6).

[a] See page 196.

Remark 4.4.4. *Both the linear selection method*[118]*, the ϵ-order selection method*[119]*, as well as all their refinements are applied in parallel. Furthermore, once again we*

[118] See Section 4.3.
[119] See Section 4.4.1.

can apply at the end (4.3.5) and then (4.3.6) to discriminate between all possible results, which ensures that as many possible paths have been explored.

Remark 4.4.5. *We see that while the linear selection method is much faster because we select line by line, it also does not take into account all the retro-correction, which could possibly lead to suboptimal corrections. On the contrary, the ϵ-order selection method takes those into consideration, but requires that we expand the full selection tree and compare all possible uncouplings, which makes it much slower. In order to tackle this problem, one can technically put time constraints on the application of the ϵ-order selection procedure for each of the $\mathcal{S}_i$ — in this case, maybe not all the configurations will be computed, but computation time can be reduced to be applicable for concrete problems.*

4.5. Global corrections and trees

In the previous two sections, we have first explained how, given an irreducible subsystem, we can uncouple it in different schemes and then select the best equations. Further, we explained how for a given set of equations obtained through this procedure, it was possible to compute the necessary corrections for the BIs and MIs. Since everything is done at the level of the irreducible subsystems and we have the local correction rules, we simply need to combine these corrections for each of the subsystems to get the full correction for the original system. Due to the preprocessing from Section 4.1, the irreducible subsystems we obtain are triangularisations of the full system, and therefore the highest system will have many MIs appearing as induced base case integrals or IBIs[120] while the lowest one will be the one where we will only need real BIs to be solved. We define the following notion:

Definition 4.5.1 (Root master integrals)**.** *We call the **root MIs** (RMIs) the MIs that do not appear as IBIs in any equation in all of the uncouplings.*

Definition 4.5.2 (Full base case integrals)**.** *We call **full BIs** (FBIs) the BIs that are not IBIs.*

Remark 4.5.1. *Since RMIs do not get any retro-correction, thet define in a sense the "root" of the correction trees, hence the name.*

In order to obtain the full correction, we proceed as follows:

1. We **identify the RMIs**.
2. We **apply the aforementioned correction rules recursively** on the

[120] See Remark 4.1.2.

RMIs, which gives rise to a tree where the roots are the RMIs, the nodes are the IBIs and the treetops are the fFBIs.

As usual, let us first work out a toy example before formalising these steps:

Example 4.5.1. *Let us consider an initial system of order $n = 5$, i.e., the MIs are $\tilde{\mathcal{I}}_i(x, \epsilon)$, $i \in \{1, \dots, 5\}$, and there is one FBI $\mathcal{J}$. Suppose the system splits into three irreducible subsystems:*

MIs	*(Full) Dependence*
$\tilde{\mathcal{I}}_1$	$\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5; \mathcal{J}$
$\tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3$	$\tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3, \tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5; \mathcal{J}$
$\tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5$	$\tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5; \mathcal{J}$

1. *The only MI that does not appear in any other uncoupling as an IBI is $\tilde{\mathcal{I}}_1$, it is the only RMI.*

2. *Now suppose that after the computation of the local $\epsilon-$order corrections, we obtained the following rules:*

$$\{(\tilde{\mathcal{I}}_1, a_1)\} \to \{\{(\tilde{\mathcal{I}}_1, \alpha_1^{(1)})\}, \{\{(\tilde{\mathcal{I}}_2, \beta_2^{(1)}), (\tilde{\mathcal{I}}_3, \beta_3^{(1)}), (\tilde{\mathcal{I}}_4, \beta_4^{(1)}), (\tilde{\mathcal{I}}_5, \beta_5^{(1)}), (\mathcal{J}, \gamma^{(1)})\}\}\} \tag{4.5.1}$$

$$\{(\tilde{\mathcal{I}}_2, a_2), (\tilde{\mathcal{I}}_3, a_3)\} \to \{\{(\tilde{\mathcal{I}}_2, \alpha_2^{(2)}), (\tilde{\mathcal{I}}_3, \alpha_3^{(2)})\}, \{\{(\tilde{\mathcal{I}}_4, \beta_4^{(2)}), (\tilde{\mathcal{I}}_5, \beta_5^{(2)}), (\mathcal{J}, \gamma^{(2)})\}\}\} \tag{4.5.2}$$

$$\{(\tilde{\mathcal{I}}_4, a_4), (\tilde{\mathcal{I}}_5, a_5)\} \to \{\{(\tilde{\mathcal{I}}_4, \alpha_4^{(3)}), (\tilde{\mathcal{I}}_5, \alpha_5^{(3)})\}, \{\{(\mathcal{J}, \gamma^{(3)})\}\}\} \tag{4.5.3}$$

where the $\alpha_i^{(j)}, \beta_i^{(j)}, \gamma^{(j)}$ are functions of the symbolic initial orders a_i, i.e. we initialize the MIs as follows:

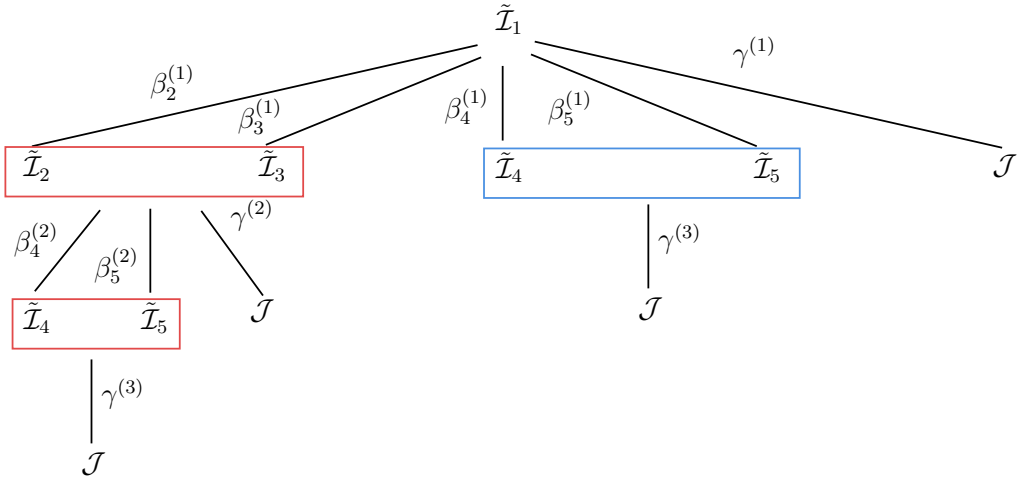
$$(\tilde{\mathcal{I}}_1, a_1), \dots, (\tilde{\mathcal{I}}_5, a_5)$$

and we have:

$$\alpha_1^{(1)} \equiv \alpha_1^{(1)}(a_1),$$
$$\forall i \in \{2, 3\},\ \alpha_i^{(2)} \equiv \alpha_i^{(2)}(a_2, a_3),$$
$$\forall i \in \{4, 5\},\ \alpha_i^{(3)} \equiv \alpha_i^{(3)}(a_4, a_5),$$
$$\forall i \in \{1, \dots, 5\},\ \beta_i^{(1)} \equiv \beta_i^{(1)}(a_1),$$
$$\forall i \in \{4, 5\},\ \beta_i^{(2)} \equiv \beta_i^{(2)}(a_2, a_3),$$
$$\gamma^{(1)} \equiv \gamma^{(1)}(a_1),\ \gamma^{(2)} \equiv \gamma^{(2)}(a_2, a_3),\ \gamma^{(3)} \equiv \gamma^{(3)}(a_4, a_5).$$

3. *We see that $\tilde{\mathcal{I}}_1$ depends for instance on $\tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3$, and those two in turn depend*

on $\tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5$, which finally gives a correction for $\tilde{\mathcal{J}}$. $\tilde{\mathcal{I}}_1$ also directly depends on $\tilde{\mathcal{I}}_4, \tilde{\mathcal{I}}_5$, which give also a correction for $\tilde{\mathcal{J}}$. But different from the previous one we do not consider intermediate corrections coming from $\tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3$. Now we build a tree with $\tilde{\mathcal{I}}_1$ as a root, following the rules (4.5.1),(4.5.2),(4.5.3). We can represent this diagramatically with a tree:



Remark 4.5.2. *We note that the corrections are applied in a recursive way, i.e., following a branch we need to compose the $\alpha_i^{(j)}/\beta_i^{(j)}/\gamma^{(j)}$ functions from the outermost node all the way down to the root. For instance, if we follow the branches singled out with the red squares, we obtain*

$$\gamma^{(3)}\left(\beta_4^{(2)}\left(\beta_2^{(1)}\left(a_1\right),\beta_3^{(1)}\left(a_1\right)\right),\beta_5^{(2)}\left(\beta_2^{(1)}\left(a_1\right),\beta_3^{(1)}\left(a_1\right)\right)\right).$$

We note also that due to this composition of the branches, all the internal corrections — be it for the MIs, due to the choice of uncouplings, or the BIs itself — will depend only on the orders of the RMIs, so here the order a_1 of $\tilde{\mathcal{I}}_1$.

Remark 4.5.3. *We note that the procedure of building the tree by applying successively the rules (4.5.1),(4.5.2),(4.5.3) terminates as each step will either add a new variable (MI or BI) to the tree, and there is only a finite number of them, or nothing happens, but that means that it was the last step.*

4. *We collect in this way all the corrections at the end of the branches, and we get the following list:*

$$L_{corr}:\left\{\begin{array}{c}\gamma^{(3)}\left(\beta_4^{(2)}\left(\beta_2^{(1)}\left(a_1\right),\beta_3^{(1)}\left(a_1\right)\right),\beta_5^{(2)}\left(\beta_2^{(1)}\left(a_1\right),\beta_3^{(1)}\left(a_1\right)\right)\right)\\ \gamma^{(2)}\left(\beta_2^{(1)}\left(a_1\right),\beta_3^{(1)}\left(a_1\right)\right)\\ \gamma^{(3)}\left(\beta_4^{(1)}\left(a_1\right),\beta_5^{(1)}\left(a_1\right)\right)\\ \gamma^{(1)}\left(a_1\right)\end{array}\right\} \quad (4.5.4)$$

Remark 4.5.4. *If we had several sets of RMIs, i.e., several trees, we would have had to return the maximum of all the lists $L_{corr,i}$, i.e.,*

$$\max(\{\max(L_{corr,i})\}_i),$$

where i describes a finite subset of $\mathbb{N}$.

5. *Since we have to satisfy all of these corrections, i.e., $\mathcal{J}$ will effectively get at some step of the computations ϵ coefficients to the power of each element of this list, we simply return $\max(L_{corr})$ as the $\epsilon-$order correction for $\mathcal{J}$.*

Remark 4.5.5. *The last step gives us the full ϵ-order correction for the BIs. We also directly provide the orders up to which we have to expand the MIs themselves to do all the computations in a consistent way. When solving recursively the system, one has to expand each MI to all of the orders appearing in the tree, so we need to be able to expand up to the maximum. Thus, it suffices to flatten the tree and take the maximum of all the orders that appear in there. In our case, flattening the tree we obtain the following set of corrections for each of the MIs:*

$$L_{corr}^{\tilde{\mathcal{I}}_1} : \left(\alpha_1^{(1)}(a_1) \right), \tag{4.5.5}$$

$$L_{corr}^{\tilde{\mathcal{I}}_2} : \left(\alpha_2^{(2)} \left(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1)\right) \right), \tag{4.5.6}$$

$$L_{corr}^{\tilde{\mathcal{I}}_3} : \left(\alpha_3^{(2)} \left(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1)\right) \right), \tag{4.5.7}$$

$$L_{corr}^{\tilde{\mathcal{I}}_4} : \begin{pmatrix} \alpha_4^{(3)} \left(\beta_4^{(2)}(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1)), \beta_5^{(2)}(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1))\right) \\ \alpha_4^{(3)} \left(\beta_4^{(1)}(a_1), \beta_5^{(1)}(a_1)\right) \end{pmatrix}, \tag{4.5.8}$$

$$L_{corr}^{\tilde{\mathcal{I}}_5} : \begin{pmatrix} \alpha_5^{(3)} \left(\beta_4^{(2)}(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1)), \beta_5^{(2)}(\beta_2^{(1)}(a_1), \beta_3^{(1)}(a_1))\right) \\ \alpha_5^{(3)} \left(\beta_4^{(1)}(a_1), \beta_5^{(1)}(a_1)\right) \end{pmatrix}. \tag{4.5.9}$$

The corrections are therefore for all $i \in \{1, \ldots, 5\}$

$$\left(\tilde{\mathcal{I}}_i, \max(L_{corr}^{\tilde{\mathcal{I}}_i})\right).$$

6. *It suffices to replace now all the symbolic orders by the actual orders given as input by the user in order to get the corrections for the BIs and the MIs.*

4.5.1. Method of combining the subsystem corrections

Input: Suppose we have clusterized our system of order n with m BIs in the form of p irreducible subsystems:

$$\forall \ell \in \{1,\dots,p\}, \quad \mathcal{S}_\ell : \frac{\mathrm{d}}{\mathrm{d}x}\bar{\mathcal{I}}_\ell(x,\epsilon) = \bar{\mathcal{M}}_\ell(x,\epsilon) + \bar{\mathcal{R}}_\ell(x,\epsilon), \tag{4.5.10}$$

where

- $\bar{\mathcal{I}}_\ell = \left(\mathcal{I}_{i^\ell_1},\dots,\mathcal{I}_{i^\ell_{q_\ell}}\right)$, $q_\ell \in \mathbb{N}$, $q_\ell \leq n$
- $\bar{\mathcal{R}}_\ell = \left(\mathcal{R}_{\ell,i^\ell_1},\dots,\mathcal{R}_{\ell,i^\ell_1}\right)$ with

$$\mathcal{R}_{\ell,i^\ell_j} \in \langle \mathcal{J}_1,\dots,\mathcal{J}_m,\mathcal{I}_{i^{p-\ell}_1},\dots,\mathcal{I}_{i^{p-\ell}_{q_{p-\ell}}},\dots,\mathcal{I}_{i^p_1},\dots,\mathcal{I}_{i^p_{qp}}\rangle_{\mathbb{K}(x,\epsilon)}$$

  so that for a system ℓ the inhomogeneous side can contain IBPs from all the systems below[a].

Furthermore,we suppose we managed to compute the ϵ-order corrections for each of the subsystems $\mathcal{S}_\ell$ in the form:

$$\begin{aligned}\left\{\left(\mathcal{I}_{i^\ell_1},a_{\xi_\ell+1}\right),\dots,\left(\mathcal{I}_{i^\ell_{q_\ell}},a_{\xi_\ell+q_\ell}\right)\right\} \to \\ \left\{\{\left(\mathcal{I}_{i^\ell_1},\alpha^{(\ell)}_{i^\ell_1}\right),\dots,\left(\mathcal{I}_{i^\ell_{q_\ell}},\alpha^{(\ell)}_{i^\ell_{q_\ell}}\right)\},\right. \\ \left.\{\{\left(\mathcal{I}_{i^{p-\ell}_1},\beta^{(\ell)}_{i^{p-\ell}_1}\right),\dots,\left(\mathcal{I}_{i^p_{q_p}},\beta^{(\ell)}_{i^p_{q_p}}\right),\left(\mathcal{J}_1,\gamma^{(\ell)}_1\right),\dots,\left(\mathcal{J}_m,\gamma^{(\ell)}_m\right)\}\}\right\}\end{aligned} \tag{4.5.11}$$

where we have $\xi_\ell := \sum_{u=1}^{\ell-1} q_u$ for $\ell > 1$ and otherwise $\xi_1 = 0$ and

$$\forall j \in \{1,\dots,q_\ell\}, \quad \alpha^{(\ell)}_{i^\ell_j} \equiv \alpha^{(\ell)}_{i^\ell_j}(a_{\xi_\ell+1},\dots,a_{\xi_\ell+q_\ell})$$

$$\forall v \in \{p-\ell,\dots,p\}, \forall w_v \in \{1,\dots,q_v\}, \quad \beta^{(\ell)}_{i^v_{w_v}} \equiv \beta^{(\ell)}_{i^v_{w_v}}(a_{\xi_\ell+1},\dots,a_{\xi_\ell+q_\ell})$$

$$\forall j \in \{1,\dots,m\}, \quad \gamma^{(\ell)}_j \equiv \gamma^{(\ell)}_j(a_{\xi_\ell+1},\dots,a_{\xi_\ell+q_\ell}).$$

Output: The full ϵ-correction orders for the MIs and BIs.

[a] See Section 4.1.

Method:

1. We identify first the root master integrals. For that, for each MI $\tilde{\mathcal{I}}_i$, it

suffices to do an extensive search through the elements in red brackets (second subset) of all the rules (4.5.11), more explicitely if $\tilde{\mathcal{I}}_i$ appears nowhere in the

$$\left\{ \left(\mathcal{I}_{i_1^{p-\ell}}, \beta^{(\ell)}_{i_1^{p-\ell}}\right), \ldots, \left(\mathcal{I}_{i^p_{q_p}}, \beta^{(\ell)}_{i^p_{q_p}}\right), \left(\mathcal{J}_1, \gamma_1^{(\ell)}\right), \ldots, \left(\mathcal{J}_m, \gamma_m^{(\ell)}\right) \right\}$$

elements, then $\tilde{\mathcal{I}}_i$ can be added to the set of RMIs. After having performed that operation for all of the MIs we obtain the full set of RMIs, of size[a] $\bar{p} \leq n$, denoted with

$$\mathcal{S}_{RMI} = \{\tilde{\mathcal{I}}_{k_1}, \ldots, \tilde{\mathcal{I}}_{k_{\bar{p}}}\}, \tag{4.5.12}$$

with for all $i \in \{1, \ldots, \bar{p}\}$, $1 \leq k_i \leq n$ and all k_i distincts.

2. Once all the RMIs are identified, we group them according to the LHS of (4.5.11). That is to say, we write

$$\bar{\mathcal{S}}_{RMI} = \bigcup_{\bar{\ell} \in L} \left\{ \underbrace{\left\{ \mathcal{I}_{i_1^{\bar{\ell}}}, \ldots, \mathcal{I}_{i^{\bar{\ell}}_{q_{\bar{\ell}}}} \right\}}_{=: \bar{\mathcal{S}}^{\bar{\ell}}_{RMI}} \right\} \tag{4.5.13}$$

where L is a subset[b] of $\{1, \ldots, p\}$.

3. Now for each subset $\bar{\mathcal{S}}^{\bar{\ell}}_{RMI}$ of RMIs of $\bar{\mathcal{S}}_{RMI}$ we do the following:
   - Apply repeatedly the rules[c] (4.5.11) to $\bar{\mathcal{S}}^{\bar{\ell}}_{RMI}$ until the obtained sets stop changing. If we denote schematically the rules as

     $$\{\text{MI}\} \to \{\{\text{MI corrections}\}, \{\{\text{(I)BI corrections}\}\}\}$$

     we mean repeatedly applying the rules on the {(I)BI corrections} part. The nested set of sets we obtain is what we call the "tree", and the most nested elements will be called the "treetops".
   - We collect all the treetops — by construction[d], those will contain only real BIs $\mathcal{J}_i$ and no IBIs $\tilde{\mathcal{I}}_i$.
   - We obtain sets of the form (4.5.4), i.e., more generally for each $\mathcal{J}_i$, $i \in \{1, \ldots, q\}$ we get:

     $$L^{\bar{\mathcal{S}}^{\bar{\ell}}_{RMI}}_{corr,i} : \bigcup_{p \in \bar{L}} \left\{ \gamma_i^{(p)} \left(\text{nested compositions of } \beta \text{ functions}\right) \right\} \tag{4.5.14}$$

     where $\bar{L}$ is a subset[e] of $\{1, \ldots, p\}$.

- We also need to compute corrections for the MIs $\tilde{\mathcal{I}}_i$. For that, it suffices to flatten the tree up to a set of tuples $\{A, B\}$ where A is either a MI $\tilde{\mathcal{I}}_i$ or a BI $\mathcal{J}_i$, and B a nested composition of correction functions α, β, γ. Now for each of the $\tilde{\mathcal{I}}_i$, $i \in \{1, \dots, n\}$, we simply single out all elements of the form $\{\tilde{\mathcal{I}}_i, \alpha_i^{(t)}(\text{composition of } \beta \text{ functions})\}$ in the set
$$L_{corr}^{\bar{\mathcal{S}}_{RMI}^{\bar{\ell}}, \tilde{\mathcal{I}}_i} \tag{4.5.15}$$
- For a given set $S = \{a_1, \dots, a_j\}$, denoting the maximum of the elements of S with $\max(L) = \max\{a_1, \dots, a_j\}$, we get the correction for each $\mathcal{J}_i$
$$C_{\mathcal{J}_i}^{(\bar{\ell})} = \max\left(L_{corr,i}^{\bar{\mathcal{S}}_{RMI}^{\bar{\ell}}}\right) \tag{4.5.16}$$
and the correction for each $\tilde{\mathcal{I}}_i$ appearing in the tree
$$C_{\tilde{\mathcal{I}}_i}^{(\bar{\ell})} = \max\left(L_{corr}^{\bar{\mathcal{S}}_{RMI}^{\bar{\ell}}, \tilde{\mathcal{I}}_i}\right). \tag{4.5.17}$$

4. Finally the full correction is obtained by taking the maximum of the corrections for each tree:
$$\tilde{\mathcal{J}}_i : \boxed{C_{\mathcal{J}_i} = \max\left(\left\{C_{\mathcal{J}_i}^{(\bar{\ell})}\right\}_{\bar{\ell} \in L}\right)} \tag{4.5.18}$$
$$\tilde{\mathcal{I}}_i : \boxed{C_{\tilde{\mathcal{I}}_i} = \max\left(\left\{C_{\tilde{\mathcal{I}}_i}^{(\bar{\ell})}\right\}_{\bar{\ell} \in L}\right)} \tag{4.5.19}$$

[a] If there are $\bar{p} = n$ RMIs it simply means the original system was already irreducible.
[b] In other words we clustered the trees in subforests which are fully unrelated to each other. Splitting the problems to unrelated subproblems lead to a significant speedup.
[c] In the sense described in Section 4.1.
[d] Indeed, if we suppose that the rule contains additonal IBIs, it means that the corresponding subsystem $\mathcal{S}$ depends on another subsystem $\mathcal{S}'$ where those IBIs are seen as MIs, and in particular there will be a rule for $\mathcal{S}'$ which can be applied on the rule for the subsystem $\mathcal{S}$.
[e] It simply means that depending on how the rules are applied, we can get only a subset of all the possible $\gamma_i^{(\ell)}$ correcting functions at the treetops for a given $\mathcal{J}_i$.

Remark 4.5.6. *In the first substep of Step 3 of the method, we say we apply repeatedly the rules. There exist however one subtle point: it might happen that in the (I)BI corrections part in red brackets (second subset of brackets), not all the IBIs that form the full LHS of another side appear. For instance, suppose that there exists a rule that has the MIs* $\{\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_2, \tilde{\mathcal{I}}_3\}$ *but one rule contains in the RHS in*

the red brackets (second subset of brackets) only $\{\tilde{\mathcal{I}}_1, \tilde{\mathcal{I}}_3\}$ *— in that case* $\tilde{\mathcal{I}}_2$ *remains untouched. We perform in this case a slight modification which consist to associate to each rule of the form* (4.5.11) *a set of what we call "incomplete rules" of the following form. Denoting with* $Q_\ell = \{1, \ldots, q_\ell\}$ *we apply in addition*

$$\left\{ \begin{array}{c} \left\{\left(\mathcal{I}_{i_j^\ell}, a_{\xi_\ell + j}\right)\right\}_{j\in S} \rightarrow \\ \left\{ \{ \left(\mathcal{I}_{i_1^\ell}, \alpha^{(\ell)}_{i_1^\ell}\right), \ldots, \left(\mathcal{I}_{i_{q_\ell}^\ell}, \alpha^{(\ell)}_{i_{q_\ell}^\ell}\right) \}, \\ \{\{ \left(\mathcal{I}_{i_1^{p-\ell}}, \beta^{(\ell)}_{i_1^{p-\ell}}\right), \ldots, \left(\mathcal{I}_{i_{q_p}^p}, \beta^{(\ell)}_{i_{q_p}^p}\right), \left(\mathcal{J}_1, \gamma_1^{(\ell)}\right), \ldots, \left(\mathcal{J}_m, \gamma_m^{(\ell)}\right) \}\} \end{array} \right\} \Big|_{\{a_{\bar{j}} \to \bar{a}_{\bar{j}}\}_{\bar{j}\in(Q_\ell \setminus S)}} \Bigg\}_{S\subset Q_\ell} \tag{4.5.20}$$

where the $\bar{a}_{\bar{j}}$ *represent the orders wanted for the MIs given by the user. To summarise, we generate subrules where instead of keeping the symbolic order we replace those for the missing MIs by the actual numerical value. Therefore, with those extra rules it becomes always possible to build a tree.*

Throughout the sections of this chapter we have described our system analysis method by breaking it down in three main steps and using toy examples: in Section 4.1, we clustered the main system in irreducible subsystems, in Section 4.4 we gradually refined a method to select optimal equations for a given subsysem, in Section 4.3 we proceeded to present the local analysis of each irreducible system and extraction of symbolic ϵ-order corrections and finally in Section 4.5 we recombined the previously obtained results in order to get the correction for the full system.

All these tools has been implemented in the package `SystemAnalysis` by executing the function `analysisSystem`. Its full functionality and concrete examples will be illustrated in Appendix B.

5. Conclusion

In the course of this thesis, we explored two different problems: the first one consists in linking certain classes of sums with their integral representations in order to compute their asymptotic expansions, and the second one aimed at optimising uncouplings of first order systems of single variate differential equations, in both cases through the use of computer algebra tools already available to us or new ones that we developed and implemented.
In Chapter 2, we used as a basis for our study the method presented in [16], which decomposed into two parts. In the first part (Sections 2.2 — 2.3), after reminding some base properties of the Mellin transforms and proving specific convolution cases (Section 2.2.1), we properly formalised the full recursive inversion method in Section 2.3.2 which has been implemented in our package `RICA`. We explored some issues related to convolutions, such as the +-prescription case where we could discover and prove some relevant identities (Section 2.3.5) and we also introduced a whole new class of functions in Definition 2.3.3 (GCPLs) to be able to deal with cases that did not fit in the setting of [16] (Section 2.3.4). Using this knowledge and the related automatic tools, we focused on the the algorithmic theory to compute asymptotic expansions (Sections 2.4 — 2.5). After some reminders about classical results concerning asymptotics, in particular of functions defined as integrals (Section 2.4.1), we managed to prove a new theorem that became a very powerful tool to compute asymptotic expansions of GCPLs around $x = 1$ provided that a bounding criterion on the derivative of its constitutent letters was verified (Section 2.4.2). This was in turn implemented in `RICA`, so that computations of asymptotic expansions of the GCPL functions are automatically guaranteed. Finally, we turned towards asymptotic expansions at $n \to +\infty$ of Mellin representations of binomial sums (Section 2.5) — there we presented two simple methods, based on factorial series (Section 2.5.1) and geometric series (Section 2.5.2), both implemented in `RICA`, comparatively fast but both limited in scope, before shifting to a general method, the change of variables method (Section 2.5.3). Based on the classical Watson's lemma, that we extended in different ways for our own usage, it allows us to easily obtain asymptotic expansions of integral representations, in particular using our algorithms to compute the Mellin transforms, provided that we manage to get the asymptotic expansions of the integrand. This is where we made the connection with our results concerning the GCPL functions (Section 2.5.4), which allowed us to obtain asymptotics of a large subclass of general binomial sums that

could not be treated so far.
The method we have studied works like a dictionary, so its power lies on the "building blocks" that are precomputed and input like base "rules" — those correspond to Mellin inverses of the internal summands of the nested sums we are studying. While we concentrated on binomial sums with a central binomial coefficient, an additional possible future task would be to try to identify Mellin inverses of more complicated and/or different objects, such as

$$\frac{1}{\sqrt{n}} = \frac{1}{\sqrt{\pi}}\mathbf{M}\left[\frac{1}{x\sqrt{-\log x}}\right](n).$$

As we have seen in Section 2.1.2.1, not all objects can be represented as a classical inverse Mellin transform — and while some studies in this direction are possible, it would need to deeply modify the theory we were building. Conversely, it has become clear at the end of this study that the GCPL representation plays an important role that may be explored and extended further in the future. In particular, we only studied basic defining properties of GCPLs (Section 2.3.4), but because any Mellin convolution could be represented with them, it would be also possible to explore their algebraic properties for instance by comparing (when possible) Mellin inverse transform results obtained with and without GCPL usage. This could lead to new insights about elliptic structures which seem to be buried partially beneath binomial objects. Coming back to the link between GCPL and asymptotics, Theorem 2.4.5 is the corner stone for computing asymptotics of GCPL functions. It requires to bound the derivatives of constituent letters of the GCPLs. While Method 2 on page 102 works for binomial structures, it might fail for certain classes. A more systematic study how to perform this task is necessary and we strongly believe that one could achieve asymptotic expansions of much more general GCPLs, and *in fine* of associated sums. Finally, on the asymptotics themselves, while our method allows us to obtain in the binomial and extended binomial case expansions with symbolic constants, because those are in general GCPL constants, further studies are necessary to bound the constant error in the big O-term — in particular, either by explicit boundings of the GCPL constants or by precise numerical approximations.

In Chapter 3, we presented an overview of the classical methods to deal with first order single variate systems of differential equations, provided that those were involving a parameter ϵ. In Chapter 4, we therefore took as a basis the method which consisted to uncouple a differential system first and then expand in ϵ to propose a method to optimise the inputs, more precisely that of the expansion in ϵ of the base case integrals. We split the method in several parts. First, we perform an analysis of the full system that we want to solve in order to split it in smaller irreducible subsystems (Section 4.1). The second part consists in dealing

individually with each of the subsystems (Section 4.3). More precisely, we proposed a method as general as possible to uncouple the subsystem in different schemes and then select differential equations and linear relations in a way that minimises the ϵ-orders for the base case integrals, or the order of the differential equations or associated recurrences. At the same time, for each system we develop a method to precisely compute the ϵ-corrections given a set of equations obtained from above (Section 4.4). Once each subsystem has been dealt with, we combine everything (Section 4.5), i.e., having obtained the symbolical corrections for each subsystem, we can reconstruct the ϵ-corrections needed in the full system. The method has been implemented in a small package called `SystemAnalysis`, where the procedure above can be applied to a system through a unique centralised function.
The method described above is rather general — as it takes into account different schemes of uncouplings and tries to get as many equations as possible — but can be extended further. One of its bottlenecks is the fact that in the Zürcher scheme, sometimes it is not possible to fully eliminate dummy variables — when this happens, those could be possibly added to the set of master integrals but this complexifies the problem, as there are more variables to solve and comparisons between different systems become equivocal as they can have different number of variables. Making sense of these comparisons, identifying new criterions for dinstiction between solutions or more indepth study of the Zürcher uncoupling could be possible paths of further explorations. Another technical issue arises because the method is trying to be as exhaustive as possible. Because for a system of order n there are theoretically $n!$ Gauß uncouplings there is a time-quality trade off that happens. We try to search for better choices but it can take much more time and Mathematica gets slower the bigger the irreducible subsystems are. Thus optimisation of time issues would allow us to get a larger choice of relations and obtain possibly better solutions. Finally, we concentrated on the method described in Section 3.4 but a radically different approach would consist into shifting to the "expand first, uncouple then" method from Section 3.3 — which would need to change in depth the approach but could possibly present an interesting alternative.

A. The RICA package, a userguide

In order to use the `RICA` package, `Sigma` [1,94] and `HarmonicSums` [9,10,12,72–74] should always be loaded beforehand[121]

```
In[1]:= << Sigma.m;
```

```
Sigma - A summation package by Carsten Schneider – © RISC
```

```
In[2]:= << HarmonicSums.m;
```

```
HarmonicSums by Jakob Ablinger – © RISC
```

and then our package can be loaded also

```
In[3]:= << RICA.m;
```

```
Rule Induced Convolutions for Asymptotics (RICA) package by Nikolai Fadeev © RISC-JKU
```

We detail below the list of available functions with their respective parameters, and examples of usage. We put arguments between brackets when they are optional, e.g. `(opts)`. If the output is the same as in the input we will write "Idem". If there are no options, we will write $\emptyset$.

A.1. Basic objects and transformations

A.1.1. Hwb

- **Input**: `Hwb[{fn1,...,fnk},x,(opts)]`.
- **Output**: **Idem**.
- **Options**: $\emptyset$.
- **Details**: A type of generalized iterated integral similar to `Hw`, except for the change in convention of integration bounds: for `Hw` the integration bounds are from 0 to `x` where for `Hwb` it is from `x` to 1. See Definition 2.1.13 for more details. The indices are $n_i \in \{0, -1, -2, 1, 2, w_1, ..., w_{32}\}$.
- **Examples**:

[121] All packages are freely available at `https://caa.risc.jku.at`.

In[4]:= **Hwb[f1, x]**

Out[4]= Hwb[f1, x]

In[5]:= **Hwb[f2, fw8, f0, f1, x]**

Out[5]= Hwb[f2, fw8, f0, f1, x]

A.1.2. ConvertGLbToGL

- **Input**: `ConvertGLbToGL[expr]`
- **Output**: `expr` where all `GLb` functions are converted to `GL` functions
- **Options**: $\emptyset$
- **Details**: Given an expression `expr` which contains a linear or polynomial combination of `GLb` functions will convert all of them to `GL` representation and return it. See Property 2.1.6 for more details.
- **Examples**:

In[6]:= $\text{ConvertGLbToGL}\left[\text{GLb}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}+\mathbf{1}}, \sqrt{\mathbf{log}(\text{VarGL})+\mathbf{1}}\right\}, \mathbf{x}\right]\right]$

Out[6]= $\text{GL}\left[\left\{\frac{1}{2-\text{VarGL}}, \sqrt{\log(1-\text{VarGL})+1}\right\}, 1-\text{x}\right]$

In[7]:= $\text{ConvertGLbToGL}\left[\mathbf{3}\text{GLb}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}}, \frac{\mathbf{1}}{\text{VarGL}+\mathbf{2}}\right\}, \mathbf{x}\right] + \text{GLb}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}}\right\}, \mathbf{x}\right]\text{GLb}\left[\left\{\frac{\mathbf{1}}{\sqrt{\text{VarGL}(\mathbf{1}-\text{VarGL})}}\right\}, \mathbf{x}\right]\right]$

Out[7]= $\text{GL}\left[\left\{\frac{1}{1-\text{VarGL}}\right\}, 1-\text{x}\right]\text{GL}\left[\left\{\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}}\right\}, 1-\text{x}\right] + 3\text{GL}\left[\left\{\frac{1}{(1-\text{VarGL})^2}, \frac{1}{3-\text{VarGL}}\right\}, 1-\text{x}\right]$

A.1.3. ConvertGLToGLb

- **Input**: `ConvertGLToGLb[expr]`
- **Output**: `expr` where all `GL` functions are converted to `GLb` functions
- **Options**: $\emptyset$
- **Details**: Given an expression `expr` which contains a linear or polynomial

combination of `GL` functions will convert all of them to `GLb` representation and return it. See Property 2.1.7 for more details.

- **Examples**:

In[8]:= $\text{ConvertGLToGLb}\left[\text{GL}\left[\left\{\frac{1}{\sqrt{\text{VarGL}+1}\text{VarGL}}, \frac{1}{\text{VarGL}}\right\}, x\right]^2 - 2\text{GL}\left(\left\{\frac{1}{\text{VarGL}+1}\right\}, x\right]\right]$

Out[8]= $\text{GLb}\left[\left\{\frac{1}{(1-\text{VarGL})\sqrt{2-\text{VarGL}}}, \frac{1}{1-\text{VarGL}}\right\}, 1-x\right]^2 - 2\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\}, 1-x\right]$

A.1.4. ConvertHwb

- **Input**: `ConvertHwb[Hwb[{fi1,...,fin},x],(opts)]`
- **Output**: `GLb[{fi1[VarGL],...,fin[VarGL]},x]`
- **Options**: `AsGL` (Boolean rule)
- **Details**: Converts a `Hwb` function to a traditional `GLb` representation. If the `AsGL` option is set to `True`, this allows to convert the function directly to a `GL` representation using the `ConvertGLbToGL`[122] function.
- **Examples**:

In[9]:= **ConvertHwb[Hwb[f1, x]]**

Out[9]= $\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\}, x\right]$

In[10]:= **ConvertHwb[Hwb[f2, fw8, f0, f1, x]]**

Out[10]= $\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}, \frac{1}{\sqrt{\text{VarGL}-\frac{1}{4}\text{VarGL}}}, \frac{1}{\text{VarGL}}, \frac{1}{1-\text{VarGL}}\right\}, x\right]$

In[11]:= **ConvertHwb[Hwb[f2, fw8, f0, f1, x], AsGL → True]**

Out[11]= $\text{GL}\left[\left\{\frac{1}{\text{VarGL}+1}, \frac{1}{\sqrt{\frac{3}{4}-\text{VarGL}}(1-\text{VarGL})}, \frac{1}{1-\text{VarGL}}, \frac{1}{\text{VarGL}}\right\}, 1-x\right]$

A.1.5. GLbToGLCb

- **Input**: `GLbToGLCb[expr]`
- **Output**: `expr` where the `GLb` functions are rewritten as `GLCb` functions.

[122] See Appendix A.1.2.

- **Options**: ∅
- **Details**: Given a polynomial expression `expr` of `GLb` functions, rewrites them all as `GLCb` functions using the defining properties of `GCPL` in Section 2.3.4.
- **Examples**:

$$\text{In[12]:=}\ \text{GLbToGLCb}\left[\text{GLb}\left[\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}}\right\},x\right]\ \text{GLb}\left[\left\{\frac{1}{\text{VarGL}},\frac{1}{\text{VarGL}},\frac{2^{\text{VarGL}}}{1-\text{VarGL}}\right\},x\right]\right]$$

$$\text{Out[12]=}\ \text{GLCb}\left[\left\{1,\left\{\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}}\right\}\right\},x\right]\ \text{GLCb}\left[\left\{1,\left\{\frac{1}{\text{VarGL}},\frac{1}{\text{VarGL}},\frac{2^{\text{VarGL}}}{1-\text{VarGL}}\right\}\right\},x\right]$$

$$\text{In[13]:=}\ \text{GLbToGLCb}\left[\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\},x\right]^2+\frac{\text{GLb}\left[\left\{\frac{1}{\text{VarGL}},\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}}\right\},x\right)}{x}\right]$$

$$\text{Out[13]=}\ \text{GLCb}\left[\left\{1,\left\{\frac{1}{1-\text{VarGL}}\right\}\right\},x\right]^2+\text{GLCb}\left[\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{\text{VarGL}},\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}}\right\}\right\},x\right]$$

A.1.6. GLbToHwb

- **Input**: `GLbToHwb[expr]`
- **Output**: `expr` where we tried to remplace as much as possible the `GLb` functions by `Hwb`
- **Options**: ∅
- **Details**: Given a polynomial expression in `GLb` functions, we try to see if for every `GLb[arg,x]` the `arg` list is composed of letters or multiples of letters from our alphabet (2.1.43). If that is the case, we return a full `Hwb` function, otherwise we still return a `GLb` function with letters that have been identified presented as inactivated functions.
- **Examples**:

$$\text{In[14]:=}\ \text{GLbToHwb}\left[\text{GLb}\left[\left\{\frac{1}{\text{VarGL}},\frac{1}{(1-\text{VarGL})\sqrt{\text{VarGL}}}\right\},x\right]\right]$$

$$\text{Out[14]=}\ \texttt{Hwb[f0,fw9,x]}$$

$$\text{In[15]:=}\ \text{GLbToHwb}\left[\text{GLb}\left[\left\{\frac{2}{1-\text{VarGL}}\right\},x\right]\ \text{GLb}\left[\left\{\frac{1}{\sqrt{\text{VarGL}(\text{VarGL}-1)}}\right\},x\right]\right]$$

Out[15]= $-2\mathrm{i}\,\mathrm{Hwb}[\mathtt{f1},\mathtt{x}]\,\mathrm{Hwb}[\mathtt{fw1},\mathtt{x}]$

In[16]:= $\mathrm{GLbToHwb}\left[\mathrm{GLb}\left[\left\{\frac{1}{\mathrm{VarGL}},\frac{1}{(\mathrm{VarGL}+1)\sqrt{\mathrm{VarGL}}}\right\},\mathtt{x}\right]\right]$

Out[16]= $\mathrm{GLb}\left[\left\{\mathrm{f0}[\mathrm{VarGL}],\frac{1}{\sqrt{\mathrm{VarGL}}(\mathrm{VarGL}+1)}\right\},\mathtt{x}\right]$

A.1.7. GLCb

- **Input**: `GLCb[arg,x]`
- **Output**: **Idem**
- **Options**: $\emptyset$
- **Details**: Defines the closure of `GLb`, i.e. GRPL functions, under Mellin convolution. Described in detail in Section 2.3.4.
- **Examples**:

In[17]:= $\mathrm{GLCb}\left[\left\{\frac{1}{\mathrm{VarGL}},\left\{\frac{1}{1-\mathrm{VarGL}}\right\}\right\},\mathtt{x}\right]$

Out[17]= $\mathrm{GLCb}\left[\left\{\frac{1}{\mathrm{VarGL}},\left\{\frac{1}{1-\mathrm{VarGL}}\right\}\right\},\mathtt{x}\right]$

In[18]:= $\mathrm{GLCb}\left[\left\{\left\{\frac{1}{\mathrm{VarGL}},\left\{\frac{1}{1+\mathrm{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\mathrm{VarGL}(1-\mathrm{VarGL})}},\{\}\right\}\right\},\mathtt{x}\right]$

Out[18]= $\mathrm{GLCb}\left[\left\{\left\{\frac{1}{\mathrm{VarGL}},\left\{\frac{1}{1+\mathrm{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\mathrm{VarGL}(1-\mathrm{VarGL})}},\{\}\right\}\right\},\mathtt{x}\right]$

A.1.8. GLCbToIntegral

- **Input**: `GLCbToIntegral[expr,(opts)]`
- **Output**: `expr` where all the `GLCb` are converted to (inactivated) integral representation
- **Options**: `DefiniteIntegral` (Boolean rule), `UniqueIntegral` (Boolean rule), `ExampleCase` (0,1,2,3)
- **Details**: Given a polynomial expression in `GCLb` functions, rewrites them in the proper (inactive) integral form. The function in particular dynamically checks if there are some convolutions that are actually in the +-prescription

and then proceeds to write the convolutions using the formulas from Property 2.1.3. Because of this dynamical check, if the user puts some symbolic function as arguments of the GLCb, then the function will not be able to decide whether to use Property 2.1.3 or not. In this case, simply use the option `ExampleCase`. The default value of the parameter is 0 and then the checks are done dynamically. Otherwise, if set to 1, the first function will be considered to have a pole of order 1 at $x = 1$ and the integral representation will be written in consequence, if set to 2 it will be the same situation but will the second function, and set at 3 all the functions are considered to be regular. The option `DefiniteIntegral` allows us to write all the integrals with fixed bound 0 and 1 if set to `True` (the default value is `False`). In the case `DefiniteIntegral` is set to `True`, we can also (cosmetically) refine the way we want to get the result: if we set the option `UniqueIntegral` to `True`, then it will try to write as much as possible the result as (sums of) multiple integals over one unique integrand while if we leave it to its default `False` value, then we preserve visually the nested integrals structure in the result.

- **Examples**:

In[19]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},x\right]\right]$$

Out[19]=
$$\frac{\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\},x\right]}{x}$$

In[20]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},x\right]\right]$$

Out[20]=
$$\frac{\int_x^1 \frac{\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\},\frac{x}{x_1}\right]}{\sqrt{-((x_1-1)x_1]}}dx_1}{x}$$

In[21]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\frac{1}{1-\text{VarGL}},\left\{\frac{1}{1-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},x\right]\right]$$

Out[21]=
$$-\frac{\int_0^x \frac{\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\},x_1\right]}{1-x_1}dx_1}{\sqrt{-((x-1)x)}}-\int_x^1 \frac{\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\},\frac{x}{x_1}\right)\left(\sqrt{-\frac{x_1}{x_1-1}}-\frac{x}{\sqrt{-((x-1)x)}}\right]}{(x-x_1)\,x_1}dx_1$$

In[22]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\frac{1}{1-\text{VarGL}},\left\{\frac{1}{1-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},x\right],\ \text{DefiniteIntegral}\to\text{True}\right]$$

Out[22]=
$$-\frac{\int_0^1 \frac{\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\},(x-1)x_1+1\right]\left(\sqrt{-((x-1)x)}-\sqrt{(x-1)x(x_1-1)}\right)}{\sqrt{1-x_1x_1}}dx_1}{(x-1)x}-\frac{x\int_0^1 \frac{\text{GLb}\left[\left\{\frac{1}{1-\text{VarGL}}\right\},xx_1\right]}{1-xx_1}dx_1}{\sqrt{-((x-1)x)}}$$

In[23]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\left\{\frac{1}{\text{VarGL}+1},\{\}\right\},\left\{\frac{1}{\text{VarGL}\sqrt{1-\text{VarGL}}},\left\{\frac{1}{\text{VarGL}}\right\}\right\}\right\},\right.\right.$$

$$\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},\mathbf{x}\right]\Bigg]$$

Out[23]=
$$\int_{\mathbf{x}}^{1}\frac{\int_{\mathbf{x}_1}^{1}\frac{\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\},\frac{\mathbf{x}_1}{\mathbf{x}_2}\right]}{\sqrt{-((\mathbf{x}_2-1)\mathbf{x}_2)}}d\mathbf{x}_2\int_{\frac{\mathbf{x}}{\mathbf{x}_1}}^{1}\frac{\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}\right\},\mathbf{x}_3\right)}{\sqrt{1-\mathbf{x}_3\left(\mathbf{x}_1\mathbf{x}_3^2+\mathbf{x}\mathbf{x}_3\right]}}d\mathbf{x}_3}{\mathbf{x}_1}d\mathbf{x}_1$$

In[24]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\left\{\frac{1}{\text{VarGL}+1},\{\}\right\},\left\{\frac{1}{\text{VarGL}\sqrt{1-\text{VarGL}}},\left\{\frac{1}{\text{VarGL}}\right\}\right\}\right\},\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},\mathbf{x}\right],\text{DefiniteIntegral}\to\text{True}\right]$$

Out[24]=
$$(\mathbf{x}-1)^3\int_0^1\frac{(\mathbf{x}_1-1)\mathbf{x}_1\int_0^1\frac{\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\},\left(\frac{\mathbf{x}}{(\mathbf{x}-1)\mathbf{x}_1+1}-1\right)\mathbf{x}_2+1\right]}{\sqrt{-((\mathbf{x}-1)\mathbf{x}(\mathbf{x}_1-1)(\mathbf{x}_2-1))}\left(\left(\frac{\mathbf{x}}{(\mathbf{x}-1)\mathbf{x}_1+1}-1\right)\mathbf{x}_2+1\right)}d\mathbf{x}_2}{((\mathbf{x}-1)\mathbf{x}_1+1)^2}$$
$$\int_0^1\frac{\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}\right\},\frac{(\mathbf{x}-1)\mathbf{x}_1+1}{(\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+1}\right]}{\sqrt{\frac{(\mathbf{x}-1)\mathbf{x}_1(\mathbf{x}_3-1)}{(\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+1}}((\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+2)}d\mathbf{x}_3d\mathbf{x}_1$$

In[25]:=
$$\text{GLCbToIntegral}\left[\text{GLCb}\left[\left\{\left\{\left\{\frac{1}{\text{VarGL}+1},\{\}\right\},\left\{\frac{1}{\text{VarGL}\sqrt{1-\text{VarGL}}},\left\{\frac{1}{\text{VarGL}}\right\}\right\}\right\},\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},\mathbf{x}\right],\text{DefiniteIntegral}\to\text{True},\text{UniqueIntegral}\to\text{True}\right]$$

Out[25]=
$$(\mathbf{x}-1)^3\int_0^1\int_0^1\int_0^1\frac{\text{GLb}\left[\left\{\frac{1}{2-\text{VarGL}}\right\},\left(\frac{\mathbf{x}}{(\mathbf{x}-1)\mathbf{x}_1+1}-1\right)\mathbf{x}_2+1\right]\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}\right\},\frac{(\mathbf{x}-1)\mathbf{x}_1+1}{(\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+1}\right]^{(\mathbf{x}_1-1)\mathbf{x}_1}}{((\mathbf{x}-1)\mathbf{x}_1+1)^2\sqrt{-((\mathbf{x}-1)\mathbf{x}(\mathbf{x}_1-1)(\mathbf{x}_2-1))}\left(\left(\frac{\mathbf{x}}{(\mathbf{x}-1)\mathbf{x}_1+1}-1\right)\mathbf{x}_2+1\right)}$$
$$\frac{1}{\sqrt{\frac{(\mathbf{x}-1)\mathbf{x}_1(\mathbf{x}_3-1)}{(\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+1}}((\mathbf{x}-1)\mathbf{x}_1\mathbf{x}_3+2)}d\mathbf{x}_3d\mathbf{x}_2d\mathbf{x}_1$$

A.1.9. GSToHSS

- **Input**: `GSToHSS[expr,(var)]`
- **Output**: `expr` rewritten using explicit sum representation
- **Options**: `var` (Symbol)
- **Details**: Function to convert a linear expression `expr` of `GS` sums to an `Harmonic- -SumsSum` (`HSS`) expression. If you give just one argument it will use a global τ as base index for the sums $(\tau_1, \tau_2 \ldots \tau_i \ldots)$, but you can also give the name of the letter `var` you wish to use as the second argument of the sum — we show an example of that usage in the fourth example below. We note that it can also rewrite doubly nested sums, i.e. , if we have `GS[{s1,...,sk,...,sq},n]` such that `sk` is itself a list, we interpret this

as "GS[{s1,...,GS[sk,var_k],...,sq},n]" — we show an example of that usage in the third example below.

- **Examples**:

In[26]:= $\text{GSToHSS}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}, \frac{\mathbf{2}^{\text{VarGL}}}{\text{VarGL}^{\mathbf{2}}}\right\}, \mathbf{n}\right]\right]$

Out[26]= $\sum_{\tau_1=1}^{\mathtt{n}} \binom{2\tau_1}{\tau_1} \left(\sum_{\tau_2=1}^{\tau_1} \frac{2^{\tau_2}}{\tau_2^2}\right)$

In[27]:= $\text{GSToHSS}\left[\text{GS}\left[\left\{\frac{\mathbf{1}}{\sqrt{\text{VarGL}^{\mathbf{2}}+\mathbf{1}}}, \frac{\mathbf{1}}{\mathbf{cos}(\text{VarGL})}, \sqrt{\text{VarGL}+\mathbf{1}}\right\}, \mathbf{n}\right]\right]$

Out[27]= $\sum_{\tau_1=1}^{\mathtt{n}} \frac{\sum_{\tau_2=1}^{\tau_1} \left(\sum_{\tau_3=1}^{\tau_2} \sqrt{\tau_3+1}\right) \sec(\tau_2)}{\sqrt{\tau_1^2+1}}$

In[28]:= $\text{GSToHSS}\left[\text{GS}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}+\mathbf{1}}, \left\{\frac{\left(\frac{\mathbf{1}}{\mathbf{4}}\right)^{\text{VarGL}}}{\text{VarGL}}, \frac{\mathbf{1}}{\text{VarGL}+\mathbf{2}}\right\}, \frac{\mathbf{1}}{\text{VarGL}+\mathbf{3}}\right\}, \mathbf{n}\right]\right]$

Out[28]= $\sum_{\tau_1=1}^{\mathtt{n}} \frac{\sum_{\tau_2=1}^{\tau_1} \left(\sum_{\tau_3=1}^{\tau_2} \frac{1}{\tau_3+3}\right) \left(\sum_{\tau_{2,1}=1}^{\tau_2} \frac{4^{-\tau_{2,1}} \left(\sum_{\tau_{2,2}=1}^{\tau_{2,1}} \frac{1}{\tau_{2,2}+2}\right)}{\tau_{2,1}}\right)}{\tau_1+1}$

In[29]:= $\text{GSToHSS}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}, \frac{\mathbf{2}^{\text{VarGL}}}{\text{VarGL}^{\mathbf{2}}}\right\}, \mathbf{n}\right], \mathbf{i}\right]$

Out[29]= $\sum_{\mathtt{i}_1=1}^{\mathtt{n}} \binom{2\mathtt{i}_1}{\mathtt{i}_1} \left(\sum_{\mathtt{i}_2=1}^{\mathtt{i}_1} \frac{2^{\mathtt{i}_2}}{\mathtt{i}_2^2}\right)$

A.1.10. ReduceGLCb

- **Input**: `ReduceGLCb[GLCb[args]]`
- **Output**: A simpler representation of `GLCb[args]`.
- **Options**: ∅
- **Details**: Given a GCPL function, attempts to simplify according to properties in Section 2.3.4.

- **Examples**:

In[30]:=
$$\mathbf{ReduceGLCb}\left[\mathbf{GLCb}\left[\left\{\frac{\mathbf{1}}{\mathbf{2}},\left\{\left\{\frac{\mathbf{1}}{\mathrm{VarGL}},\{\}\right\}\right\},\left\{\frac{\mathrm{VarGL}}{\sqrt{(1-\mathrm{VarGL})\mathrm{VarGL}}},\left\{\left\{\frac{\mathbf{1}}{\mathrm{VarGL}},\{\}\right\},\left\{\frac{\mathbf{1}}{\mathrm{VarGL}},\{\}\right\}\right\}\right\}\right\},\mathbf{x}\right]\right]$$

Out[30]=
$$\frac{\mathrm{GLb}\left[\left\{\frac{1}{\sqrt{(1-\mathrm{VarGL})\mathrm{VarGL}}},\frac{1}{\mathrm{VarGL}}\right\},\mathtt{x}\right]}{2\mathtt{x}}$$

In[31]:=
$$\mathbf{Reduce}\left[\mathbf{GLCb}\left[\left\{\left\{\left\{\frac{\mathbf{1}}{\sqrt{\mathrm{VarGL}+\mathbf{1}}},\{\}\right\},\left\{\frac{\mathbf{1}}{\mathrm{VarGL}+\mathbf{1}},\{\}\right\}\right\},\left\{\frac{\mathbf{1}}{\mathrm{VarGL}+\mathbf{1}}\left\{\left\{\frac{\mathbf{1}}{\mathrm{VarGL}},\{\}\right\},\left\{\frac{\mathbf{1}}{\mathrm{VarGL}+\mathbf{4}},\{\}\right\}\right\}\right\},\mathbf{x}\right]\right]$$

Out[31]=
$$\mathtt{GLCb}\left[\left\{\left\{\frac{1}{\mathrm{VarGL}},\left\{\frac{1}{\mathrm{VarGL}+1}\right\}\right\},\left\{\frac{1}{\mathrm{VarGL}+3}\left\{\frac{1}{\mathrm{VarGL}+4}\right\}\right\}\right\},\mathtt{x}\right]$$

A.1.11. TransformRootGL

- **Input**: `TransformRootGL[expr,x,(opts)]`
- **Output**: `expr` where all GL originally with root expressions are rewritten in simpler form if possible
- **Options**: `AsGL` (Boolean rule)
- **Details**: Given an expression containing GL functions of the following arguments: $\sqrt{1-x}, p-\sqrt{1-x}, \frac{1-\sqrt{1-x}}{1+\sqrt{1-x}}, (\frac{1-\sqrt{1-x}}{1+\sqrt{1-x}})^p$ with p$\in \mathbb{N}$, will linearize the expression to try to get only `GL` and `GLb` functions of `x`. The `opts` argument is `AsGL`, which by default is set to `False`. When it is set to `True`, the `ConvertGLbToGL` function in Appendix A.1.2 will be applied on the result so that it is expressed only with `GL` functions. This function is basically an extension of the `TransformGL` function of the `HarmonicSums` package by Jakob Ablinger to add some simple cases of square roots.
- **Examples**:

In[32]:=
$$\mathrm{TransformRootGL}\left[\mathrm{GL}\left[\left\{\frac{\mathbf{1}}{\mathrm{VarGL}+\mathbf{1}}\right\},\sqrt{\mathbf{1}-\mathbf{x}}\right],\mathbf{x}\right]$$

Out[32]=
$$\frac{1}{2}\left(\mathrm{GLb}\left[\left\{\frac{1}{\sqrt{1-\mathrm{VarGL}}\mathrm{VarGL}}\right\},\mathtt{x}\right]-\mathrm{GLb}\left[\left\{\frac{1}{\mathrm{VarGL}}\right\},\mathtt{x}\right]\right)$$

In[33]:=
$$\mathrm{TransformRootGL}\left[\mathrm{GL}\left[\left\{\frac{\mathbf{1}}{\mathrm{VarGL}}\right\},\frac{\mathbf{1}-\sqrt{\mathbf{1}-\mathbf{x}}}{\sqrt{\mathbf{1}-\mathbf{x}}+\mathbf{1}}\right],\mathbf{x}\right]$$

Out[33]= $\frac{1}{2}\left(-\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}\right\},\text{x}\right]-\text{GLb}\left[\left\{\frac{1}{\sqrt{1-\text{VarGL}}\text{VarGL}}\right\},\text{x}\right]\right)$
$+\frac{1}{2}\left(\text{GLb}\left[\left\{\frac{1}{\text{VarGL}}\right\},\text{x}\right]-\text{GLb}\left[\left\{\frac{1}{\sqrt{1-\text{VarGL}}\text{VarGL}}\right\},\text{x}\right]\right)$

In[34]:= $\text{TransformRootGL}\left[\text{GL}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}+\mathbf{1}}\right\},\frac{\mathbf{1}}{\mathbf{2}}-\sqrt{\mathbf{1}-\mathbf{x}}\right],\mathbf{x}\right]$

Out[34]= $\text{GL}\left[\left\{\frac{1}{\text{VarGL}^2+1}\right\},\frac{1}{2}\right]-2\text{GLb}\left[\left\{\frac{-9\sqrt{1-\text{VarGL}}+4\text{VarGL}\left(\sqrt{1-\text{VarGL}}+1\right)-4}{(\text{VarGL}-1)\left(16\text{VarGL}^2-56\text{VarGL}+65\right)}\right\},\text{x}\right]$

In[35]:= $\text{TransformRootGL}\left[\text{GL}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}+\mathbf{1}}\right\},\sqrt{\mathbf{1}-\mathbf{x}}\right],\mathbf{x},\text{AsGL}\to\text{True}\right]$

Out[35]= $\frac{1}{2}\left(\text{GL}\left[\left\{\frac{1}{\text{VarGL}}\right\},\text{x}\right]-\text{GL}\left[\left\{\frac{\sqrt{1-\text{VarGL}}}{\text{VarGL}}\right\},\text{x}\right]+\text{GL}\left[\left\{\frac{\sqrt{\text{VarGL}}}{1-\text{VarGL}}\right\},1\right]+2\sqrt{1-\text{x}}\right)$

A.2. Mellin convolutions

A.2.1. ExtendedMellinConvolve

- **Input**: `ExtendedMellinConvolve[f,g,x,y,(opts)]`
- **Output**: Computes the Mellin convolution of f and g in x, with y being the integration variable.
- **Options**: `PlusPrescription` (boolean flag), `AdditionalAssumptions` (set of conditions or `True`)
- **Details**: Attempts to compute the Mellin convolution as defined in Section 2.1.1. As opposed to Mathematica's `MellinConvolve`, it uses the definition of Mellin transform in Definition 2.1.8 and not the standard one. If `PlusPrescription` is set to `True` we consider that f is the function in +-prescription and compute the convolution accordingly (default value is `False`). `AdditionalAssumptions` can be any set of conditions on f, g, x, for example any bounds on x.
- **Examples**:

In[36]:= $\text{ExtendedMellinConvolve}\left[\frac{\mathbf{1}}{\sqrt{\mathbf{x}}},\frac{\mathbf{1}}{\mathbf{x}+\mathbf{1}},\mathbf{x},\mathbf{y}\right]$

Out[36]= $\frac{\pi-4\tan^{-1}\left(\sqrt{\text{x}}\right)}{2\sqrt{\text{x}}}$

In[37]:= $\text{ExtendedMellinConvolve}\left[\frac{\mathbf{1}}{\mathbf{x}-\mathbf{1}},\frac{\mathbf{1}}{\mathbf{x}+\mathbf{1}},\mathbf{x},\mathbf{y},\text{PlusPrescription}\to\text{True}\right]$

Out[37]= $$-\frac{\log(1-x)}{x+1}-\frac{\log\left(\frac{x+1}{2x}\right)}{x+1}$$

A.2.2. SumToMellin

- **Input**: `SumToMellin[expr,C,x,(opts)]`
- **Output**: Linear combination of `Mellin[a^nx^n-1,f_i[x]]` objects
- **Options**: `ToGLbBasis` (Boolean rule), `AllowSymbolicValues` (Boolean rule), `CustomMellinConvolution` (Boolean rule), `UseGLCb` (Boolean rule)
- **Details**: Computes the Mellin inversion of the `expr`, written as a `GS[args,n]` function expression, where the args is a list of functions in the `VarGL` variable. The algorithm will try to express the result as much as possible in terms of `Hwb` functions, and in the case there are some convolutions that did not work, will also use `GLCb` functions. See Section 2.3 for more details. The option `ToGLBBasis`, when set to True, uses shuffle algebra relations implemented in Ablinger's `HarmonicSums` [9,10,12,72–74] to try to simplify GRPLs appearing in the Mellin inverse (`True` is the default value). If `AllowSymbolicValues` is set to `True`, symbolic values stay untouched (`False` is the default value). If set to `True`, `CustomMellinConvolution` uses an internal Mellin convolution function instead of relying to the one used by Mathematica, which might be a bit more efficient in some cases[123]. (`False` is the default value). `UseGLCb`, when set to `True`, skips all the other convolution method and forces the result to be represented using GCPLs (`False` is the default value).
- **Examples**:

In[38]:= $$\text{SumToMellin}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}},\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}}\right\},\mathbf{n}\right],\mathbf{C},\mathbf{x}\right]$$

Out[38]= $$\left\{\frac{2\text{Mellin}\left[4^n x^n-1,\frac{\sqrt{x}\left(2z2-\text{Hwb}(\{fw1\},x)^2\right)}{\sqrt{1-x}(4x-1)}\right]}{\pi},\{\}\right\}$$

In[39]:= $$\text{SumToMellin}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}},\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}}\right\},\mathbf{n}\right],\mathbf{C},\mathbf{x},\text{ToGLbBasis}\to\text{False}\right]$$

[123] Note that this applies only in the case when other methods did not follow through and we do not want to use GCPLs, which is usually still the prefered option.

Out[39]= $\left\{-\frac{4\text{Mellin}\left[4^{\mathtt{n}}\mathtt{x}^{\mathtt{n}}-1,\frac{\sqrt{\mathtt{x}}\text{Hwb}(\{\text{fw1},\text{fw1}\},\mathtt{x})-\sqrt{\mathtt{x}}\mathtt{z2}}{\sqrt{1-\mathtt{x}}(4\mathtt{x}-1)}\right]}{\pi},\{\}\right\}$

In[40]:= $\text{SumToMellin}\left[\text{GS}\left[\left\{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}},\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}}\right\},\mathbf{n}\right],\mathbf{C},\mathbf{x},\text{UseGLCb}\to\text{True}\right]$

Out[40]= $\left\{\frac{4\text{Mellin}\left[4^{\mathtt{n}}\mathtt{x}^{\mathtt{n}}-1,\frac{\mathtt{x}\sqrt{1-\mathtt{x}}\mathtt{GLCb}\left[\left\{\left\{\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}},\{\}\right\},\left\{\frac{1}{\text{VarGL}-1},\left\{\frac{1}{\text{VarGL}}\right\}\right\}\right\},\mathtt{x}\right]+\sqrt{\mathtt{x}}\mathtt{z2}}{\sqrt{1-\mathtt{x}}(4\mathtt{x}-1)}\right]}{\pi},\{\}\right\}$

A.3. Asymptotic expansions

A.3.1. AsymptoticsMellint

- **Input**: `AsymptoticMellint[expr,x,n,ord,(opts)]`
- **Output**: Asymptotic expansion up to order `ord` as `n`$\to+\infty$ of `expr`, a linear combination of Mellin transforms of the form `Mellin[a^nx^n,f[x]]` or `Mellin[a^nx^n,f[x]]`
- **Options**: `ComputeConstants` (Boolean rule), `TimeLimit` (Positive real number), `ExtendedAsymptotics` (Boolean rule), `AdvancedIntegration` (Boolean rule), `ConvergenceTesting` (Boolean rule)
- **Details**: Computes the asymptotic expansion up to order `ord` as $n\to+\infty$ of `expr` that depends on `x`. `expr` should be represented using the `Mellin` function from `HarmonicSums`, i.e.,

$$\sum_{i=1}^{k}\alpha_i\text{Mellin}\left[(a_ix)^n-1,f_i(x)\right]\quad\text{with }\alpha_i,a_i\in\mathbb{R}^*.$$

See Section 2.5 for more details. The option `ComputeConstants` when set to `True` attemps to explicitely compute possible non-GCPL constants appearing in the expression as integrals (default value `True`). `TimeLimit` gives the time limit in seconds one allows for constant computations or attempts trying simpler schemes, like factorial series (Section 2.5.1) or geometric series (Section 2.5.2), before shifting to the change of variable approach (Section 2.5.3) (default value is 10). `ExtendedAsymptotics`, when set to `True`, allows for computation of Asymptotics to be done for all the sums, i.e., the full information will be given without suppressing exponentially subdominant parts of the expansion (default value is `False`). `AdvancedIntegration`, when set to `True`, corresponds to the special integration scheme described in Remark 2.5.8

(default value is `True`). `ConvergenceTesting` enforces the checks of Theorem 2.4.5 in Section 2.4.2 to warranty the expansions of GCPLs (default value is `True`).

- **Examples**:

In[41]:=
$$\text{AsymptoticsMellint}\left[-\frac{4\text{Mellin}\left[4^{n}x^{n}-1,\frac{\sqrt{x}\text{Hwb}(\{\text{fw1},\text{fw1}\},x)-\sqrt{x}z2}{\sqrt{1-x}(4x-1)}\right]}{\pi},x,n,4\right]$$

Out[41]=
$$\frac{\pi^{3/2}2^{2n+1}}{9\sqrt{n}}+\frac{(\pi-12)(12+\pi)2^{2n-2}}{27\sqrt{\pi}n^{3/2}}+\frac{(288+59\pi^{2})\,2^{2n-6}}{27\sqrt{\pi}n^{5/2}}+\frac{97\,(25\pi^{2}-432)\,2^{2n-9}}{81\sqrt{\pi}n^{7/2}}$$

In[42]:=
$$\text{AsymptoticsMellint}\left[\text{Mellin}\left[4^{n}x^{n}-1,\text{Hwb}[\{\text{fw1},\text{fw2}\},x]\right]+\text{Log}[2]\text{Mellin}\left[2^{n}x^{n}-1,\text{Hwb}[\{\text{fw3},\text{fw1}\},x]\right],x,n,6\right]$$

Out[42]=
$$\frac{\sqrt{\pi}2^{2n-\frac{3}{2}}}{n^{5/2}}-\frac{49\sqrt{\pi}2^{2n-\frac{11}{2}}}{n^{7/2}}+\frac{827\sqrt{\pi}2^{2n-\frac{17}{2}}}{n^{9/2}}-\frac{46799\sqrt{\pi}2^{2n-\frac{27}{2}}}{n^{11/2}}$$

In[43]:=
$$\text{AsymptoticsMellint}\left[\text{Mellin}\left[4^{n}x^{n}-1,\text{Hwb}[\{\text{fw1},\text{fw2}\},x]\right]+\text{Log}[2]\text{Mellin}\left[2^{n}x^{n}-1,\text{Hwb}[\{\text{fw3},\text{fw1}\},x]\right],x,n,6,\text{ExtendedAsymptotics}\to\text{True}\right]$$

Out[43]=
$$\frac{\sqrt{\pi}2^{2n-\frac{3}{2}}}{n^{5/2}}-\frac{49\sqrt{\pi}2^{2n-\frac{11}{2}}}{n^{7/2}}+\frac{827\sqrt{\pi}2^{2n-\frac{17}{2}}}{n^{9/2}}-\frac{46799\sqrt{\pi}2^{2n-\frac{27}{2}}}{n^{11/2}}$$
$$\log(2)\left(\frac{\sqrt{\pi}2^{n}}{n^{3/2}}-\frac{11\sqrt{\pi}2^{n-3}}{n^{5/2}}+\frac{201\sqrt{\pi}2^{n-7}}{n^{7/2}}-\frac{1713\sqrt{\pi}2^{n-10}}{n^{9/2}}+\frac{56475\sqrt{\pi}2^{n-15}}{n^{11/2}}\right.$$
$$\left.+\int_{-2\log 2}^{\infty}\frac{1}{2}e^{-w}\text{GL}\left[\left\{\frac{1}{(1-\text{VarGL})\sqrt{\text{VarGL}}}\right\},1-\frac{e^{-w}}{2}\right]dw\right)$$
$$+\int_{-2\log 2}^{\infty}\frac{1}{4}e^{-w}\text{GL}\left[\left\{\frac{1}{\sqrt{(1-\text{VarGL})\text{VarGL}}},\frac{1}{\sqrt{(1-\text{VarGL})(2-\text{VarGL})}}\right\},1-\frac{e^{-w}}{4}\right]dw$$

A.3.2. AsymptoticsSum

- **Input**: `AsymptoticsSum[expr,n,x,ord,(opts)]`
- **Output**: Asymptotic expansion up to order `ord` in `n` as `n`$\to+\infty$ of the nested sum `expr`, with outer bound `n`, `x` the integration variable of the intermediate Mellin transforms.
- **Options**: `ComputeConstants` (Boolean rule), `TimeLimit` (Positive real number), `ExtendedAsymptotics` (Boolean rule), `AdvancedIntegration` (Boolean rule), `ConvergenceTesting` (Boolean rule), `ToGLbBasis` (Boolean rule), `AllowSymbolicValues` (Boolean rule), `CustomMellinConvolution` (Boolean rule), `UseGLCb` (Boolean rule).
- **Details**: Combination of `SumToMellin` and `AsymptoticsMellint`, see their respective Sections A.2.2 and Section A.3.1 for more information and usage of the options.

- **Examples**:

In[44]:= $\text{AsymptoticsSum}\left[\text{GS}\left[\left\{\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}}\right\},\mathbf{n}\right],\mathbf{n},\mathbf{x},\mathbf{6}\right]$

Out[44]= $z2-\frac{1}{n}+\frac{1}{2n^2}-\frac{1}{6n^3}+\frac{1}{30n^5}$

In[45]:= $\text{AsymptoticsSum}\left[\text{GS}\left[\left\{\frac{\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}^{\mathbf{2}}}{\text{VarGL}^{\mathbf{2}}}\right\},\mathbf{n}\right],\mathbf{n},\mathbf{x},\mathbf{6}\right]$

Out[45]= $\frac{2^{4n+4}}{15\pi n^3}-\frac{2^{4n+2}}{75\pi n^4}+\frac{1003\ 2^{4n-1}}{1125\pi n^5}-\frac{22477\ 2^{4n-3}}{3375\pi n^6}$

In[46]:= $\text{AsymptoticsSum}\left[\text{GS}\left[\left\{\frac{e^{\text{VarGL}}}{\text{VarGL}},\frac{\mathbf{1}}{\text{VarGL}^{\mathbf{2}}},\binom{\mathbf{2}\text{VarGL}}{\text{VarGL}}\right\},\mathbf{n}\right],\mathbf{n},\mathbf{x},\mathbf{6}\right]$

Out[46]= $\frac{2^{2n+6}e^{n+1}}{9(4e-1)\sqrt{\pi}n^{7/2}}+\frac{7(3+4e)2^{2n+3}e^{n+1}}{9(4e-1)^2\sqrt{\pi}n^{9/2}}+\frac{(841+3352e+5392e^2)\,2^{2n-1}e^{n+1}}{9(4e-1)^3\sqrt{\pi}n^{11/2}}$

A.3.3. GLCbSeries

- **Input**: `GLCbSeries[GLCb[args,x],(opts)]`
- **Output**: Series expansion (`SeriesData` object) or `$Failed`
- **Options**: `ConvergenceTesting` (Boolean rule)
- **Details**: Given a GCPL in the form of `GLCb[args,x]`, attempts to compute its series expansion around $x = 1$. The option `ConvergenceTesting`, which is set to `True` by default, ensures that the Theorem 2.4.4 is checked and the procedure described in length in the Section 2.4 is applied. In the case the conditions are not verified, the computation is stopped and `$Failed` is returned together with an explicit error message — see the second example below for the illustration of this situation. Nevertheless, the user might choose to set `ConvergenceTesting` to `False`, and in this case the function will always compute a formal series expansion around $x = 1$, although without formal proof of existence outside of $x = 1$ — see the third example below.
- **Examples**:

In[47]:= $\text{GLCbSeries}\left[\text{GLCb}\left[\left\{\left\{\frac{\mathbf{1}}{\text{VarGL}+\mathbf{1}},\{\}\right\},\left\{\frac{\mathbf{1}}{\sqrt{\text{VarGL}(\mathbf{1}-\text{VarGL})}},\{\}\right\}\right\},\mathbf{x}\right],\mathbf{5}\right]$

Out[47]= $\frac{\sqrt{1-x}\sqrt{x-1}}{\sqrt{x-1}}+\frac{5(1-x)^{3/2}(x-1)^{3/2}}{6(x-1)^{3/2}}+\frac{27(1-x)^{5/2}(x-1)^{5/2}}{40(x-1)^{5/2}}+\frac{943(1-x)^{7/2}(x-1)^{7/2}}{1680(x-1)^{7/2}}$

$$+\frac{19417(1-x)^{9/2}(x-1)^{9/2}}{40320(x-1)^{9/2}}+0\left((x-1)^{11/2}\right)$$

In[48]:= GLCbSeries $\left[\text{GLCb}\left[\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},x\right],5\right]$

Out[48]= $Failed

In[49]:= GLCbSeries $\left[\text{GLCb}\left[\left\{\left\{\frac{1}{\text{VarGL}},\left\{\frac{1}{2-\text{VarGL}}\right\}\right\},\left\{\frac{1}{\sqrt{\text{VarGL}(1-\text{VarGL})}},\{\}\right\}\right\},x\right],5,\right.$
ConvergenceTesting → False]

Out[49]= $$\frac{4(1-x)^{3/2}(x-1)^{3/2}}{3(x-1)^{3/2}}+\frac{6(1-x)^{5/2}(x-1)^{5/2}}{5(x-1)^{5/2}}+\frac{107(1-x)^{7/2}(x-1)^{7/2}}{70(x-1)^{7/2}}$$
$$+\frac{5563(1-x)^{9/2}(x-1)^{9/2}}{3780(x-1)^{9/2}}+0\left((x-1)^{11/2}\right)$$

A.4. Useful functions

A.4.1. FindPSLQRelations

- **Input**: `FindPSLQRelations[csts,p,(opts)]`
- **Output**: `{A,B}` where `A` is the list of constants from `csts` reduced modulo relations `B`, in the form of rules
- **Options**: `ShowSteps` (boolean rule), `KeepAllRelations` (boolean rule), `NumericalPrecision` (`False` or list $\{a_{\text{integerpositive}}, b_{\text{integerpositive}}, c_{\text{rules}}\}$)
- **Details**: Given a list of constants `csts` and an integer p `FindPSLQRelations` will use the Mathematica PSLQ function `FindIntegerNullVector` to find relations as much as possible between the constants, before reducing them to a set of rules where as much redundant information has been eliminated; the rules express the most 'complicated' constants as function of less complicated ones. The parameter `p` quantifies the 'sharpness' of the PSLQ result that we accept, i.e., at least how much zeroes we need in order for the relation to be deemed acceptable. The default value is $p = 1$. The option `ShowSteps`, if set to `True`, will detail the computation steps (default value is `False`). For `NumericalPrecision`, if set to `False`, `FindIntegerNullVector` is fed with the symbolic value of constants. Otherwise, the user has to give a tuple or triplet $\{A, B, (C)\}$ where A is the number of digits up to which we compute the constants, B the threshold norm of the PSLQ vector after which we consider the result to be wrong, and C is an optional list of rules giving the precomputed numerical values of some constants (in the case those were computed beforehand by different methods). Concerning B, more precisely, if

numerical values are given, `FindIntegerNullVector` will always return some result, and in the case it is wrong it will give usually a linear combination with rather huge coefficients; we can then compute the norm of it and if it is too big we consider it's a failure. For `KeepAllRelations`, when set to `False`, once the set of PSLQ relations has been found using `FindIntegerNullVector`, it will attempt to tidy it up using Mathematica's `Simplify`, which means some relations of the form $a_1C_1 + ... + a_nC_n==0$ (with C_i the constants and a_i the PSLQ coefficients) might reduce to 0, which means the relation is already known to Mathematica (in this case, it will simply be deleted from the set of relations we dealt with). Otherwise, if set to `True`, we simplify the expression naively, and keep all relations.

- **Example**: We define below a list of constant for which we want to find relations using PSLQ:

In[50]:=
$$\mathbf{listConst} = \left\{\mathbf{\log(2)}, \mathbf{\sqrt{2}}, \mathbf{\log\left(\sqrt{2}+2\right)}, \mathbf{\pi^2}, \mathbf{\log^2(2)}, \mathbf{\sqrt{2}\pi^2}, \mathbf{\sqrt{2}\log^2(2)},\right.$$
$$\mathbf{\sqrt{2}\log(2)\log\left(2-\sqrt{2}\right)}, \mathbf{\log^2\left(2-\sqrt{2}\right)}, \mathbf{\sqrt{2}\log^2\left(2-\sqrt{2}\right)}, \mathbf{\log(2)\log\left(\sqrt{2}+2\right)},$$
$$\mathbf{\sqrt{2}\log(2)\log\left(\sqrt{2}+2\right)}, \mathbf{\log^2\left(\sqrt{2}+2\right)}, \mathbf{\sqrt{2}\log^2\left(\sqrt{2}+2\right)}, \mathbf{\mathrm{Li}_2\left(\frac{1}{4}\left(2-\sqrt{2}\right)\right)},$$
$$\mathbf{\sqrt{2}\mathrm{Li}_2\left(\frac{1}{4}\left(2-\sqrt{2}\right)\right)}, \mathbf{\mathrm{Li}_2\left(\frac{1}{4}\left(\sqrt{2}+2\right)\right)}, \mathbf{\sqrt{2}\mathrm{Li}_2\left(\frac{1}{4}\left(\sqrt{2}+2\right)\right)}, \mathbf{\sqrt{2}\log(2)},$$
$$\mathbf{\mathrm{Li}_2\left(-\sqrt{2}-1\right)}, \mathbf{\sqrt{2}\mathrm{Li}_2\left(-\sqrt{2}-1\right)}, \mathbf{\log(2)\log\left(2-\sqrt{2}\right)}, \mathbf{\mathrm{Li}_2\left(\sqrt{2}-1\right)},$$
$$\left.\mathbf{\sqrt{2}\mathrm{Li}_2\left(\sqrt{2}-1\right)}, \mathbf{\sqrt{2}\log\left(\sqrt{2}+2\right)}\right\}$$

Out[50]=
$$\left\{\left\{\sqrt{2}, \pi^2, \sqrt{2}\pi^2, \log(2), \sqrt{2}\log(2), \log^2(2), \sqrt{2}\log^2(2), \log\left(\sqrt{2}+2\right), \sqrt{2}\log\left(\sqrt{2}+2\right),\right.\right.$$
$$\log^2\left(\sqrt{2}+2\right), \sqrt{2}\log^2\left(\sqrt{2}+2\right), \mathrm{Li}_2\left(\sqrt{2}-1\right), \sqrt{2}\mathrm{Li}_2\left(\sqrt{2}-1\right), \log(2)\log\left(\sqrt{2}+2\right),$$
$$\left.\sqrt{2}\log(2)\log\left(\sqrt{2}+2\right)\right\},$$
$$\left\{\log(2)\log\left(2-\sqrt{2}\right) \to \log^2(2) - \log(2)\log\left(\sqrt{2}+2\right),\right.$$
$$\log^2\left(2-\sqrt{2}\right) \to \log^2(2) + \log^2\left(\sqrt{2}+2\right) - 2\log\left(\sqrt{2}+2\right)\log(2),$$
$$\mathrm{Li}_2\left(-\sqrt{2}-1\right) \to -\mathrm{Li}_2\left(\sqrt{2}-1\right) - \frac{\pi^2}{24} - \frac{1}{4}\log^2(2) - \log^2\left(\sqrt{2}+2\right) + \log\left(\sqrt{2}+2\right)\log(2),$$
$$\mathrm{Li}_2\left(\frac{1}{2}+\frac{1}{2\sqrt{2}}\right) \to 6\mathrm{Li}_2\left(\sqrt{2}-1\right) - \frac{7\pi^2}{24} + \frac{\log^2(2)}{2} + \frac{7}{2}\log^2\left(\sqrt{2}+2\right)$$
$$-5\log\left(\sqrt{2}+2\right)\log(2),$$
$$\mathrm{Li}_2\left(\frac{1}{2}-\frac{1}{2\sqrt{2}}\right) \to -6\mathrm{Li}_2\left(\sqrt{2}-1\right) + \frac{11\pi^2}{24} - \frac{1}{2}5\log^2(2) - \frac{5}{2}\log^2\left(\sqrt{2}+2\right)$$
$$\left.\left.+4\log\left(\sqrt{2}+2\right)\log(2)\right\}\right\}$$

A.4.2. GLCbToLatex

- **Input**: `GLCbToLatex[GLCb[args]]`

- **Output**: String corresponding to LaTeX code
- **Options**:∅
- **Details**: Writes the LaTeX expression of `GLCb[args]`
- **Example**:

In[51]:= **GLCbToLatex** $\left[\textbf{GLCb}\left[\left\{\left\{\frac{1}{\text{VarGL}},\{\}\right\},\left\{\frac{1}{\text{VarGL}},\{\}\right\}\right\},\mathbf{x}\right]\right]$

Out[51]=

```
GLCb\left[\left\{\left\{\frac{1}{\text{VarGL}}
,\left\{\right\}\right\},
\left\{\frac{1}{\text{VarGL}},
\left\{\right\}\right\}\right\},x\right]
```

A.4.3. VerifyExpansion

- **Input**: `VerifyExpansion[sum,expr,n,(opts)]`
- **Output**: Respectivelly absolute and relative error for the expansion `expr` of the sum `sum` as n goes to $+\infty$.
- **Options**: `ShowSteps` (Boolean rule).
- **Details**: Computes numerically the relative and absolute error for each term of the expansion `expr` of a sum `sum` and returns at the end the full error. If the option `ShowSteps` is set to `True`, it also shows the intermediate steps. The order of the expansion is identified automatically. See more details in Section 2.5.5.
- **Examples**: See examples in Section 2.5.5.

B. Implementation of the analysis of differential systems

The full analysis described in Chapter 4 to find the optimal orders of the ϵ-expansions of the integrals arising in the given coupled system has been implemented in the package called `SystemAnalysis`. Because we adopted a centralised approach, one single function, with many subroutines and many options, does the full computation. After describing its syntax, we show how it works on some examples.

B.1. Using SystemAnalysis

First and foremost, as `SystemAnalysis` relies on several packages, one has to load them beforehand:

```
In[1]:= << Sigma.m;
```

```
Sigma - A summation package by Carsten Schneider – © RISC
```

```
In[2]:= << HarmonicSums.m;
```

```
HarmonicSums by Jakob Ablinger – © RISC
```

```
In[3]:= << SumProduction.m;
```

```
SumProduction package by Carsten Schneider – © RISC
```

```
In[4]:= << EvaluateMultiSums.m;
```

```
EvaluateMultiSums package by Carsten Schneider – © RISC
```

```
In[5]:= << OreSysG.m;
```

```
OreSys Package by Stefan Gerhold
```

```
In[6]:= << SolveCoupledSystem.m;
```

```
SolveCoupledSystem package by Carsten Schneider – © RISC
```

and then our package can be loaded also

```
In[7]:= << SystemAnalysis.m;
```

```
SystemAnalysis package by Nikolai Fadeev – © RISC
```

The main function we will be using is called `analysisSystem`, and is called in the

following way:

```
analysisSystem[sys,x,H,J,W,R,ord,d,Ep,(options)]
```

where the arguments are:

- `sys`: This is the system we want to analyse, it has the form

$$\left\{\begin{aligned}
\alpha_1(x,d)\frac{\mathrm{d}}{\mathrm{d}x}H[a_1^{(1)},\dots,a_{k_1}^{(1)}][x] ==& \sum_{i=1}^{n}\beta_{1,i}(x,d)H[a_1^{(i)},\dots,a_{k_i}^{(i)}][x]\\
&+\sum_{j=1}^{q}\gamma_{1,j}(x,d)H[b_1^{(j)},\dots,b_{p_j}^{(j)}][x],\\
\alpha_2(x,d)\frac{\mathrm{d}}{\mathrm{d}x}H[a_1^{(2)},\dots,a_{k_2}^{(2)}][x] ==& \sum_{i=1}^{n}\beta_{2,i}(x,d)H[a_1^{(i)},\dots,a_{k_i}^{(i)}][x]\\
&+\sum_{j=1}^{q}\gamma_{2,j}(x,d)H[b_1^{(j)},\dots,b_{p_j}^{(j)}][x],\\
&\vdots\\
\alpha_n(x,d)\frac{\mathrm{d}}{\mathrm{d}x}H[a_1^{(n)},\dots,a_{k_n}^{(n)}](x) ==& \sum_{i=1}^{n}\beta_{n,i}(x,d)H[a_1^{(n)},\dots,a_{k_i}^{(n)}][x]\\
&+\sum_{j=1}^{q}\gamma_{n,j}(x,d)H[b_1^{(j)},\dots,b_{p_j}^{(j)}][x]
\end{aligned}\right\} \tag{B.1.1}$$

  where the $a_i^{(j)}, b_i^{(j)} \in \mathbb{N}$, $\alpha_i, \beta_{i,j}, \gamma_{i,j} \in \mathbb{K}(x,d)$, for $i \in \{1,\dots,n\}$ the functions $H[a_1^{(i)},\dots,a_{k_i}^{(i)}](x)$ being the MIs and for $j \in \{1,\dots,q\}$ the $H[a_1^{(j)},\dots,a_{p_j}^{(j)}](x)$ being the BIs.

- `x` is the variable in which we differentiate the system, as seen above in (B.1.1).

- `H` is the general name of the MIs and BIs, because frequently when we get examples from physics, they will be of the form

$$H[\delta_1,\dots,\delta_p][x], \tag{B.1.2}$$

  with the the δ_i being integers. Most frequently H = HFORMXYZT where XYZT is a sequence of letters and numbers describing the type and loop order of the computation we are doing.

- `J` is the name of the variable we will use to rename our MIs and BIs internally.

- `W` is the name for the dummy variables during the uncoupling.

- `R` denotes the RHS vector[124].
- `ord` is the list of the ϵ-orders up to which we want to solve the system, in the form, using the convention of (B.1.1):

$$\left\{\left\{H[a_1^{(1)},\ldots,a_{k_1}^{(1)}](x),o_1\right\},\ldots,\left\{H[a_1^{(n)},\ldots,a_{k_n}^{(n)}](x),o_n\right\}\right\} \tag{B.1.3}$$

  with the o_i being integers. We note the existence of a small helper function called

$$\texttt{getMI[sys,ord,x]}, \tag{B.1.4}$$

  which, given a system `sys` of the form (B.1.1), the order `ord` $\in \mathbb{Z}$ up to which we want to get all of the MIs and the variable `x` return such a list.
- `d` is the spacetime symbolic dimension
- `Ep` is ϵ, which we will use for the relation $d = 4 - \epsilon$ replacement[125].
- `Options` describes the possible additional options that we can give to the function. They will be all of the form

$$\texttt{"Options_name"} \to \texttt{value}.$$

  Here they are:
  - `"Speed"` : Takes three possible values, 0, 1 or 2. 0 corresponds to a case when we perform all of the possible uncouplings as described in Section 4.3, 1 to an intermediate case where after performing the first uncouplings we use the time it took to do them to put a time limit on the other uncouplings, and 2 is the most restrictive one as it launches the computation of Gauß and Zürcher uncouplings at the same time and returns only the first one that finishes.
  - `"Save"` : Takes two possible boolean values, `True` or `False` — if set to `True`, it will create intermediate save states at different steps, in particular of the uncouplings, so that if the computation is relaunched it will not start from the beginning again. The final solutions are also saved in a folder called `Output_sys`, where `sys` is the name of our system in Mathematica. We recommend to set this option to `True` as soon as your are doing long computations, as any crash or failure would make you loose any intermediate steps.

[124] Internally and to improve speed we replace the linearly combination of BIs in each equation by an abstract $R_i(x)$ element before uncoupling, and then only plugging back the actual value.

[125] We include both d and ϵ in the arguments because in particular, the user can give the original system already in ϵ and then this can be identified.

- `"Parallelization"` : Takes two possible boolean values, `True` or `False` — if set to `True`, will try to parallelise uncouplings.
- `"Scheme"` : Takes four possible values, which are `"Default"`, `"EpsOrder"`, `"DiffEqn"` and `"RecEqn"`. These correspond to the different functions described in Section 4.3.4, p.221. In particular:
  - `"Default"` corresponds to the case when we take
  $$f_1(\{-n_\epsilon, n_d, n_r\}) = -n_\epsilon + n_d + n_r.$$
  - `"EpsOrder"` corresponds to the case when we take
  $$f_2(\{-n_\epsilon, n_d, n_r\}) = -n_\epsilon.$$
  - `"DiffEqn"` corresponds to the case when we take to
  $$f_2(\{-n_\epsilon, n_d, n_r\}) = n_d.$$
  - `"RecEqn"` corresponds to the case when we take to
  $$f_2(\{-n_\epsilon, n_d, n_r\}) = n_r.$$

The output of `analysisSystem` takes the form

$$\Big\langle\Big|\text{MasterIntegralsUpdated} \to \{\ldots, \{I_i, o_i^{MI}, d_i\}, \ldots\}, \text{BaseIntegralsUpdated} \to \{\ldots, \{J_j, o_j^{BI}\}, \ldots\}, \text{Equations} \to \{\ldots, \{I_i, \text{LHS}_i == \text{RHS}_i\}, \ldots\}\Big|\Big\rangle \quad \text{(B.1.5)}$$

where the I_i represent the MIs, the o_i^{MI} the ϵ-order up to which we have to correct them and d_i the order of the HODE we will need to solve for them[126], J_j are the BIs and o_j^{BI} the order in ϵ up to which they need to be given, and the $\text{LHS}_i == \text{RHS}_i$ represent either the HODEs or LFs.

B.2. Examples

Because most examples come from physics and are of a really big size, we will show here some small examples, and will mostly cut off the equations which are usually too long. We also will use here fixed options for `analysisSystem`, in particular `"Scheme"` to `"Default"`, `"Parallelization"` to `False` and `"Speed"` to `1`.

[126] 0 meaning it is a LF and not a HODE.

Example B.2.1. *We consider the following system:*

$$
\begin{aligned}
-\frac{x^2 J(1)'(x)}{(x-1)(x+1)} =& \frac{(d-4)(3d-10)J(2)(x)x^2}{2(d-3)^2(x-1)^2(x+1)^2} + \frac{(d-2)(3d-8)\left(11d^2-74d+124\right)J(6)(x)x^2}{16(d-3)^2(2d-7)(x-1)^2(x+1)^2} \\
&- \frac{(3d-8)\left(dx^2-2x^2-10dx+32x+d-2\right)J(7)(x)x^2}{4(d-3)^2(x-1)^2(x+1)^4} \\
&- \frac{(2d-5)(3d-8)\left(dx^2-4x^2+10dx-32x+d-4\right)J(4)(x)x^2}{8(d-3)^2(x-1)^2(x+1)^4} \\
&+ \frac{\left(15x^2d^2+8xd^2+15d^2-98x^2d-48xd-98d+160x^2+72x+160\right)J(3)(x)x}{4(d-3)^2(x-1)^2(x+1)^2} \\
&- \frac{\left(2dx^2-7x^2+2dx-6x+2d-7\right)J(1)(x)x}{(x-1)^2(x+1)^2} \\
&+ \frac{(d-4)x\left(3dx^4-12x^4+4dx^3-12x^3-38dx^2+120x^2+4dx-12x+3d-12\right)J(5)(x)}{8(d-3)^2(x-1)^2(x+1)^4}, \\
-\frac{x^2 J(2)'(x)}{(x-1)(x+1)} =& \frac{x\left(3dx^2-10x^2-10dx+36x+3d-10\right)J(1)(x)(d-3)^2}{(3d-10)(x-1)^2(x+1)^2} \\
&+ \frac{3(3d-8)x^2\left(3dx^2-10x^2-22dx+76x+3d-10\right)J(7)(x)(d-3)}{(d-4)(3d-10)(x-1)^2(x+1)^4} \\
&- \frac{(2d-5)(3d-8)x^2\left(3dx^2-10x^2-22dx+76x+3d-10\right)J(4)(x)(d-3)}{(d-4)(3d-10)(x-1)^2(x+1)^4} \\
&+ \frac{(2d-7)x\left(x^2+1\right)J(2)(x)}{(x-1)^2(x+1)^2} \\
&+ \frac{x\left(3d^2x^4-16dx^4+20x^4-94d^2x^3+606dx^3-972x^3+86d^2x^2-556dx^2\right)J(5)(x)}{2(3d-10)(x-1)^2(x+1)^4} \\
&+ \frac{x\left(896x^2-94d^2x+606dx-972x+3d^2-16d+20\right)J(5)(x)}{2(3d-10)(x-1)^2(x+1)^4} \\
&- \frac{x\left(3x^2d^2-16xd^2+3d^2-19x^2d+110xd-19d+30x^2-188x+30\right)J(3)(x)}{(3d-10)(x-1)^2(x+1)^2}, \\
-\frac{x^2 J(3)'(x)}{(x-1)(x+1)} =& -\frac{xJ(1)(x)(d-3)^2}{(x-1)^2} + \frac{2x\left(x^2+x+1\right)J(3)(x)(d-3)}{(x-1)^2(x+1)^2} \\
&+ \frac{(d-4)xJ(5)(x)}{(x+1)^2} + \frac{(d-2)(3d-8)x^2J(6)(x)}{4(x-1)^2(x+1)^2} + \frac{(3d-8)x^2J(7)(x)}{(x-1)^2(x+1)^2} \\
&- \frac{(2d-5)(3d-8)x^2J(4)(x)}{2(x-1)^2(x+1)^2}
\end{aligned}
$$

We suppose that it is called `example1` *in Mathematica. Here* J_i, $i = 1, \ldots, 3$ *correspond to MIs and* J_i, $j = 4, \ldots, 7$ *correspond to BIs. Suppose we want to compute the MIs up to order 5 in* ϵ. *Then, we can first use* `getMI` *to write the vector of* (J_i, o_i), $i = 1, \ldots, 3$*:*

```
In[8]:= miOrd = getMI[example1, 5, x]

Out[8]= {{J[1][x], 5}, {J[2][x], 5}, {J[3][x], 5}}
```

Now, we call the `analysisSystem` *function in the following way:*

```
In[9]:= resAnaSys1 = analysisSystem[example1, x, J, P, W, R, miOrd, d, Ep, "Save" →
        True,
        "Parallelization" → False, "Speed" → 1]
```

```
Out[9]= "MasterIntegralsUpdated" → {{J[1][x], 5, 3}, {J[2][x], 5, 1}, {J[3][x], 5, 0}},
        "BaseIntegralsUpdated" → {{J[4][x], 5}, {J[5][x], 5}, {J[6][x], 5}, {J[7][x], 5}},
        "Equations" → {{{J[1][x], -((-1+x)(1+x)(-40+46(4+Ep)-17(4+Ep)^2+2(4+Ep)^3+
        184x-176(4+Ep)x+56(4+Ep)^2x-6(4+Ep)^3x-40x^2+46(4+Ep)x^2-17(4+Ep)^2x^2+
        2(4+Ep)^3x^2)J[1][x]) + ...,}}}
```

In this case, we see that we have to give all the BIs up to order 5*, and for the relations for* J_1 *and* J_2 *we will respectively need to solve a HODE of order* 3 *and another of order* 1*.*

Example B.2.2. *Let us now take a system of the following form, involving the HFORM physical variables instead of the* J*. We do not write it fully but it has the form*

$$
\begin{aligned}
&-\frac{x^2\,HFORM3l1a'_{(0,1,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{(x-1)(x+1)}\\
&\quad=\frac{(2d-5)(3d-8)x\left(3dx^2-4dx+3d-10x^2+12x-10\right)}{48(d-3)^2(x-1)^2(x+1)^2}\\
&\quad\times HFORM3l1a_{(0,0,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)+\cdots,\\
&-\frac{x^2\,HFORM3l1a'_{(0,1,0,1,1,0,0,1,2,0,0,0,0,0,0,0,0,0,0,0)}(x)}{(x-1)(x+1)}\\
&\quad=\frac{(2d-5)(3d-8)(5d-18)x^2\,HFORM3l1a_{(0,0,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{12(d-4)(x-1)^2(x+1)^2}+\cdots,\\
&-\frac{x^2\,HFORM3l1a'_{(0,1,0,2,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{(x-1)(x+1)}\\
&\quad=-\frac{2(d-3)^2x^2\,HFORM3l1a_{(0,1,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{(x-1)^2(x+1)^2}+\cdots,\\
&-\frac{x^2\,HFORM3l1a'_{(0,2,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{(x-1)(x+1)}\\
&\quad=-\frac{(2d-5)(3d-10)(3d-8)x\left(x^2+1\right)HFORM3l1a_{(0,0,0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0)}(x)}{24(d-3)(x-1)^2(x+1)^2}-\cdots
\end{aligned}
$$

It is a system of order 4*, we call it* `example2` *in Mathematica, and exactly like in the previous case, we define the row of (master integrals, orders):*

```
In[10]:= miOrd2 = getMI[example2, 5, x]

Out[10]= {{HFORM3l1a[0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5},
          {HFORM3l1a[0, 1, 0, 1, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5},
          {HFORM3l1a[0, 1, 0, 2, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5},
          {HFORM3l1a[0, 2, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5}}
```

Now we can run `analysisSystem`*, and we get*

```
In[11]:= resAnaSys2 = analysisSystem[example32x, HFORM3l1a, P, W, R, miOrd2, d, Ep,
          "Save" → True, "Parallelization" → False, "Speed" → 1]

Out[11]= "MasterIntegralsUpdated" →
          {{HFORM3l1a[0, 1, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5, 0},
          {HFORM3l1a[0, 1, 0, 1, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5, 0},
```

```
{HFORM3l1a[0, 1, 0, 2, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5, 4},
{HFORM3l1a[0, 2, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5, 0}} ,
"BaseIntegralsUpdated" →
{{HFORM3l1a[0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5},
{HFORM3l1a[1, 0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0][x], 5}} ,
"Equations" → ···
```

We see here that we have only two BIs and they should both be given up to order 5 *in* ϵ*. There is only one HODE we need to solve to obtain the full set of solutions.*

Example B.2.3. *The systems above were irreducible, let us also show an example of an irreducible system. Because it takes a lot of space we just precise here its characteristics but do not print it out: it is a system of order* 5 *that has* 35 *BIs and decomposes into two irreducible systems, one of order* 2 *and the other of order* 3*. We will call the MIs with* J_i*,* $i = 1, \ldots, 5$ *and the BIs with* J_j*,* $j = 5, \ldots, 40$*. We suppose it is contained in* `example3`*. We compute the row of orders again:*

```
In[12]:= miOrd3 = getMI[example3, 5, x]

Out[12]= {{J[1][x], 5}, {J[2][x], 5}, {J[3][x], 5}, {J[4][x], 5}, {J[5][x], 5}}
```

Then we can use `analysisSystem`*:*

```
In[13]:= resAnaSys3 = analysisSystem[example3, x, HFORM3l1a, J, W, R, miOrd3, d, Ep,
           "Save" → True, "Parallelization" → False, "Speed" → 1]

Out[13]= "MasterIntegralsUpdated" →
         {{J[1][x], 6, 2}, {J[2][x], 5, 1}, {J[3][x], 5, 0}, {J[4][x], 5, 3}, {J[5][x], 5, 1}},
         "BaseIntegralsUpdated" →
         {{J[6][x], 7}, {J[7][x], 5}, {J[8][x], 7}, ..., {J[39][x], 6}, {J[40][x], 5}},
         "Equations" → ···
```

The effects of the combined ϵ*-order/degree of the HODE/recurrence start to kick in, because the proposed solution requires to go up to order* 6*, which means that the proposed HODEs or the associated recurrences will be in counterpart of smaller degree.*

Bibliography

[1] C. Schneider. *Symbolic summation in difference fields*. PhD thesis, RISC, Johannes Kepler University, Linz, 2001.

[2] C. Schneider. A collection of denominator bounds to solve parameterized linear difference equations in ΠΣ-extensions. *An. Univ. Timişoara Ser. Mat.-Inform.*, 42(2):163–179, 2004.

[3] C. Schneider. Solving parameterized linear difference equations in terms of indefinite nested sums and products. *Journal of Difference Equations and Applications*, 11(9):799–821, 2005.

[4] C. Schneider. Degree bounds to find polynomial solutions of parameterized linear difference equations in ΠΣ-fields. *Applicable Algebra in Engineering, Communication and Computing*, 16(1):1–32, 2005.

[5] C. Schneider. Simplifying sums in ΠΣ-extensions. *J. Algebra Appl.*, 6(3):415–441, 2007.

[6] S. A. Abramov, M. Bronstein, M. Petkovšek, and C. Schneider. On rational and hypergeometric solutions of linear ordinary difference equations in ΠΣ∗-field extensions. *J. Symb. Comput.*, 107:23–66, 2021.

[7] J. Blümlein and S. Kurth. Harmonic sums and Mellin transforms up to two loop order. *Phys. Rev. D*, 60:014018, 1999.

[8] E. Remiddi and J. A. M. Vermaseren. Harmonic polylogarithms. *Int. J. Mod. Phys. A*, 15:725–754, 2000.

[9] J. Ablinger. *Computer algebra algorithms for special functions in particle physics*. PhD thesis, RISC, Johannes Kepler University, Linz, Linz, Austria, April 2012. Available at `http://www.risc.jku.at/publications/download/risc_4542/ablingerphdthesis.pdf`.

[10] J. Ablinger, J. Blümlein, and C. Schneider. Harmonic sums and polylogarithms generated by cyclotomic polynomials. *J. Math. Phys.*, 52:102301, 2011.

[11] J. Ablinger, J. Blümlein, and C. Schneider. Generalized harmonic, cyclotomic, and binomial sums, their polylogarithms and special numbers. *J. Phys. Conf. Ser.*, 523:012060, 2014.

[12] J. Ablinger, J. Blümlein, and C. Schneider. Analytic and algorithmic aspects of generalized harmonic sums and polylogarithms. *J. Math. Phys.*, 54:082301, 2013.

[13] A. Behring, J. Blümlein, and K. Schönwald. The inverse Mellin transform via analytic continuation. *JHEP*, 06:062, 2023.

[14] S. Moch, P. Uwer, and S. Weinzierl. Nested sums, expansion of transcendental functions and multiscale multiloop integrals. *J. Math. Phys.*, 43:3363–3386, 2002.

[15] M.Y. Kalmykov and O. Veretin. Single-scale diagrams and multiple binomial sums. *Physics Letters B*, 483(1):315–323, 2000.

[16] J. Ablinger, J. Blümlein, C. G. Raab, and C. Schneider. Iterated binomial sums and their associated iterated integrals. *J. Math. Phys.*, 55:112301, 2014.

[17] J. Ablinger. Inverse Mellin transform of holonomic sequences. *PoS*, LL2016:067, 2016.

[18] J. Ablinger. Computing the inverse Mellin transform of holonomic sequences using Kovacic's algorithm. *PoS*, RADCOR2017:001, 2018.

[19] J. Ablinger, J. Blümlein, C. G. Raab, and C. Schneider. Nested (inverse) binomial sums and new iterated integrals for massive Feynman diagrams. *PoS*, LL2014:020, 2014.

[20] J. Fleischer, A.V. Kotikov, and O.L. Veretin. Analytic two-loop results for self-energy- and vertex-type diagrams with one non-zero mass. *Nuclear Physics B*, 547(1):343–374, 1999.

[21] U. Aglietti and R. Bonciani. Master integrals with 2 and 3 massive propagators for the 2-loop electroweak form factor—planar case. *Nuclear Physics B*, 698(1):277–318, 2004.

[22] J. Ablinger, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, M. Round, and C. Schneider. The $O(\alpha_s^3 T_F^2)$ contributions to the gluonic operator matrix element. *Nucl. Phys. B*, 885:280–317, 2014.

[23] J. Ablinger, J. Blümlein, C. Raab, C. Schneider, and F. Wißbrock. Calculating massive 3-loop graphs for operator matrix elements by the method of hyperlogarithms. *Nucl. Phys. B*, 885:409–447, 2014.

[24] C. Krattenthaler and C. Schneider. Evaluation of binomial double sums involving absolute values, 2020.

[25] C. Schneider. Simplifying multiple sums in difference fields. In *Computer algebra in quantum field theory*, Texts Monogr. Symbol. Comput., pages 325–360. Springer, Vienna, 2013.

[26] K. M. Borwein, D. J. Broadhurst, and J. Kamnitzer. Central binomial sums, multiple Clausen values and Zeta values. *Exper. Math.*, 10:25–34, 2001.

[27] R. Osburn and C. Schneider. Gaussian hypergeometric series and supercongruences. *Mathematics of Computation*, 78(265):275–275, 2009.

[28] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. Goedicke, A. von Manteuffel, C. Schneider, and K. Schönwald. The unpolarized and polarized single-mass three-loop heavy flavor operator matrix elements $\mathrm{A}_{gg,Q}$ and $\Delta\mathrm{A}_{gg,Q}$. *JHEP*, 12:134, 2022.

[29] S. Weinzierl. Expansion around half integer values, binomial sums and inverse binomial sums. *J. Math. Phys.*, 45:2656–2673, 2004.

[30] G. N. Watson. The harmonic functions associated with the parabolic cylinder. *Proceedings of the London Mathematical Society*, s2-17(1):116–148, 1918.

[31] J. Wimp and D. Zeilberger. Resurrecting the asymptotics of linear recurrences. *Journal of Mathematical Analysis and Applications*, 111(1):162–176, 1985.

[32] M. Kauers. A Mathematica package for computing asymptotic expansions of solutions of P-finite recurrence equations. *RISC Technical Report 11-04.*, 2011.

[33] H. Ferguson and R. Forcade. Generalization of the Euclidean algorithm for real numbers to all dimensions higher than two. *Bulletin of the American Mathematical Society*, 1:912–914, 1979.

[34] D. H. Bailey and D. J. Broadhurst. Parallel integer relation detection: techniques and applications. *Math. Comput.*, 70:1719–1736, 2001.

[35] K.G. Chetyrkin and F.V. Tkachov. Integration by parts: The algorithm to calculate β-functions in 4 loops. *Nuclear Physics B*, 192(1):159–204, 1981.

[36] C.F. Gauss. *Theoria attractionis corporum sphaeroidicorum ellipticorum homogeneorum methodo nova tractata.* Königliche Gesellschaft der Wissenschaften, 1813.

[37] G. Green. Essay on the mathematical theory of electricity and magnetism. *Essay on the Mathematical Theory of Electricity and Magnetism*, pages 1–115, 1828.

[38] J. Lagrange. Nouvelles recherches sur la nature et la propagation du son, Miscellanea Taurinensis, t. II. *Oeuvres t.*, 1:263, 1760.

[39] S. Laporta. High-precision calculation of multiloop Feynman integrals by difference equations. *International Journal of Modern Physics A*, 15(32):5087–5159, 2000.

[40] A. von Manteuffel and C. Studerus. Reduze 2 - Distributed Feynman integral reduction. 1 2012.

[41] P. Marquard and D. Seidel. The package Crusher. unpublished.

[42] M. Ostrogradski. Théorie de la chaleur. *Mem.Ac.Sci.St.Peters.*, 6:129–133, 1831.

[43] J. Henn. Multiloop integrals in dimensional regularization made simple. *Phys. Rev. Lett.*, 110:251601, 2013.

[44] R. N. Lee. Reducing differential equations for multiloop master integrals. *JHEP*, 04:108, 2015.

[45] O. Gituliar and V. Magerya. Fuchsia : A tool for reducing differential equations for Feynman master integrals to ϵ-form. *Computer Physics Communications*, 219:329–338, 2017.

[46] M. Prausa. Epsilon : A tool to find a canonical basis of master integrals. *Computer Physics Communications*, 219:361–376, 2017.

[47] C. Meyer. Algorithmic transformation of multi-loop master integrals to a canonical basis with CANONICA. *Computer Physics Communications*, 222:295–312, 2018.

[48] J. Ablinger, J. Blümlein, P. Marquard, N. Rana, and C. Schneider. Automated solution of first order factorizable systems of differential equations in one variable. *Nucl. Phys. B*, 939:253–291, 2019.

[49] C. Schneider, J. Ablinger, A. Behring, J. Blümlein, and A. De Freitas. Algorithms to solve coupled differential systems in terms of power series. *PoS*, LL2016:005, 2016.

[50] C. Schneider, J. Blümlein, and P. Marquard. A refined machinery to calculate

large moments from coupled systems of linear differential equations. *PoS*, RADCOR2019:078, 2019.

[51] J. Ablinger, J. Blümlein, A. Freitas, A. Hasselhuhn, A. Manteuffel, M. Round, C. Schneider, and F. Wißbrock. The transition matrix element $A_{gq}(N)$ of the variable flavor number scheme at $O(\alpha_s^3)$. *Nuclear Physics B*, 882:263–288, 2014.

[52] A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The 3-loop non-singlet heavy flavor contributions to the structure function $g_1(x, Q^2)$ at large momentum transfer. *Nuclear Physics B*, 897:612–644, 2015.

[53] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The 3-loop pure singlet heavy flavor contributions to the structure function $F_2(x, Q^2)$ and the anomalous dimension. *Nuclear Physics B*, 890:48–151, 2015.

[54] A. Behring, J. Blümlein, A. De Freitas, A. Hasselhuhn, A. von Manteuffel, and C. Schneider. $O(\alpha_s^3)$ heavy flavor contributions to the charged current structure function $xF_3(x, Q^2)$ at large momentum transfer. *Phys. Rev. D*, 92(11):114005, 2015.

[55] A. Behring, J. Blümlein, G. Falcioni, A. De Freitas, A. von Manteuffel, and C. Schneider. Asymptotic 3-loop heavy flavor corrections to the charged current structure functions $F_L^{W^+-W^-}(x, Q^2)$ and $F_2^{W^+-W^-}(x, Q^2)$. *Phys. Rev. D*, 94(11):114006, 2016.

[56] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. Calculating three loop ladder and V-topologies for massive operator matrix elements by computer algebra. *Comput. Phys. Commun.*, 202:33–112, 2016.

[57] J. Ablinger, A. Behring, J. Blümlein, G. Falcioni, A. De Freitas, P. Marquard, N. Rana, and C. Schneider. Heavy quark form factors at two loops. *Phys. Rev. D*, 97(9):094022, 2018.

[58] J. Ablinger, J. Blümlein, A. De Freitas, C. Schneider, and K. Schönwald. The two-mass contribution to the three-loop pure singlet operator matrix element. *Nuclear Physics B*, 927:339–367, 2018.

[59] J. Blümlein, A. De Freitas, C. Schneider, and K. Schönwald. The variable flavor number scheme at next-to-leading order. *Physics Letters B*, 782:362–366, 2018.

[60] J. Ablinger, J. Blümlein, P. Marquard, N. Rana, and C. Schneider. Heavy quark form factors at three loops in the planar limit. *Physics Letters B*, 782:528–532, 2018.

[61] J. Blümlein and C. Schneider. The method of arbitrarily large moments to calculate single scale processes in quantum field theory. *Phys. Lett. B*, 771:31–36, 2017.

[62] J. Blümlein, P. Marquard, N. Rana, and C. Schneider. The heavy fermion contributions to the massive three loop form factors. *Nucl. Phys. B*, 949:114751, 2019.

[63] A. De Freitas, J. Blümlein, P. Marquard, and C. Schneider. Challenges for analytic calculations of the massive three-loop form factors. *PoS*, LL2024:031, 2024.

[64] J. Blümlein, A. De Freitas, M. Saragnese, C. Schneider, and K. Schönwald. Logarithmic contributions to the polarized $O(\alpha_s^3)$ asymptotic massive Wilson coefficients and operator matrix elements in deeply inelastic scattering. *Phys. Rev. D*, 104(3):034030, 2021.

[65] J. Blümlein, A. Hasselhuhn, S. Klein, and C. Schneider. The $O(\alpha_s^3 n_f T_F^2 C_{A,F})$ contributions to the gluonic massive operator matrix elements. *Nucl. Phys. B*, 866:196–211, 2013.

[66] J. Blümlein, P. Marquard, C. Schneider, and K. Schönwald. The massless three-loop Wilson coefficients for the deep-inelastic structure functions F_2, F_L, xF_3 and g_1. *JHEP*, 11:156, 2022.

[67] J. Blümlein, P. Marquard, C. Schneider, and K. Schönwald. The three-loop unpolarized and polarized non-singlet anomalous dimensions from off shell operator matrix elements. *Nuclear Physics B*, 971:115542, 2021.

[68] J. Blümlein, P. Marquard, C. Schneider, and K. Schönwald. The three-loop polarized singlet anomalous dimensions from off-shell operator matrix elements. *JHEP*, 01:193, 2022.

[69] J. Blümlein, N. Fadeev, and C. Schneider. Computing Mellin representations and asymptotics of nested binomial sums in a symbolic way: the RICA package. *ACM Commun. Comp. Alg.*, 57(2):31–34, 2023.

[70] J. A. M. Vermaseren. Harmonic sums, Mellin transforms and integrals. *Int. J. Mod. Phys. A*, 14:2037–2076, 1999.

[71] J. Blümlein. Large acale analytic calculations in quantum field theories. In *Algorithmic combinatorics: Enumerative combinatorics, apecial functions and computer algebra: In honour of Peter Paule on his 60th birthday*, pages 63–87. Springer International Publishing, Cham, 2020.

[72] C. Schneider, J. Ablinger, J Blümlein, and A. De Freitas. A toolbox to solve coupled systems of differential and difference equations. *PoS*, RADCOR2015:060, 2016.

[73] J. Ablinger. The package HarmonicSums: computer algebra and analytic aspects of nested sums. *PoS*, LL2014:019, 2014.

[74] J. Ablinger. A computer algebra toolbox for harmonic sums related to particle physics. Master's thesis, Linz U., 2009.

[75] N. Nielsen. *Handbuch der Theorie der Gammafunktion*. B.G.Teubner, Leipzig, 1906.

[76] E. Landau. Über die Grundlagen der Theorie der Fakultätenreihen. XXXVI, Heft I:151–218, 1906.

[77] R. Argand. *Essai sur une manière de représenter les quantités imaginaires dans les constructions géométriques*. Gauthier-Villars, Paris,1874, 1806,1813, 2nd. Edition.

[78] J. Ablinger, J. Blümlein, and C. Schneider. Iterated integrals over letters induced by quadratic forms. *Phys. Rev. D*, 103(9):096025, 2021.

[79] J. Blümlein, M. Saragnese, and C. Schneider. Hypergeometric structures in Feynman integrals. *Ann. Math. Artif. Intell.*, 91(5):591–649, 2023.

[80] J.-M. Nicolas and R. Badeau. Histoire de la transformée de Mellin. In *Colloque GRETSI*, Actes du XXVIème Colloque GRETSI, Juan-Les-Pins, France, September 2017.

[81] J. Blümlein. Leading log radiative corrections to deep inelastic neutral and charged current scattering at HERA. *Z. Phys. C*, 47:89–94, 1990.

[82] A. H. Zemanian. *Distribution theory and transform analysis: an introduction to generalized functions, with applications*. Dover Publications, Inc., USA, 1987.

[83] L. Schwartz. *Théorie des distributions Vol 1,2*. Hermann,Paris, 1950,1951.

[84] J. Hadamard. *Lectures on Cauchy's problem in linear partial differential equations*. Dover Phoenix editions (Dover Publications, New York), 1923.

[85] M. E. Hoffman. Quasi-shuffle products. *Journal of Algebraic Combinatorics*, 11(1):49–68, 2000.

[86] P. L. Chebyshev. Théorie des mécanismes connus sous le nom de parallélogrammes. *Mémoires des Savants étrangers présentés à l'Académie de Saint-Pétersbourg*, 7:539–586, 1854.

[87] M. Abramowitz and I. A. (Eds) Stegun. *Handbook of mathematical functions with formulas, graphs, and mathematical tables, 9th printing*. New York:Dover, 1972.

[88] E. W. Weisstein. Beta Function. `https://mathworld.wolfram.com/BetaFunction.html` [Accessed: 8/2/2024].

[89] J. Ablinger, J. Blümlein, A. De Freitas, M. van Hoeij, E. Imamoglu, C. G. Raab, C. S. Radu, and C. Schneider. Iterated elliptic and hypergeometric integrals for Feynman diagrams. *J. Math. Phys.*, 59(6):062305, 2018.

[90] C. Koutschan. *Advanced applications of the holonomic systems approach*. PhD Thesis, RISC, Johannes Kepler University, Linz, September 2009.

[91] C. Koutschan. A fast approach to creative telescoping. *Mathematics in Computer Science*, 4(2–3):259–266, 2010.

[92] H. R. P. Ferguson and D.H. Bailey. A polynomial time, numerically stable integer relation algorithm. *RNR Technical Report RNR-91-032*, 1992.

[93] J.J. Kovacic. An algorithm for solving second order linear homogeneous differential equations. *Journal of Symbolic Computation*, 2(1):3–43, 1986.

[94] C. Schneider. Symbolic summation assists combinatorics. *Sem.Lothar.Combin.*, 56:1–36, 2007.

[95] S.A. Abramov and M. Petkovšek. D'Alembertian solutions of linear differential and difference equations. *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, page 169–174, 1994.

[96] M. Bronstein. Linear ordinary differential equations: breaking through the order 2 barrier. In *Papers from the International Symposium on Symbolic and Algebraic Computation*, ISSAC '92, page 42–48, New York, NY, USA, 1992. Association for Computing Machinery.

[97] P.A. Hendricks and M.F. Singer. Solving difference equations in finite terms. *Journal of Symbolic Computation*, 27(3):239–259, 1999.

[98] M. Petkovšek. Hypergeometric solutions of linear recurrences with polynomial coefficients. *Journal of Symbolic Computation*, 14(2):243–264, 1992.

[99] F. Klein, E. Ritter, and O. Haupt. *Vorlesungen über die Hypergeometrische Funktion*. Grundlehren Der Mathematischen Wissenschaften (Springer Hardcover). Springer, Berlin, Germany, 1933.

[100] W.N. Bailey. *Generalized hypergeometric series*. Cambridge tracts in math. and math. physics. 32. Stechert-Hafner Service Agency, 1964.

[101] L.J. Slater. *Generalized hypergeometric sunctions*. Cambridge University Press, 1966.

[102] P. Appell and J.K. de Fériet. *Fonctions hypergéométriques et hypersphériques: polynomes d'Hermite*. Gauthier-Villars, 1926.

[103] Paul Appell. *Sur les fonctions hypergéométriques de plusieurs variables, les polynômes d'Hermite et autres fonctions sphériques dans l'hyperespace*. Gauthier-Villars, 1925.

[104] D. E. Knuth. *The art of computer programming, Vol. 1: Fundamental algorithms*. Addison-Wesley, Reading, Mass., third edition, 1997.

[105] T.M. Apostol. *Calculus, Volume 1*. John Wiley & Sons, Nashville, TN, 1991.

[106] N. Blestein and R.A. Handelsman. *Asymptotic expansions of integrals*. Dover Pub., New York, 1986.

[107] E. T. Bell. Partition polynomials. *Annals of Mathematics*, 29(1/4):38–46, 1927.

[108] E. T. Bell. Exponential polynomials. *Annals of Mathematics*, 35(2):258–277, 1934.

[109] E. Remiddi. Differential equations for Feynman graph amplitudes. *Nuovo Cim. A*, 110:1435–1452, 1997.

[110] A.V. Kotikov. Differential equations method. New technique for massive Feynman diagram calculation. *Physics Letters B*, 254(1):158–164, 1991.

[111] N. Fadeev. First-order factorizable systems of differential equations in one variable. *RISC Report Series*, (20-20), 2020.

[112] B. Zürcher. Rationale Normalformen von pseudo-linearen Abbildungen. Master's thesis, ETH Zürich, 1994.

[113] J. Blümlein, A. De Freitas, and C. Schneider. Recent symbolic summation methods to solve coupled systems of differential and difference equations. *PoS*, LL2014:017, 2014.

[114] A. Bostan, F. Chyzak, and É. de Panafieu. Complexity estimates for two uncoupling algorithms. *Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation*, pages 85–92, 2013.

[115] J. Blümlein. Algebraic relations between harmonic sums and associated quantities. *Comput. Phys. Commun.*, 159:19–54, 2004.

[116] S. Gerhold. *Uncoupling systems of linear operator equations*. PhD thesis, RISC, Johannes Kepler University, Linz, August 2002.

[117] M. Kauers. *D-finite functions*. Springer Cham, November 2023.

[118] C. Mallinger. *Algorithmic manipulations and transformations of univariate holonomic functions and sequences*. PhD thesis, RISC, Johannes Kepler University, Linz, August 1996.

[119] J. Blümlein, S. Klein, C. Schneider, and F. Stan. A symbolic summation approach to Feynman integral calculus. *Journal of Symbolic Computation*, 47(10):1267–1289, 2012.

[120] J. Ablinger, J. Blümlein, M. De Freitas, A.and Saragnese, C. Schneider, and K. Schönwald. New 2- and 3-loop heavy flavor corrections to unpolarized and polarized deep-inelastic scattering. *SciPost Phys. Proc.*, 8:137, 2022.

[121] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, and C. Schneider. The three-loop splitting functions $P_{qg}^{(2)}$ and $P_{gg}^{(2),n_f}$. *Nuclear Physics B*, 922:1–40, 2017.

[122] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, C. Schneider, and K. Schönwald. The three-loop single mass polarized pure singlet operator matrix element. *Nuclear Physics B*, 953:114945, 2020.

[123] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, C. Schneider, and K. Schönwald. The first–order factorizable contributions to the three–loop massive operator matrix elements $A_{Qg}^{(3)}$ and $\Delta A_{Qg}^{(3)}$. *Nuclear Physics B*, 999:116427, 2024.

[124] A. Behring, J. Blümlein, A. De Freitas, A. Goedicke, S. Klein, A. von Manteuffel, C. Schneider, and K. Schönwald. The polarized three-loop anomalous dimensions from on-shell massive operator matrix elements. *Nucl. Phys. B*, 948:114753, 2019.

[125] A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, K. Schönwald, and C. Schneider. The polarized transition matrix element $A_{gq}(N)$ of the variable flavor number scheme at $O(\alpha_s^3)$. *Nuclear Physics B*, 964:115331, 2021.

[126] J. Blümlein, A. Maier, P. Marquard, G. Schäfer, and C. Schneider. From momentum expansions to post-Minkowskian Hamiltonians by computer algebra algorithms. *Physics Letters B*, 801:135157, 2020.

[127] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel, C. Schneider, and K. Schönwald. The non-first-order-factorizable contributions to the three-loop single-mass operator matrix elements $A_{Qg}^{(3)}$ and $\Delta A_{Qg}^{(3)}$. *Phys. Lett. B*, 854:138713, 2024.