Prepared for submission to JHEP

BaikovLetter: On Symbol Letters from Gram Determinants

Xuhang Jiang,[a] **Jiahao Liu,**[a,b] **Xiaofeng Xu,**[c] **Li Lin Yang**[d]

[a] *CAS Key Laboratory of Theoretical Physics, Institute of Theoretical Physics, Chinese Academy of Sciences, Beijing 100190, China*

[b] *School of Physical Sciences, University of Chinese Academy of Sciences, No.19A Yuquan Road, Beijing 100049, China*

[d] *Zhejiang Institute of Modern Physics, School of Physics, Zhejiang University, Hangzhou 310027, China*

E-mail: xhjiang@itp.ac.cn, liujiahao@itp.ac.cn, third@one.univ, yanglilin@zju.edu.cn

Abstract: Abstract...

Contents

1 Introduction

In the last decade, significant progress has been made in the calculation of Feynman integrals with the (canonical) differential equation method [1–6]. The canonical differential equation (CDE) method, combined with integration by part (IBP) reduction [7–10], is highly powerful in solving multi-scale multi-loop Feynman integral families. One-loop multi-leg integral families [11–13] and numerous two-loop or higher-loop integral families in $d = 4-2\epsilon$ dimension have been calculated using this method. For example, the five-point integral families with massless [14–17], one massive external leg [18–21] or two external massive legs [22] have been studied. Partial results for massless two-loop six point integral families [23–26] and some three-loop four-point integral families [27–30] or even five-point integral families [31] have also been obtained.

The CDE method is particularly well-suited for solving a special class of iterated integrals known as multiple polylogarithms (MPL) [32–35]. MPL functions f can be graded by their transcendentality $\mathcal{T}(f)$, allowing us to study them based in their transcendentality weights. As a simple example, $\mathcal{T}(\log R) = 1, \mathcal{T}(\mathrm{Li}_2(z)) = 2$. More powerful tools, such as the symbol [36, 37] and coproducts [38], have been introduced to simplify these MPL functions by leveraging their algebraic properties. The symbol $\mathcal{S}(f)$ of an MPL function f is defined recursively as

$$\mathrm{d}f = \sum_i f_i \mathrm{d}\log R_i,\, \mathcal{S}(f) = \mathcal{S}(f_i) \otimes R_i,\, \mathcal{S}(\log R) = R, \tag{1.1}$$

where R_i is called the symbol letter of this MPL function. The symbol determines the function up to boundary values, and the symbol letters contain the singularity properties of the function.

The CDE method exploits the graded transcendentality structure and algebraic properties of MPL functions. The key idea of this method is to find a set of canonical master integrals $\mathbf{I}$ such that the differential equations for $\mathbf{I}$ satisfy the following canonical form

$$\mathrm{d}\mathbf{I} = \epsilon \sum A_i \mathrm{d}\log \alpha_i \,\mathbf{I}\,, \tag{1.2}$$

where α_i is free of ϵ and A_i is a matrix with rational numbers. Once the canonical differential equations are obtained, $\mathbf{I}$ can be solved iteratively, and the symbol structure of $\mathbf{I}$ is derived. The set $\{\alpha_i\}$ is called the symbol alphabet of this family. $\mathbf{I}$ satisfying (1.1) is also referred to as a uniformly transcendental (UT) basis of this family, as it is naturally graded by weight. If we define the weight of ϵ to be -1, then $\mathbf{I}$ is uniformly of weight 0. One major challenge in this method is constructing a UT basis for a specific problem. Many algorithms and public packages have been developed to search for such a UT basis [39–48]. These methods are primarily based on performing transformation of basis and bring differential equation matrix to the ϵ factorized form. However, when an integral family involves too many scales, finding such a transformation may be challenging. In such cases, alternative methods [16, 49–53] that search for $\mathrm{d}\log$ forms or calculate leading singularities at the integrand level of an integral family can be particularly helpful.

Typically, after obtaining a set of UT basis, we need to compute their partial derivatives and then reconstruct the total derivatives. There are two main challenges in this step. First, we need to compute partial derivatives with respect to all independent variables in the problem and reconstruct symbol letters from these partial derivatives. If there are many variables, this step can be redundant, as symbol alphabets typically have a simpler structure than their partial derivatives. The second challenge is that we need to perform IBP reduction analytically after computing the partial derivatives. This can be particularly challenging for complicated integral families. However, if the symbol alphabet of the family is known a priori, we can bypass these two challenges by performing IBP reduction numerically and directly bootstrap all coefficient matrices A_i before the symbol letter $\mathrm{d}\log\alpha_i$. Additionally, sometimes we only need to determine certain analytic properties of this family without solving the CDE. Therefore, significant efforts in the literature have been devoted to obtaining the symbol letters or even the symbol without explicitly calculating the CDE matrix. Symbol letters for one-loop families with arbitrary multiplicity have been well understood by various different methods [54–58], but general case for higher-loop families remains unknown. Most approaches to this problem involve relating symbol letters to Landau singularities [59–63], as symbol letters describe the locations of singularities, which can also be solved from Landau equations. Another powerful method [64–69] relates symbol letters to cross ratios constructed from the solutions of Schubert analysis. In this paper we study this problem within the Baikov representation and provide an ansatz for algebraic symbol letters in higher-loop Feynman integral families[1]. We have tested this ansatz on several complicated examples and successfully predicted the symbol alphabet (while also obtaining the analytic CDE) of pentagon box family with two massive external legs, which had not been previously studied in [70]. In this paper, we provide a detailed discussion of our algorithms and present additional examples to illustrate the method.

This paper is organized as follows. In section 2, we briefly review the Baikov representation and demonstrate how leading singularities are calculated from the Baikov representations of a Feynman integral family. They serve as candidates for rational letters appearing in the CDE. In section 3, we present our ansatz for algebraic letters in the integral family and demonstrate its application. In section 4, we provide examples to illustrate some nontrivial aspects of this method. We have also observed that this method for algebraic letters fails in a nonplanar integral family and analyzed the possible reasons for this failure. Finally, we conclude in section 5. Many important details and subtleties are left to appendix for readers interested in further insights. In appendix A, we analyze how 4d external kinematics may impact our analysis in six-point examples. In appendix B, we provide a simple tutorial for the packages `BaikovAll` and `BaikovLetter`. In particular, we describe in details our algorithms for calculating leading singularities and algebraic letters.

2 Singularities from Baikov representation and rational letters

The most familiar form of Feynman integrals in general space-time dimensions $d = 4 - 2\epsilon$ is the loop-momentum representation,

$$I[a_1, a_2, \ldots, a_n] = \int \prod_{i}^{L} \frac{\mathrm{d}^d l_i}{i\pi^{d/2}} \frac{N}{D_1^{a_1} D_2^{a_2} \ldots D_n^{a_n}} , \tag{2.1}$$

where $\{D_i\}$ is a complete set of inverse propagators and N is a possible numerator. When $a_i > 0$, D_i corresponds to a propagator and when $a_i \leq 0$, D_i is referred to as an irreducible scalar product (ISP). A set of inverse propagators with $a_i > 0$ specifies a sector of this integral family. We will use a list of number like $\{1, 3, 4, 5, 7\}$ to denote a sector with $a_{1,3,4,5,7} > 0$. Most analyses of the singularities of Feynman integrals are carried out in the loop-momentum representation or the Feynman parameterization representation. However, we will show that Baikov representations of a Feynman integral family are closely related to its symbol letter structure. Therefore, we will briefly review the structure of Baikov representation and then derive the Landau singularities of an integral family within this framework.

[1] Symbol letters for one-loop integral families can be reproduced using the same ansatz.

2.1 Brief review of structures of Baikov representation

The scalar Feynman integrals (2.1) can always be expressed in Baikov representation as

$$I[a_1, a_2, \ldots, a_n] = C \int \prod_i^n \mathrm{d}x_i \prod_j P_j(\boldsymbol{x})^{\gamma_j} \frac{N(\boldsymbol{x})}{x_1^{a_1} x_2^{a_2} \ldots x_n^{a_n}}, \tag{2.2}$$

where $\{D_i\}$ is renamed as $\boldsymbol{x} = \{x_i\}$ to simplify the notation. C is a constant involving kinematics variables and the dimension regulator ϵ. γ_j is some power in the form $a_j + b_j\epsilon$, where a_j can be integers or half-integers. $P_j(\boldsymbol{x})$ is a Baikov polynomial, which is essentially a Gram determinant of loop momenta and external momenta. $N(\boldsymbol{x})$ is a polynomial in $\boldsymbol{x}$. Baikov representation with the twist $P(\mathbf{x})^{\gamma_i}$ allows us to introduce a denominator $P(\mathbf{x})$ which corresponds to a shift in γ_i. For an integral family, there are many different Baikov representations (referred to as *loop-by-loop representations*) for all its sectors, and all of them can be derived from the standard representation of the top sector. We refer to this as the recursive structures of Baikov representation. Therefore, given an integral family, we can easily derive all Baikov Gram determinants involved in the family[2] and analyze singularities based on them. Here we illustrate this recursive structure in Fig. 1. As an example, we explicitly show the expressions of the representations relevant to the highlighted path in Fig. 1,

$$\begin{aligned} &\text{std repr. : } G(k_1, k_2, p_1, p_2)^{-1/2-\epsilon} \\ &\qquad\qquad \downarrow \text{integrating } k_1 \cdot p_1 \text{ out} \\ &G(k_2, p_2)^{-1/2+\epsilon} G(k_2, p_1, p_2)^{-\epsilon} G(k_1, k_2, p_2)^{-\epsilon} \\ &\qquad\qquad \downarrow \text{integrating } k_2 \cdot p_1 \text{ out} \\ &G(k_1, k_2, p_2)^{-\epsilon} \end{aligned} \tag{2.3}$$

where constant coefficients are omitted, and only Baikov polynomials are shown for simplicity. Here we define Gram determinants as

$$G(\{p_1, p_2, \cdots, p_n\}, \{q_1, q_2, \cdots, q_n\}) \equiv \det \begin{pmatrix} p_1 \cdot q_1 & p_1 \cdot q_2 & \cdots & p_1 \cdot q_n \\ p_2 \cdot q_1 & p_2 \cdot q_2 & \cdots & p_2 \cdot q_n \\ \vdots & \vdots & \ddots & \vdots \\ p_n \cdot q_1 & p_n \cdot q_2 & \cdots & p_n \cdot q_n \end{pmatrix}, \tag{2.4}$$

and $G(p_1, p_2, \cdots, p_n)$ is shorthand for $G(\{p_1, p_2, \cdots, p_n\}, \{p_1, p_2, \cdots, p_n\})$. These Gram determinants $G(\cdots)$ (i.e. $P_j(\boldsymbol{x})$ as defined earlier) are the building blocks for our construction of symbol letters since they characterize the singularity structure of an integral family, as we will see in the next section.

Finally, we delve a litter deeper into the relations between the standard representation and loop-by-loop representations derived from it. Let us again take (2.3) as an example. We can find that $G(k_2, p_1, p_2)G(k_1, k_2, p_2)$ is just the discriminant of $G(k_1, k_2, p_1, p_2)$ with respect to $k_1 \cdot p_1$ (where $G(k_1, k_2, p_1, p_2)$ is a quadratic polynomial of $k_1 \cdot p_1$). Therefore, if we have solutions of Baikov variables that annihilate this discriminant, the contour of

[2] For more details, we refer to ref. [71] and the package provided there.

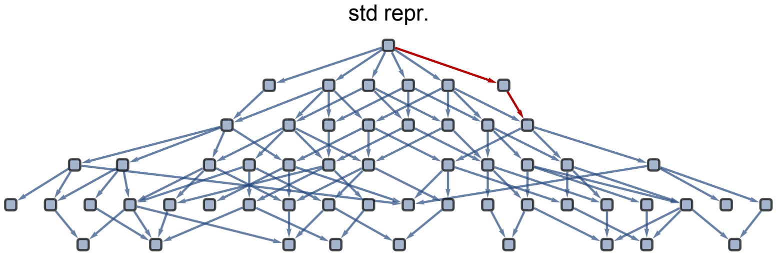

Figure 1. The relation graph between Baikov representations derived in a nonplanar triangle family. This is used to show the general structure of Baikov representation of a given integral family. Every node denotes a Baikov representation of some sector in this family and every arrow denotes integrating some variable out from the parent representation and arriving at the child representation. We start from the standard Baikov representation on the top and end with representations identified to be zero at the bottom. Explicit expressions for the path highlighted will be shown in (2.3). This structure depends on how the ISPs are defined in this family.

standard representation is pinched along the direction of $k_1 \cdot p_1$. Similarly, $G(k_2, p_2) = 0$ corresponds to the pinch at infinity, as $G(k_2, p_2)$ is the coefficient of quadratic term in $k_1 \cdot p_1$. Studying Landau singularities in the standard Baikov representations is typically challenging because, for subsectors, too many ISPs remain in the expressions after cutting all the propagators (these correspond to subleading Landau singularities for the top sector, but we will also refer to them as leading singularities of subsectors hereafter). So we study these singularities in the loop-by-loop representations, where some variables (which are ISPs) can be integrated out. This simplifies the analysis, as fewer integration variables remain in the expressions. However, for a specific sector in an integral family, there are usually multiple independent loop-by-loop representations, corresponding to integrating *different* Baikov variables out. Here, independence means that one representation can not be derived from the other by further integrating some variables out. Thus, if we have different solutions under these independent representations, they will describe the pinch of standard representation in different directions. In general, we need to consider the pinches in all directions as stated in [72]. Therefore, we recover full solutions for all Baikov variables in the standard representation (or some higher representations with more Baikov variables) by combining solutions from its different decendents. This is crucial for us to get the final expressions of algebraic letters. An example of this has been presented around (B.17) in App. B.3.

2.2 Singularities and rational letters from Baikov Gram determinants

Singularities of an integral typically arise from pinched poles or branch points of the integrand, and this is the standard approach to derive Landau singularities [73] of a Feynman integral (see textbook [74]). In addition to directly solving the Landau equations, recent years have seen numerous studies addressing this problem using principle A-determinants, Euler discriminants (principal Landau determinants) [63, 75–79] or Whitney stratifications [80], which provide a refined definition for these singularities using algebraic geometry. All these analyses usually start from the Feynman parameterization representation or its variants, such as Lee-Pomeransky representation [81]. There is also a powerful recursive method for calculating Landau singularities, proposed in [72]. This method is carried out in Baikov representations [82], which can be viewed as a variant of loop-momentum representation. In this article, we will use Baikov representations to calculate all the leading singularities[3] of an integral family, owing to the convenience of performing maximal cut in it, as well as the recursive structure in this representation of the whole integral family. Certainly, the most important motivation is that the Baikov representation allows us to address the algebraic letters in a natural way. Our analysis is not designed for a single integral, but for the entire integral families defined by the set of propagators. It is generally believed that the union of leading singularities from each sector will yield all the Landau singularities of the integral family.

Before delving into further details, let us be familiar with some notations and methods we use. In Baikov representation there are two different types of singularities at the *integrand* level. This depends on the exponent γ_j of $P_j(\mathbf{x})$ in (2.2). As mentioned earlier, $\gamma_j = a_j + b_j\epsilon$ where a_j can be an integer or a half integer. When a_j is an integer, $P(\mathbf{x}) = 0$ *can* appear as a pole at the integrand level for some x_j in the polynomial[4]. When a_j is a half integer, $P(\mathbf{x}) = 0$ appears as a branch point. This distinction is important, and we will classify these $P(\mathbf{x})$'s, which are Gram determinants in the Baikov representation, based on their powers. To analyze the leading singularities of a sector, we first perform the maximal cut of all propagators in this sector (which is trivially done in Baikov representation by setting a set of x_i to 0). The remaining integration variables are then all ISPs. Now, we will restrict ourselves to the special case where *leading singularities are calcuated by taking the multivariate residues of these ISPs*. In general, this is just a subset of all the Landau singularities of this family. The reason is clear: we only study the simple poles, and the pinches of polynomials under square roots are overlooked (See Fig. 2 for the distinction between the cyan and orange curves). This is because our main goal is to analyze integral families belonging to MPL, which have a well-defined symbol structure. The singularities associated with such integral families are simple poles (or $d\log$ forms).

The pinches of polynomials under square roots can be incorporated in a similar manner, and this will be described in more detail in App. B.2. Since our calculations are performed in d dimensional Baikov representations, where external legs are assumed to be general, a direct consequence of overlooking pinches of polynomials under square roots is that we may miss some leading singularities of MPL families under special kinematic constraints. This is because, under these constraints, polynomials under square roots may degenerate into perfect squares and thus become poles that need to be analyzed. For example, for external legs in fixed dimensions and degenerate kinematics, such as those integral families

[3] *leading singularities* here means the remaining kinematic coefficients of Feynman integral, which are the multivariate residues of the integrand around all the possible multivariate poles of the integration variables. They are related to the leading Landau singularities, as these multivariate poles can be viewed as the solutions of Landau equations with all the propagators cut.

[4] Of course, in dimensional regularization, the power will always be deformed by an infinitesimal complex variable ϵ. However, we will take ϵ to 0 in the end. It is really different whether the power is close to an integer or a half integer. This also results in the difference between integrals in even dimensions and odd dimensions.

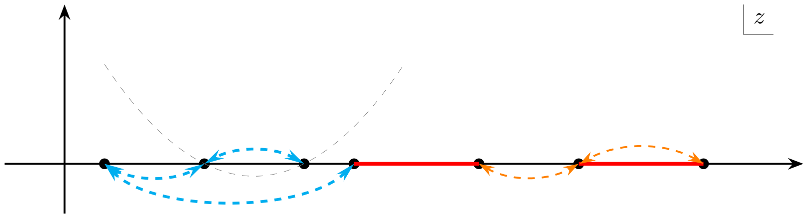

Figure 2. Here we show the pinches of poles or branch points for single integration variable z. The red lines state where the branch cuts lie, the cyan curves mean the possible pinch between simple poles or between simple poles and branch points and the orange curves denote the pinches between branch points. Only cyan curves are related to our analysis of MPL family since the singularities are always related to simple poles. The dashed parabola indicates that some poles are two roots of a quadratic polynomial and the pinch of them means the discriminant of this quadratic polynomial vanishes. All pinches can be described by the resultants of two polynomials.

in [24], we present an example of how new leading singularities can arise in our analysis when considering the four-dimensional limit in Baikov representations in App. A.

2.2.1 Residues and leading singularities

Let us begin with the simplest case, where there is only one integration variable z remaining. In this case, the definition of the residue is straightforward. We localize z to a possible pole by deforming the contour to a path that circles it. The residues correspond to the pinching of poles or branch points, as illutrated in Fig. 2.

We illustrate this with a simple example. Consider the following integral, which captures the basic character of a Baikov representation

$$\int \mathrm{d}z(z-a)^{\gamma_1}(z-b)^{\gamma_2}(Az^2+Bz+C)^{\gamma_3}\frac{1}{(z-a)(z-b)\sqrt{Az^2+Bz+C}}, \tag{2.5}$$

where $\gamma_1, \gamma_2, \gamma_3$ are complex regulators close to integers and a, b, A, B, C are all rational expressions of kinematic variables. There are two simple poles of z here. That is, $z = a$ and $z = b$. Localizing z to a, we obtain the residue, which consists of two factors

$$a-b,\ \sqrt{Aa^2+Ba+C}\,. \tag{2.6}$$

These two expressions correspond to the pinching of two poles between $z = a$ and $z = b$ and pinching a pole $z = a$ with the branch points determined by $\sqrt{Az^2+Bz+C}$ respectively. Localzing z to b, the factors of the residue will be

$$a-b,\ \sqrt{Ab^2+Bb+C}\,. \tag{2.7}$$

which has a similar interpretation as before. We will always neglect the negtive sign or rational number in our analysis, as they are not important for our purpose. Thus, we identify the following three polynomials which are leading singularities, as they represent pinch conditions for this integral.

$$L_1 \equiv Aa^2+Ba+C, \quad L_2 \equiv Ab^2+Bb+C, \quad L_3 \equiv a-b \tag{2.8}$$

For each leading singularity, we can find a corresponding $\mathrm{d}\log$ form. For example, $L_{1,2}$ correspond to the following two $\mathrm{d}\log$ forms:

$$D_1 = \frac{\sqrt{Aa^2+Ba+C}}{(z-a)\sqrt{Az^2+Bz+C}}\mathrm{d}z,\ D_2 = \frac{\sqrt{Ab^2+Bb+C}}{(z-b)\sqrt{Az^2+Bz+C}}\mathrm{d}z. \tag{2.9}$$

In addition to these two simple poles, there is also an infinity pole that must be considered. We will address it by promoting polynomials in Baikov representation to projective space (see also [83]),

$$z \to \frac{z}{x_0}, \quad [z:1] \to [z:x_0].$$

Similarly, for the multivariate case, we perform the following transformation:

$$x_i \to \frac{x_i}{x_0}, \quad [x_1 : x_2 : \ldots : x_n : 1] \to [x_1 : x_2 : \ldots : x_n : x_0]$$

In affine space, where $x_0 \neq 0$, we can return to the usual representation by scaling x_0 to 1. When $x_0 = 0$, we transition to the infinity plane. We analyze the singularities in the infinity plane by first homogenizing the Baikov polynomial and then setting $x_0 = 0$. In this simple example, we find in the infinity plane, the corresponding residue[5] is simply

$$\sqrt{L_4} \equiv \sqrt{A}. \tag{2.10}$$

For simplicity, We will denote this infinity pole as $z = \infty$.

Thus, we have the following three possible poles for z:

$$z = a, \quad z = b, \quad \text{or} \quad z = \infty\,. \tag{2.11}$$

The residues at these three poles provide all the leading singularities in this representation. Since $z-a$, $z-b$ and Az^2+Bz+C are all Gram determinants in Baikov representations, we establish a correspondence between Gram determinants and leading singularities. This correspondence is established by setting Baikov variables to the solutions obtained when calculating multivariate residues. These leading singularities, or residues, can be divided into two types: rational types like L_3, and square-root types, such as L_1, L_2, L_4, which are actually under square roots. If the expression under square root is a perfect square, we factor it out and treat it as a rational type. As emphasized earlier, the pinches of $Az^2 + Bz + C$ under square root are not considered by default, as they are not simple poles but branch cuts. Otherwise, the discriminant $B^2 - 4AC$ would also be included as a singularity.

The analysis for the multivariate case is similar. We choose an order of Baikov variables and perform the analysis sequentially. Thus at every step, it is essentially a single-variable problem. For more details and subtleties regarding how this is done for multivariate case, we refer to App. B.2, where a self-contained description of the algorithm is provided, along with the usage of package functions. Finally, all the leading singularities obtained for an integral family will serve as the candidates for rational letters.

[5] Note that in an integral family, we can freely introduce a numerator. Thus, although the infinity pole in above integral is not a simple pole, we can add a numerator z^2 to bring it into a simple pole.

2.2.2 From square-root type leading singularities to algebraic letters

In the previous section, we discussed how leading singularities are calculated from residues. In the process of calculating multivariate residues, we localize x_i to the corresponding poles, such as those in (2.11). These solutions, along with the cut condition for propagators, determine the Baikov variables in a given sector of an integral family. In this section, we mainly discuss how algebraic letters can arise from the perspective of leading singularity analysis, which is closely related to [69]. We will also discuss another perspective based on the structure of algebraic letters in the next section.

Let us first focus on how square-root leading singularities arise. In the analysis of leading singularities, we will keep track of the power of a Baikov polynomial, i.e., whether it is under a square root or not. Square-root type leading singularities are recorded in this manner. They also represent all possible square roots that may appear in an algebraic letter. These square roots have three different origins: the first is Baikov polynomials under square root after substituting solutions of Baikov variables into them (this corresponds to the pinch between poles and branch points). The second is from taking the discriminant of the quadratic polynomials in the intermediate step (this corresponds to the pinch of the quadratic polynomial itself). The third type is special: if the Baikov polynomial is pure kinematic (without Baikov variables) and has a half-integer power, it directly is a square-root leading singularity if it is not canceled by the first two types. It arises from the derivation of Baikov representations and describes the degeneracy of external kinematics. The third type may overlap with the first two, which is why it may be canceled.

Next, we elaborate on the role these solutions of Baikov variables play in algebraic letters related to square-root leading singularities. This is also closely related to our ansatz for algebraic letters, which will be discussed in the next section. The first type of square-root leading singularities corresponds to rational solutions of Baikov variables, as it involves the poles solved from linear polynomials. Examples can be found in (2.6) and (2.7). The second type arises naturally from the solutions of quadratic polynomials of Baikov variables. For example, consider the following quadratic polynomials of z (in Baikov representation, polynomials of Baikov variables are typically related to Gram determinants, which is why we can write it in the following way)

$$
\begin{aligned}
G(q_1,q_2,q_3,q_4) &= \left|\begin{array}{c|cc|c} q_1^2 & q_1\cdot q_2 & q_1\cdot q_3 & z \\ \hline q_2\cdot q_1 & q_2^2 & q_2\cdot q_3 & q_2\cdot q_4 \\ q_3\cdot q_1 & q_3\cdot q_2 & q_3^2 & q_3\cdot q_4 \\ \hline z & q_4\cdot q_2 & q_4\cdot q_3 & q_4^2 \end{array}\right| \\
&= G(q_1,q_2)z^2 + 2\,G(\{q_1,q_2,q_3\},\{q_1,q_2,q_4\})|_{z=0}\,z + G(q_1,q_2,q_3,q_4)|_{z=0} = 0,
\end{aligned}
\tag{2.12}
$$

The ratio of solutions for z will take the form

$$
\frac{r_1}{r_2} = \left.\frac{-G(\{q_1,q_2,q_3\},\{q_1,q_2,q_4\}) + \sqrt{G(q_1,q_2,q_3)G(q_1,q_2,q_4)}}{-G(\{q_1,q_2,q_3\},\{q_1,q_2,q_4\}) - \sqrt{G(q_1,q_2,q_3)G(q_1,q_2,q_4)}}\right|_{z=0}.
\tag{2.13}
$$

Here, we have used

$$
\frac{\Delta}{4} = G^2(\{q_1,q_2,q_3\},\{q_1,q_2,q_4\}) - G(q_1,q_2)G(q_1,q_2,q_3,q_4) = G(q_1,q_2,q_3)G(q_1,q_2,q_4),
$$

where Δ is the discriminant. We use the ratio of two roots because $\log r_1/r_2$ has definite parity when the sign of square roots is changed, and it indeed describes the singularity when r_1 and r_2 are pinched. This serves as a good ansatz for algebraic letters. The constraint $z = 0$ is also important, as it indicates that when Baikov variable z is integrated out, it should be set to 0 for the ansatz constructed. The corresponding square root is a leading singularity because

$$\frac{\sqrt{G(q_1, q_2, q_3)G(q_1, q_2, q_4)}\, \mathrm{d}z}{G(q_1, q_2)z^2 + 2G(\{q_1, q_2, q_3\}, \{q_1, q_2, q_4\})z + G(q_1, q_2, q_3, q_4)} = \mathrm{d}\log\frac{z - r_1}{z - r_2}. \tag{2.14}$$

This also explains why Gram determinants that lie in the same path of Baikov representations are needed in the later construction rule for algebraic letters: Gram determinants in the same path are directly related by integrating some variable z out, where the discriminants Δ naturally arise (see (2.3) for example). Thus, we can directly construct algebraic letters like (2.13), involving square roots $\sqrt{G(q_1, q_2, q_3)G(q_1, q_2, q_4)}$, in our ansatz instead of solving quadratic polynomials. Note that in the multivariate case, $\log r_1/r_2$ may still involve unsolved Baikov variables, as q_i can also involve loop momenta. In this case, we must substitute the rational solutions of remaining Baikov variables (linear poles) into above expression to obtain the final result for algebraic letters. The third type of square roots can also be incorporated into our ansatz for algebraic letters. This corresponds to the case where q_i in above Grams are (linear combinations of) external momenta without loop momenta.

The above relations between square-root leading singularities and algebraic letters are actually not exhaustive. For example, there is also another kind of relation related to the "mixing" between two solutions from the same representation. This picture aligns with [69] in general but is described using different language here. Such relations will be described by another type of ansatz in the next section. However, the above analysis do provide stong motivation to directly construct algebraic letters from the Baikov Gram determinants. In summary, we only need to retain the rational solutions of Baikov variables (linear poles) when calculating the multivariate residues. The square roots will be taken care of by the direct ansatz for them.

3 Constructing algebraic letters from Baikov Gram determinants

Before presenting the explicit forms of the algebraic ansatz for general higher-loop algebraic letters, we can gain valuable insights from the study of one-loop algebraic symbol letters [54–58]. For general kinematics, we can derive all one-loop symbol letters and write them as expressions of minors of some Gram determinants [68, 71]. It is particularly enlightening to express the one-loop algebraic symbol letters in the following forms:

$$\begin{aligned} A_1:\ \log\frac{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) + \sqrt{-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}}{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) - \sqrt{-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}}\Bigg|_{\text{maximal cut}} \\ A_2:\ \log\frac{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) + \sqrt{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_i\})G(\{\boldsymbol{q}, q_j\}, \{\boldsymbol{q}, q_j\})}}{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) - \sqrt{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_i\})G(\{\boldsymbol{q}, q_j\}, \{\boldsymbol{q}, q_j\})}}\Bigg|_{\text{maximal cut}} \end{aligned} \tag{3.1}$$

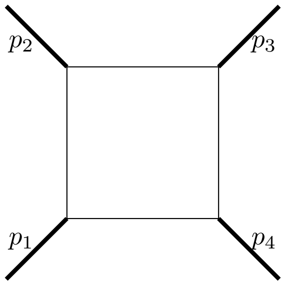

Figure 3. One-loop four-mass box. External legs are all massive with four different masses $p_i^2 = m_i^2$. Internal lines are all massless.

where $\boldsymbol{q}$ is a list of momenta and $q_i, q_j \notin \boldsymbol{q}$. Each q can be loop momentum or external momentum. $G(\{\boldsymbol{q}\},\{\boldsymbol{q}\})$, $G(\{\boldsymbol{q}, q_j\},\{\boldsymbol{q}, q_j\})$, $G(\{\boldsymbol{q}, q_i\},\{\boldsymbol{q}, q_i\})$, and $G(\{\boldsymbol{q}, q_i, q_j\},\{\boldsymbol{q}, q_i, q_j\})$ are all Gram determinants appearing in the Baikov representation of one-loop integral family and $G(\{\boldsymbol{q}, q_i\},\{\boldsymbol{q}, q_j\})$ is related to them by the following identity

$$
\begin{aligned}
G(\{\boldsymbol{q}, q_i\},\{\boldsymbol{q}, q_i\})G(\{\boldsymbol{q}, q_j\},\{\boldsymbol{q}, q_j\}) = G(\{\boldsymbol{q}, q_i\},\{\boldsymbol{q}, q_j\})^2 \\
+ G(\{\boldsymbol{q}\},\{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\},\{\boldsymbol{q}, q_i, q_j\}) \,.
\end{aligned} \tag{3.2}
$$

The term "maximal cut" in the one-loop context means cutting all propagators, that is, setting $x_i \to 0, \forall i$. To illustrate this, consider the one-loop four-mass box depicted in Fig. 3, with notations taken from [65]. We have the following expression for one of the algebraic symbol letters:

$$
W_{46} = \frac{f_{46} + r_1 r_2}{f_{46} - r_1 r_2} = \left. \frac{G(\{p_1, p_2, k_1\},\{p_1, p_2, p_3\}) + \sqrt{-G(p_1, p_2)G(k_1, p_1, p_2, p_3)}}{G(\{p_1, p_2, k_1\},\{p_1, p_2, p_3\}) - \sqrt{-G(p_1, p_2)G(k_1, p_1, p_2, p_3)}} \right|_{\text{mc}} \tag{3.3}
$$

where $\{\boldsymbol{q}\} = \{p_1, p_2\}, q_i = k_1, q_j = p_3$. "mc" means taking maximal cut which is equal to setting all Baikov variables x_i to 0. Other symbol letters are similar. The key idea here is to promote the algebraic letters, which are functions over kinematic space $\mathbb{C}[S]$, to functions of Gram determinants defined over a larger space that includes Baikov variables $\mathbb{C}[S, \{x_i\}]$. S denotes the set of kinematic variables. Then we project these expressions back into the kinematic space by "maximal cut", that is, fixing Baikov variables to a set of values. It is difficult to directly generate expressions in $\mathbb{C}[S]$ to higher loop because they are all expressions of external kinematics. Higher-loop integral family can share the same external kinematics as one-loop integral families, thus more information must be hidden in the loop-momentum structure which is incarnated by x_i. The ansatz with functions in $\mathbb{C}[S, \{x_i\}]$ (3.1) allows us to capture this additional information perfectly.

The generalization of above expressions to higher loops is straightforward. All Gram determinants $G(\ldots)$ should be taken from Baikov representations for the higher-loop family and the maximal cut will be replaced by the solutions solved for Baikov variables when calculating the leading singularities. One difference from one-loop case is that higher-loop families may have multiple solutions for Baikov variables in a sector. Thus for higher-loop families, there is another kind of ansatz

$$A_3: \log \frac{G(\{l', \boldsymbol{p}\}, \{l, \boldsymbol{p}\}) + \sqrt{G(\{l', \boldsymbol{p}\}, \{l', \boldsymbol{p}\})G(\{l, \boldsymbol{p}\}, \{l, \boldsymbol{p}\})}}{G(\{l', \boldsymbol{p}\}, \{l, \boldsymbol{p}\}) - \sqrt{G(\{l', \boldsymbol{p}\}, \{l', \boldsymbol{p}\})G(\{l, \boldsymbol{p}\}, \{l, \boldsymbol{p}\})}}, \tag{3.4}$$

where l' and l stand for different solutions for loop momentum derived from the same representation and $\boldsymbol{p}$ is a set of external momenta. $G(l, \boldsymbol{p})$ should be Gram determinant which generates the corresponding square-root leading singularities. This "mixing" of different solutions and the ansatz A_3 has also been derived in the Appendix of [70] using languages of embedding space. Although ansatz A_3 can be considered a special case of A_2, we will distinguish it as the second type ansatz in package `BaikovLetter` (A_1 and A_2 are referred to as the first type).

Now, let us analyze why above ansatz for algebraic letters is reasonable from another perspective, which put more emphasis on the structure of leading singularities themselves. Symbol letters are analytic functions $W(S)$ of kinematic variables which we denote as S. It is their total derivatives that will appear in the canonical differential equations. Study these symbol letters from their partial derivatives with respect to a *single* kinematic variable can be complicated, as the dependence on a single variable may be intricate. Due to the same reason, reconstructing the total derivatives from partial derivatives may also be complicated. From the experience of one-loop calculations, it is much simpler to treat the "independent variables" of symbol letters as polynomials of kinematic variables, denoted as $L(S)$. It is natural to take these polynomials to be leading singularities. From the discussion in the previous section, these leading singularities are calculated from Gram determinants and thus it motivates the usage of Gram determinants as building blocks. The relations between symbol letters and Baikov Gram determinants are summarized in Fig. 4.

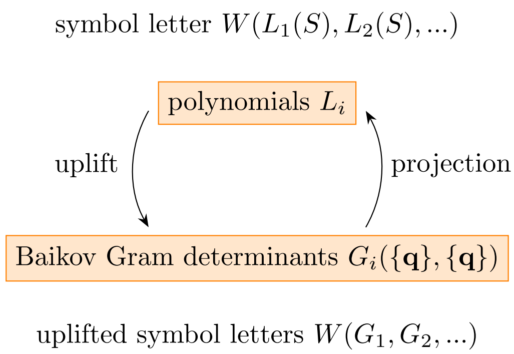

Figure 4. Relations between symbol letters and Gram determinants. Here the projection is performed by the process of fixing Baikov variables to its solutions when calculating multivariate residues. uplift is an inverse of projection. Both are not unique. In practice, we directly write down the ansatz of Gram determinants without truely perform the uplift.

However, these $L(S)$ are not independent. Their relations are typically captured by the relations between Gram determinants (3.2). In these cases, we can study these symbol letters directly by the uplifted symbol letters and their derivatives with respect to those G_i rather than specific kinematic variables S. Let us now consider how the ansatz (3.1) is related to its building blocks using the following abbreviation

$$
\begin{aligned}
&b = G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}),\ a = G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_i\}),\ c = G(\{\boldsymbol{q}, q_j\}, \{\boldsymbol{q}, q_j\})\,, \\
&d = G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\}),\ e = G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\}).
\end{aligned} \tag{3.5}
$$

Note that these five items are not all independent, they satisfy relation (3.2):

$$
b^2 - ac = -de\,. \tag{3.6}
$$

If we take a, c as independent variables, then we will have two types of denominators in the derivatives of symbol letters. One is a square root and the other is rational.

$$
\begin{aligned}
&\partial_b \log \frac{b + \sqrt{b^2 - ac}}{b - \sqrt{b^2 - ac}} = \frac{2}{\sqrt{b^2 - ac}} = \frac{2}{\sqrt{-de}} = \frac{2}{\sqrt{-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}}\,, \\
&\partial_b \log \frac{b + \sqrt{ac}}{b - \sqrt{ac}} = \frac{-2\sqrt{ac}}{b^2 - ac} = \frac{2\sqrt{ac}}{de} = \frac{2\sqrt{ac}}{G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}\,.
\end{aligned} \tag{3.7}
$$

If we take d, e as independent variables, then another two symmetric expressions can be derived,

$$
\begin{aligned}
&\partial_b \log \frac{b + \sqrt{b^2 + de}}{b - \sqrt{b^2 + de}} = \frac{2}{\sqrt{b^2 + de}} = \frac{2}{\sqrt{ac}} = \frac{2}{\sqrt{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_i\})G(\{\boldsymbol{q}, q_j\}, \{\boldsymbol{q}, q_j\})}}\,, \\
&\partial_b \log \frac{b + \sqrt{-de}}{b - \sqrt{-de}} = \frac{-2\sqrt{-de}}{b^2 + de} = \frac{-2\sqrt{de}}{ac} = \frac{-2\sqrt{-de}}{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_i\})G(\{\boldsymbol{q}, q_j\}, \{\boldsymbol{q}, q_j\})}\,.
\end{aligned} \tag{3.8}
$$

Here in (3.7) and (3.8) only the denominators are important[6] since it determines what kind of $\mathrm{d}\log$ we will construct. According to above classification of denominators, we divide all Gram determinants into two groups. The first group is those with half-integer power and the second with integer power. The first group can be used to construct $\mathrm{d}\log$s corresponding to the first line of (3.7) and (3.8). The second group can be used to construct $\mathrm{d}\log$s corresponding to the second line of (3.7) and (3.8). And we will never mix Gram determinants from these two groups. In this way, given two groups of Gram determinants, we can always construct above two kinds of algebraic letters, that is, A_1 and A_2 in (3.1).

Finally, although the Gram determinants used to construct algebraic letters are highly restricted, the ansatz for algebraic letters can still yield a large number of possibilities in complicated cases. We impose additional constraints based on consistency conditions:

$\mathbb{T}_1$: After performing the projection of Baikov determinants in the ansatz, the square roots in the expression should be either single or products of square-root type leading singularities. This is natural because all square roots in the canonical differential equations originate from those appearing in the UT basis which are those square-root leading singularities.

[6]Recall that when we want to reconstruct a $\mathrm{d}\log$ from its partial derivatives, only the denominators are important. Actually, performing the derivatives with respect to a, c, there is always a common denominator just the same as derivative with respect to b.

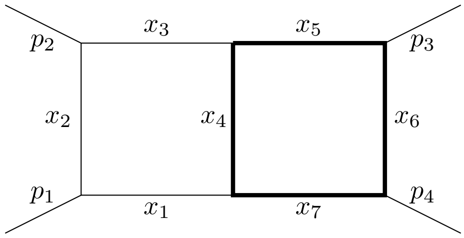

Figure 5. Double box family with one inner loop massive. All massive propagators are with the same mass m^2.

$\mathbb{T}_2$: The second constraint is that, Gram determinants selected as building blocks of the ansatz must come from the same Baikov representation or from two Baikov representations that lie on the same path in the Baikov representation tree (i.e., one representation can be derived from the other by integrating some variables out). Only these Gram determinants are related to each other by (3.2) and they are naturally related to the appearing of algebraic letters as mentioned in Sec. 2.2.2.

$\mathbb{T}_3$: The product of the numerator and denominator of an algebraic letter should be a product of some leading singularities. This is natural since those leading singularities we calculated in the last section should have contained all the singularities in a family.

After obtaining a set of algebraic letters satisfying above constraints, we check their independence through numerical sampling and ultimately identify all independent ones. The implementation of this process in our package `BaikovLetter` is described in greater detail in App. B.3.

4 Examples for symbol alphabets from Baikov Gram determinants

In this section, we apply our methods to several examples and mainly focus on some non-trivial features in them. For detailed usage of functions in the package `BaikovLetter`, we refer to App. B.3. These examples can be divided into two categories: those with massive internal lines and those with all internal lines massless. There are some differences in the general features between them. Finally, we will discuss why this method may fail in certain non-planar integral families.

4.1 Family with massive propagators: double box with inner massive loop

The first example is the double box family with one massive internal loop. This integral family serves as an example that the candidates for rational letters derived from our method can sometimes exceed those actually appearing in the CDE. This is a non-trivial feature, as we will see some square-root leading singularities do not appear as rational letters, even though they indeed appear as leading singularities in the UT basis. This family is depicted in Fig. 5. It has been studied in [53] and is defined as

$$
\begin{aligned}
&x_1 = k_1^2,\ x_2 = (k_1 - p_1)^2,\ x_3 = (k_1 - p_1 - p_2)^2,\ x_4 = (k_1 - k_2)^2 - m^2,\\
&x_5 = (k_2 - p_1 - p_2)^2 - m^2,\ x_6 = (k_2 - p_1 - p_2 - p_3)^2 - m^2,\ x_7 = k_2^2 - m^2,\\
&x_8 = (k_2 - p_1)^2 - m^2,\ x_9 = (k_1 - p_1 - p_2 - p_3)^2\,,
\end{aligned} \tag{4.1}
$$

where the last two are ISPs and the kinematic configuration is

$$
p_i^2 = 0,\ (p_1 + p_2)^2 = s,\ (p_2 + p_3)^2 = t\,. \tag{4.2}
$$

We first generate a set of candidates for rational letters, which are

$$
s, t, m^2, s + t, 4m^2 - s, 4m^2 + s, 4m^2 - t, 4m^2(s + t) - st, m^4 s - 2m^2 st - 4m^2 t^2 + st^2\,. \tag{4.3}
$$

And the corresponding square-root type leading singularities are

$$
\begin{aligned}
&\sqrt{s(s - 4m^2)}, \sqrt{t(t - 4m^2)}, \sqrt{s(s + 4m^2)}, \sqrt{st(st - 4m^2 s - 4m^2 t)},\\
&\sqrt{s(m^4 s - 2m^2 st - 4m^2 t^2 + st^2)}
\end{aligned} \tag{4.4}
$$

Among the above candidates, the last one does not appear as a rational letter in the final CDE, although it is a leading singularity of one UT integral in this family. Usually leading singularities will appear as a rational letter in the entry which corresponds to the dependence of the total derivative of the corresponding UT integral on itself. This seems to be a special phenomenon occurring in families with massive propagators.

Next, we construct a set of algebraic letters using Gram determinants according to ansatz A_1, A_2, A_3 and project these expressions onto the final algebraic letters. After applying all three constraints, we finally obtain a set of independent algebraic symbol letters.

$$
\begin{aligned}
&W_9 = \frac{1 + \sqrt{1 + u + v}}{1 - \sqrt{1 + u + v}},\ W_{10} = \frac{1 - \sqrt{1 + u}}{1 + \sqrt{1 + u}},\ W_{11} = \frac{1 - \sqrt{1 + v}}{1 + \sqrt{1 + v}},\ W_{12} = \frac{1 - \sqrt{1 - u}}{1 + \sqrt{1 - u}},\\
&W_{13} = \frac{1 + v - \sqrt{(1 + v)(1 + u + v)}}{1 + v + \sqrt{(1 + v)(1 + u + v)}},\ W_{14} = \frac{1 + u - \sqrt{(1 + u)(1 + u + v)}}{1 + u + \sqrt{(1 + u)(1 + u + v)}},\\
&W_{15} = \frac{2 - u + v - 2\sqrt{(1 - u)(1 + v)}}{2 - u + v + 2\sqrt{(1 - u)(1 + v)}},
\end{aligned}
$$

$$
\begin{aligned}
&W_{16} = \frac{4 - v - \sqrt{16 + 16u + 8v + v^2}}{4 - v + \sqrt{16 + 16u + 8v + v^2}},\ W_{17} = \frac{4 + v - \sqrt{16 + 16u + 8v + v^2}}{4 + v + \sqrt{16 + 16u + 8v + v^2}},\\
&W_{18} = \frac{4 + 4u + v + \sqrt{(1 + u)(16 + 16u + 8v + v^2)}}{4 + 4u + v - \sqrt{(1 + u)(16 + 16u + 8v + v^2)}},\\
&W_{19} = \frac{4 + 4u + 3v - \sqrt{(1 + u + v)(16 + 16u + 8v + v^2)}}{4 + 4u + 3v + \sqrt{(1 + u + v)(16 + 16u + 8v + v^2)}}\,.
\end{aligned} \tag{4.5}
$$

where $u = -4m^2/s, v = -4m^2/t$. We have verified that the above 11 algebraic letters exactly match those appearing in the CDE. We also directly reconstruct the symbol letters from the partial derivatives of the UT basis and find that W_{18} is a combination of two algebraic letters, which always appear together with a fixed relative sign. One advantage of our method is that we can directly identify the combination without reconstructing the two letters separately. Among the above algebraic letters, W_{19} originates from the ansatz A_3, and we present its expression here as an explicit example.

$$W_{19} = \frac{G(\{k_2', p_1, p_2, p_3\}, \{k_2, p_1, p_2, p_3\}) + \sqrt{G(k_2', p_1, p_2, p_3)G(k_2, p_1, p_2, p_3)}}{G(\{k_2', p_1, p_2, p_3\}, \{k_2, p_1, p_2, p_3\}) - \sqrt{G(k_2', p_1, p_2, p_3)G(k_2, p_1, p_2, p_3)}}, \tag{4.6}$$

where k_2' and k_2 correspond to the two solutions of loop momenta k_2

$$\begin{aligned} &\sigma_1 : \{\{x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_8 \to 0\}\}, \\ &\sigma_2 : \{\{x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_8 \to -m^2\}\}. \end{aligned} \tag{4.7}$$

Note that we have suppressed Baikov variables relevant to k_1 in the above expression since $G(k_2, p_1, p_2, p_3)$ is only relevant to k_2. The mixing between these two solutions have the feature that they differ by only one variable and it corresponds to the Landau singularity $m^2 = 0$. Thus it is consistent with the picture in [69].

We have also verified other families with massive propagators, such as the sunrise family with two massive propagators and the double box family with a massive outer loop. The latter shares the same feature presented above, that is, rational letters in CDE are less than the leading singularities but algebraic letters exactly match those in CDE.

4.2 Family with massive propagators: pentagon box with one massive propagator

There are two penta-box families PB_A and PB_C which have been studied using canoncial differential equations in the process $pp \to t\bar{t}H$ and $pp \to t\bar{t}j$ at hadron colliders [84, 85]. The symbol alphabets of them can also be calculated using our package and the results are verified to be consistent with CDE provided by the authors[7]. These two families are similar and here we will take one of them, the PB_A topology, as an example to demonstrate how some non-trivial poles of Baikov variables can be generated from quadratic polynomials. This family is depicted in Fig. 6 and is defined as

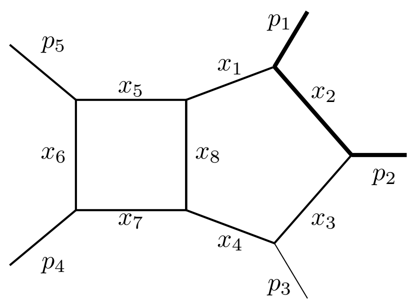

Figure 6. The pentabox family PB_A. The thick lines are massive with mass m_t.

$$\begin{aligned} &x_1 = k_1^2,\ x_2 = (k_1 - p_1)^2 - m_t^2,\ x_3 = (k_1 - p_1 - p_2)^2,\ x_4 = (k_1 - p_1 - p_2 - p_3)^2, \\ &x_5 = k_2^2,\ x_6 = (k_2 + p_1 + p_2 + p_3 + p_4)^2,\ x_7 = (k_2 + p_1 + p_2 + p_3)^2,\ x_8 = (k_1 + k_2)^2, \\ &x_9 = (k_1 - p_1 - p_2 - p_3 - p_4)^2, x_{10} = (k_2 + p_1)^2 - m_t^2, x_{11} = (k_2 + p_1 + p_2)^2. \end{aligned} \tag{4.8}$$

[7]Note that, there are also families like PB_B in these process which contains elliptic sectors or sectors with nested square roots as leading singularities. These cases are beyond consideration of our package.

The kinematics are defined as

$$
\begin{aligned}
&p_1^2 = p_2^2 = m_t^2,\, p_3^2 = p_4^2 = p_5^2 = 0, \\
&(p_1+p_2)^2 = s_{12},\, (p_2+p_3)^2 = s_{23},\, (p_3+p_4)^2 = s_{34},\, (p_4+p_5)^2 = s_{45},\, (p_5+p_1)^2 = s_{15}.
\end{aligned}
\tag{4.9}
$$

We calculate all the leading singularities for this family and obtain a set of 33 candidates, which exactly match the rational letter set in the CDE. Among them,the square-root type leading singularities are

$$
\begin{aligned}
&r_1 \equiv \sqrt{s_{12}(s_{12}-4m_t^2)},\ r_2 \equiv \sqrt{s_{12}\left(s_{12}s_{15}^2 - 2s_{12}s_{15}s_{23} + s_{12}s_{23}^2 + 4m_t^2 s_{34}s_{45}\right)}, \\
&r_3 \equiv \sqrt{\lambda(m_t^2, s_{23}, s_{45})},\ r_4 \equiv \sqrt{\lambda(m_t^2, s_{15}, s_{34})}, \\
&r_5 \equiv \sqrt{m_t^4 + s_{12}^2 + s_{23}^2 - 2m_t^2 s_{12} - 2m_t^2 s_{23} + 2s_{12}s_{23} - 4m_t^2 s_{45}}, \\
&\mathrm{tr}_5 \equiv 4\sqrt{G(p_1,p_2,p_3,p_4)},\ r_1\mathrm{tr}_5 = 4\sqrt{s_{12}(s_{12}-4m_t^2)G(p_1,p_2,p_3,p_4)},
\end{aligned}
\tag{4.10}
$$

where $\lambda(x,y,z) = x^2+y^2+z^2-2xy-2yz-2zx$. r_i's are defined for simplicity. They are related to the square roots defined in [85] by the following relations:

$$
r_1 = s_{12}\beta_{12},\, r_2 = 2s_{12}\Lambda_6,\, r_3 = 2\Delta_{3,1},\, r_4 = 2\Delta_{3,2},\, r_5 = 2\Delta_{3,4} \tag{4.11}
$$

Next, we construct the ansatz for algebraic letters and project them onto kinematic space. We classify the independent algebraic letters by the square roots involved and list their numbers in Tab. 1. The total number of independent algebraic letters is 38, which agrees with the results presented in [85]. Note that the number of algebraic letters is 41 in the CDE of PB_A. However, some letters always appear in fixed combinations:

$$
\mathrm{d}\log W_{81}W_{86},\, \mathrm{d}\log\frac{W_{82}}{W_{83}},\, \mathrm{d}\log\frac{W_{76}}{W_{77}}, \tag{4.12}
$$

So the independent number is 38. W_i are symbol letters defined in [85]. One feature of our method is that we can directly identify the algebraic letters appearing in CDE (the combinations) which indicates that the ansatz is proper for this kind of problems.

Now we discuss a non-trivial pole solved in the subsector $\{1,2,3,4,6,7,8\}$, which is closely related to the algebraic letters involving r_2:

$$
\begin{aligned}
&\{\{x_1\to 0, x_2\to 0, x_3\to 0, x_4\to 0, x_6\to 0, x_7\to 0, x_8\to 0\}, \\
&\quad\{x_5\to\infty, x_{10}\to\infty, x_{11}\to\infty, x_{11}\to\frac{s_{34}x_{10}}{s_{15}-s_{23}}, x_5\to -\frac{s_{45}x_{10}}{s_{15}-s_{23}}, x_{10}\to 0\}\}.
\end{aligned}
\tag{4.13}
$$

The first part corresponds to the maximal cut of propagators in this sector, and the second part represents one solution for the remaining ISPs. The infinity indicates that this solution is obtained by first homogenizing the Baikov polynomials with x_0 and then setting x_0 to 0. Since the polynomials are homogenized, the last term $x_{10}\to 0$ indicates that x_{10} is taken as the reserved variable, meaning all variables are rescaled by x_{10}. This pole corresponds

square roots involved	number of independent algebraic letters
tr_5	9
$r_1(\beta_{12})$	4
$r_3(\Delta_{3,1})$	4
$r_4(\Delta_{3,2})$	4
$r_2(\Lambda_6)$	3
$r_5(\Delta_{3,4})$	3
$r_1\mathrm{tr}_5$	2
$r_1 r_2$	1
$r_1 r_3$	1
$r_1 r_4$	1
$r_1 r_5$	1
$r_3 r_5$	1
$r_2\mathrm{tr}_5$	1
$r_3\mathrm{tr}_5$	1
$r_4\mathrm{tr}_5$	1
$r_5\mathrm{tr}_5$	1
total	38

Table 1. The number of independent algebraic letters classified by square roots involved in PB_A.

to a second-type Landau singularity. It is nontrivial because it is entirely derived from a quadratic polynomial Q:

$$\begin{aligned}
&s_{34}x_5^2\left(s_{15}m_t^2-s_{23}m_t^2+s_{23}^2-s_{15}s_{23}+s_{23}s_{34}\right)-s_{34}x_{10}x_5\left(s_{12}s_{15}-s_{12}s_{23}-s_{34}m_t^2-s_{45}m_t^2\right.\\
&\left.+s_{23}s_{34}-s_{15}s_{45}+2s_{23}s_{45}+s_{34}s_{45}\right)+x_{11}x_5\left(s_{23}s_{34}m_t^2+s_{15}s_{45}m_t^2-s_{15}s_{34}m_t^2-s_{23}s_{45}m_t^2\right.\\
&\left.+s_{12}s_{15}^2+s_{12}s_{23}^2-2s_{12}s_{15}s_{23}-s_{23}^2s_{34}+s_{15}s_{23}s_{34}-s_{15}^2s_{45}+s_{15}s_{23}s_{45}+s_{15}s_{34}s_{45}+s_{23}s_{34}s_{45}\right)\\
&-s_{45}x_{11}^2\left(s_{15}m_t^2-s_{23}m_t^2-s_{15}^2+s_{15}s_{23}-s_{15}s_{45}\right)+s_{45}x_{10}x_{11}\left(s_{34}m_t^2+s_{45}m_t^2+s_{12}s_{15}-s_{12}s_{23}\right.\\
&\left.-2s_{15}s_{34}+s_{23}s_{34}-s_{15}s_{45}-s_{34}s_{45}\right)-s_{34}\left(s_{12}-s_{34}-s_{45}\right)s_{45}x_{10}^2.
\end{aligned} \tag{4.14}$$

Note that we solve the leading singularities using a variable-by-variable approach (see App. B.2 for more details). When solving the first variable x_{11}[8], the solution will involve square roots:

$$\begin{aligned}
x_{11}=&\Big[x_5\left(-s_{15}s_{34}m_t^2+s_{23}s_{34}m_t^2+s_{15}s_{45}m_t^2-s_{23}s_{45}m_t^2+s_{12}s_{15}^2+s_{12}s_{23}^2-2s_{12}s_{15}s_{23}\right.\\
&\left.-s_{23}^2s_{34}+s_{15}s_{23}s_{34}-s_{15}^2s_{45}+s_{15}s_{23}s_{45}+s_{15}s_{34}s_{45}+s_{23}s_{34}s_{45}\right)+s_{45}x_{10}\big(s_{34}m_t^2+s_{45}m_t^2\\
&+s_{12}s_{15}-s_{12}s_{23}-2s_{15}s_{34}+s_{23}s_{34}-s_{15}s_{45}-s_{34}s_{45}\big)+\mathrm{tr}_5\left(\left(s_{15}-s_{23}\right)x_5+s_{45}x_{10}\right)\Big]\Big/\Big[2s_{45}\\
&\left(s_{15}^2-s_{15}m_t^2-s_{15}s_{23}+s_{15}s_{45}+s_{23}m_t^2\right)\Big],
\end{aligned} \tag{4.15}$$

[8] It is a quadratic polynomial for all the three variables x_5, x_{10} and x_{11}, here we choose to start from x_{11}. It will be the same starting from another two variables.

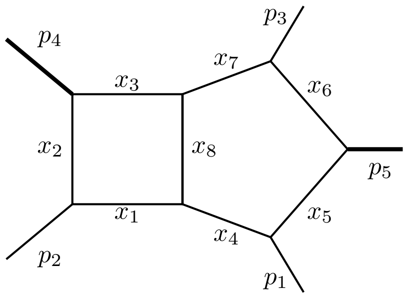

Figure 7. The two-loop pentabox family with p_4 and p_5 massive.

where tr_5 is the square root defined above. Next, we solve x_5[9]

$$x_5 = \frac{s_{45}x_{10}}{s_{23} - s_{15}}, \tag{4.16}$$

and substitute its solution into that of x_{11}. The term proportional to square roots vanish and we get the linear pole presented in (4.13). As we can see, the complexity mainly arises from the variable-by-variable approach we employ. If we use the pinch condition [72] for the Baikov polynomial Q[10]:

$$Q = 0, \frac{\partial Q}{\partial x_5} = 0, \frac{\partial Q}{\partial x_{11}} = 0 \tag{4.17}$$

We can directly obtain the solution in (4.13). Nevertheless, the variable-by-variable approach is more systematic and easier to implement in the package.

Another family, PB_C, can be analyzed in the same way and the results also agree perfectly with the CDE calculated in [85].

4.3 Family with massless propagators: pentabox with two massive external legs

Now we study the pentabox families with two massive external legs, which have been studied in [22]. We use one of six families PBzmz as an example to demonstrate the necessity to consider independent loop-by-loop Baikov representations of one sector to obtain all the leading singularities. The second thing we want to illustrate with this example is that there may still be spurious algebraic letters remaining even if they pass all the three constraints $\mathbb{T}_{1,2,3}$ in Sec. 3. This family is depicted in Fig. 7 and is defined as

$$\begin{aligned}
&x_1 = k_1^2,\, x_2 = (k_1+p_2)^2,\, x_3 = (k_1+p_2+p_4)^2,\, x_4 = k_2^2, \\
&x_5 = (k_2+p_1)^2,\, x_6 = (k_2-p_2-p_3-p_4)^2,\, x_7 = (k_2-p_2-p_4)^2,\, x_8 = (k_1+k_2)^2, \\
&x_9 = (k_1+p_2+p_3+p_4)^2,\, x_{10} = (k_1-p_1)^2,\, x_{11} = (k_2-p_2)^2.
\end{aligned} \tag{4.18}$$

[9]After solving x_{11}, the square root of discriminant of this quadratic polynomial $\sqrt{\Delta} = \mathrm{tr}_5\left[\,(s_{15}-s_{23})\,x_5 + s_{45}x_{10}\right]$ provides a pole for x_5.

[10]This is a homogeneous polynomial, only two free Baikov variables are present, here we take them to be x_5, x_{11}. That is, they can all be recaled by a x_{10}.

And the kinematics are defined as

$$\begin{aligned}&p_1^2=p_2^2=p_3^2=0,\ p_4^2=s_4,\ p_5^2=s_5,\\&(p_1+p_2)^2=s_{12},\ (p_2+p_3)^2=s_{23},\ (p_3+p_4)^2=s_{34},\ (p_4+p_5)^2=s_{45},\ (p_5+p_1)^2=s_{15}.\end{aligned} \tag{4.19}$$

Running the analysis for all the leading singularities yields 43 candidates which agrees with the results calculated in CDE. Among them, one rational letter is special:

$$L_1 = s_{12}s_{15}s_4 - s_{15}s_{34}s_{45} + s_{23}s_{34}s_5. \tag{4.20}$$

It originates exclusively from the subsector $\{2,3,4,5,6,8\}$. More importantly, in two independent Baikov representations of this sector, it only emerges as a leading singularity in one of them. The two representations are as follows:

$$\begin{aligned}u_1 =&G(p_1,p_2,p_3,p_4)^{1/2+\epsilon}G(k_2,p_1,p_2,p_3,p_4)^{-1-\epsilon}[-G(k_2-p_2,p_4)]^{-1/2+\epsilon}\\&\qquad\qquad\qquad\qquad\times G(k_1+p_2,k_2-p_2,p_4)^{-\epsilon},\\u_2 =&G(p_1,p_2,p_3,p_4)^{1/2+\epsilon}G(k_1,p_1,p_2+p_4,p_3,p_4)^{-1-\epsilon}[-G(k_1,p_1,p_2+p_3+p_4)]^{\epsilon}\\&\qquad\qquad\qquad\qquad\times G(k_1,k_2,p_1,p_2+p_3+p_4)^{-1/2-\epsilon}.\end{aligned} \tag{4.21}$$

These two loop-by-loop representations correspond to integrating k_1 out first or integrating k_2 first. They are referred to as two independent representations for this sector. After the maximal cut of this sector, there are 2 ISPs x_7, x_{11} remaining in the first representation and 3 ISPs x_1, x_9, x_{10} remaining in the second representation. However, L_1 only appears in the second representation u_2 and emerges as a second type Landau singularity. Although it has not been identified as a leading singularity in u_1, this does not imply that it is a spurious letter. In fact, if we impose the condition $L_1 = 0$, the dimension of twisted cohomology group defined by u_1 will drop. This indicates that L_1 is a genuine leading singularity of this sector. This idea of using dimension drop to identify genuine Landau singularities has been introduced in [78, 79] and is extensively used in our package to remove spurious letters, as described in App. B.2. As an example, we use `GetDimension[]` in the package to calculate the dimension of twisted cohomology group defined by u_1 (with maximal cut applied):

```
 In:= GetDimension[u1, {x7,x11}]
Out:= 8
```

Here $\texttt{u}_1$ should be in the form `Power[...,a]*Power[...,b]*..` and the variables must be named `Subscript[x,i]`. Next, we choose a set of rational number that satisfy $L_1 = 0$:

```
numrep = {s12→214, s23→294, s34→307, s45→160, s15→5, s4→-8,
                                                  s5→42360/15043}.
```

Running the command again yields

```
 In:= GetDimension[u1/.numrep, {x7,x11}]
Out:= 7
```

Thus, there is indeed a dimension drop here.

Then we calculate all the algebraic letters using the ansatz and project them into kinematic space. Again, we find our results require two algebraic letters defined in [22] to appear in a specific combination

$$\mathrm{d}\log W_{217}W_{219}. \tag{4.22}$$

However, in our final results there are two algebraic letters that do not appear in CDE:

$$\begin{aligned}
a_1 =&\log \frac{s_{15}s_4+s_{23}s_4-s_4^2+s_{15}s_{45}-s_{23}s_{45}+s_4s_{45}-s_5s_{15}+s_5s_{23}+s_4s_5-\sqrt{\Delta_3^{(1)}\Delta_3^{(4)}}}{s_{15}s_4+s_{23}s_4-s_4^2+s_{15}s_{45}-s_{23}s_{45}+s_4s_{45}-s_5s_{15}+s_5s_{23}+s_4s_5+\sqrt{\Delta_3^{(1)}\Delta_3^{(4)}}},\\
a_2 =&\log \frac{P+\sqrt{\Delta_3^{(1)}\Delta_3^{(7)}}}{P-\sqrt{\Delta_3^{(1)}\Delta_3^{(7)}}}.
\end{aligned} \tag{4.23}$$

where $\Delta_3^{(1)}$, $\Delta_3^{(4)}$ and $\Delta_3^{(7)}$ are defined as in [22]:

$$\begin{aligned}
&\Delta_3^{(1)}=\lambda(s_4,s_{45},s_5),\ \Delta_3^{(4)}=\lambda(s_4,s_{15},s_{23}),\\
&\Delta_3^{(7)}=(s_4+s_{12}-s_{34}-s_{45}+s_{15}-s_5)^2+4s_5(s_{12}+s_{23}-s_{45}).
\end{aligned} \tag{4.24}$$

and P is defined as

$$\begin{aligned}
P \equiv& s_4^2-2s_5s_4+s_{12}s_4+s_{15}s_4-s_{34}s_4-2s_{45}s_4+s_5^2+s_{45}^2+s_5s_{12}-s_5s_{15}+2s_5s_{23}+s_5s_{34}\\
&-2s_5s_{45}-s_{12}s_{45}-s_{15}s_{45}+s_{34}s_{45}.
\end{aligned} \tag{4.25}$$

They are all related to $\Delta_3^{(1)}$ and a detailed analysis of their origin shows that they are related to the poles of subsector $\{2,3,4,5,6,8\}$ and $\{1,2,3,5,6,8\}$. These two subsectors are both box-triangle configurations and are related to each other through permutations of external legs. It is interesting to investigate why such two algebraic letters do not appear in the CDE.

In the cases of 6-point integral families as those studied in [24], the appearance of spurious algebraic letters becomes more common. There are mainly two reasons. First, the kinematics is in 4 dimension but we perform calculations using d dimensional representation. Some algebraic letters degenerate in the 4d kinematics to rational ones. However, even after removing such terms, we still find some that does not appear in the final result of CDE. Thus, there may be additional constraints required for the general ansatz we have proposed. Nevertheless, the leading singularities we calculated match those present in CDE perfectly after removing those vanish in the 4d limit.

4.4 Family with massless propagators: three-loop pentaboxbox

Now we study a three-loop integral family that has been recently studied in [31]. For three-loop families, the calculation typically takes much longer time. It takes about 2 days[11] to obtain the symbol alphabet for this family using `BaikovLetter`. The result includes all the

[11]This is tested on an M1 Mac mini.

symbol letters in CDE calculated by [31]. In the following, we focus on the additional ones obtained, which are spurious for this family. We have observed that for three-loop families, there are usually more symbol letters predicted by the package.

This family is depicted in Fig. 8 and is defined the same as in [31]:

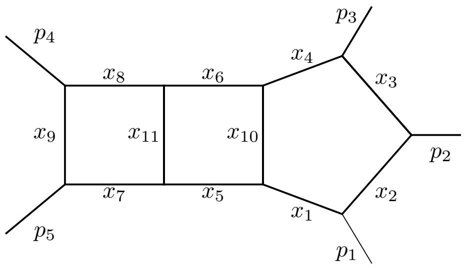

Figure 8. The three-loop pentaboxbox family. All lines are massless.

$$
\begin{aligned}
&x_1 = k_1^2,\ x_2 = (k_1 - p_1)^2,\ x_3 = (k_1 - p_1 - p_2)^2,\ x_4 = (k_1 - p_1 - p_2 - p_3)^2,\\
&x_5 = k_2^2,\ x_6 = (k_2 - p_1 - p_2 - p_3)^2,\ x_7 = k_3^2,\ x_8 = (k_3 - p_1 - p_2 - p_3)^2,\\
&x_9 = (k_3 - p_1 - p_2 - p_3 - p_4)^2,\ x_{10} = (k_1 - k_2)^2,\ x_{11} = (k_2 - k_3)^2,\\
&x_{12} = (k_1 - p_1 - p_2 - p_3 - p_4)^2,\ x_{13} = (k_2 - p_1)^2,\ x_{14} = (k_2 - p_1 - p_2)^2,\\
&x_{15} = (k_2 - p_1 - p_2 - p_3 - p_4)^2,\ x_{16} = (k_3 - p_1)^2,\ x_{17} = (k_3 - p_1 - p_2)^2,\ x_{18} = (k_1 - k_3)^2.
\end{aligned}
\tag{4.26}
$$

The kinematics are defined as

$$
\begin{aligned}
&p_1^2 = p_2^2 = p_3^2 = p_4^2 = p_5^2 = 0,\\
&(p_1 + p_2)^2 = s_{12},\ (p_2 + p_3)^2 = s_{23},\ (p_3 + p_4)^2 = s_{34},\ (p_4 + p_5)^2 = s_{45},\ (p_5 + p_1)^2 = s_{15}.
\end{aligned}
\tag{4.27}
$$

There are totally 21 candidated for rational letters obtained. In addition to the 19 ones appearing in the CDE, there are two additional ones.

$$
\begin{aligned}
S_1 =&(s_{12}s_{15} - s_{12}s_{23} - s_{15}s_{45} + s_{34}s_{45})^2 + 4s_{12}s_{23}s_{34}s_{45}.\\
S_2 =&(s_{12}s_{15} - s_{12}s_{23} - s_{12}s_{34} - s_{23}s_{34} - s_{15}s_{45} + s_{34}s_{45})^2 + 4s_{12}s_{23}s_{34}(s_{15} - s_{23} - s_{34}).
\end{aligned}
\tag{4.28}
$$

They take a form very similar to the genuine singularity tr_5:

$$
\begin{aligned}
\mathrm{tr}_5 \equiv& 4\sqrt{G(p_1, p_2, p_3, p_4)}\\
=&\sqrt{(s_{12}s_{15} - s_{12}s_{23} - s_{23}s_{34} - s_{15}s_{45} + s_{34}s_{45})^2 + 4s_{12}s_{23}s_{34}(s_{15} - s_{23} + s_{45})}.
\end{aligned}
\tag{4.29}
$$

Both of them originate from the subsector $\{2, 3, 5, 8, 9, 10, 11\}$ and the same representation for this sector. In principle, we can verify them by the dimension drop trick introduced before. However, several independent Baikov representations for this sector are too complicated that the dimension calculation can not be finished in given time. Therefore, in such cases, we retain these leading singularities, even though they may be spurious. Interestingly, S_1 is actually one of the five square-root singularities identified ($\Delta_4^{(4)}$ or $\widetilde{W}_{19}$) in the three-loop Feynman integrals for the ladders in the negative geometry [86]. This can be understood because many three-loop Feynman integral families share the same above subsector. It is not a genuine singularity for the integral family discussed here, but it could be one for another integral family with different top sectors. There are also many spurious square-root type leading singularities calculated. Some of them are listed below

$$\sqrt{s_{12}s_{45}},\ \sqrt{s_{15}s_{23}s_{45}(s_{15}-s_{23}+s_{45})},\ \sqrt{S_1} \tag{4.30}$$

Some of them, such as the first two, are unlikely to be square-root type leading singularities, while others, like the last one, are present because they were not identified as spurious singularities. The reason for the abundance of spurious square-root type leading singularities in three-loop or nonplanar families (see the next section) may be that more spurious poles can arise when the representations become increasingly complicated. For this family, tr_5 is the only square-root type leading singularity, while all others are spurious. We obtain 5 independent algebraic letters related to tr_5, which are those appearing in the CDE. The spurious ones related to $\sqrt{S_1}$ are

$$\begin{aligned}&\log\frac{s_{12}s_{15}-s_{12}s_{23}-s_{15}s_{45}+s_{34}s_{45}-\sqrt{S_1}}{s_{12}s_{15}-s_{12}s_{23}-s_{15}s_{45}+s_{34}s_{45}+\sqrt{S_1}},\ \log\frac{s_{12}s_{15}-s_{12}s_{23}+s_{15}s_{45}-s_{34}s_{45}-\sqrt{S_1}}{s_{12}s_{15}-s_{12}s_{23}+s_{15}s_{45}-s_{34}s_{45}+\sqrt{S_1}},\\&\log\frac{s_{12}s_{15}+s_{12}s_{23}-s_{15}s_{45}+s_{34}s_{45}-\sqrt{S_1}}{s_{12}s_{15}+s_{12}s_{23}-s_{15}s_{45}+s_{34}s_{45}+\sqrt{S_1}},\ \log\frac{D-\sqrt{S_1}\mathrm{tr}_5}{D+\sqrt{S_1}\mathrm{tr}_5},\end{aligned} \tag{4.31}$$

where

$$\begin{aligned}D\equiv{}& s_{15}^2s_{12}^2+s_{23}^2s_{12}^2-2s_{15}s_{23}s_{12}^2-s_{23}^2s_{34}s_{12}+s_{15}s_{23}s_{34}s_{12}-2s_{15}^2s_{45}s_{12}+2s_{15}s_{23}s_{45}s_{12}\\&+2s_{15}s_{34}s_{45}s_{12}+2s_{23}s_{34}s_{45}s_{12}+s_{15}^2s_{45}^2+s_{34}^2s_{45}^2-2s_{15}s_{34}s_{45}^2-s_{23}s_{34}^2s_{45}+s_{15}s_{23}s_{34}s_{45}.\end{aligned} \tag{4.32}$$

As mentioned in [86], algebraic letters like the above related to S_1 will also appear in other three-loop families. We explicitly calculate symbol alphabets for the two 8-propagator subsectors in [86]. They are re-depicted in Fig. 9. We run the analysis for these two families

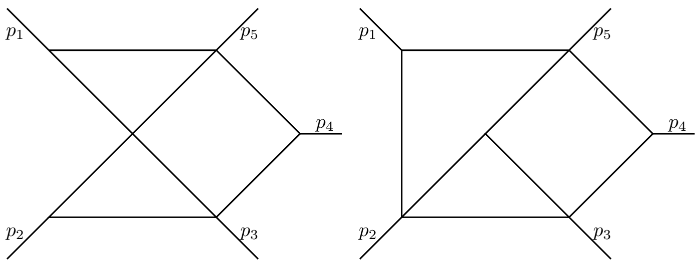

Figure 9. The two 8-propagator sub-sectors, left (a) and right (b), of a three-loop five-point planar family in [86].

separately. For family (a), except for the 18 rational letters[12] which already appear in two-loop planar cases, there are two additional ones just as presented in [86]:

$$\begin{aligned}\widehat{W}_1 ={}& s_{23}s_{34}+s_{15}s_{45}-s_{34}s_{45},\\ \widetilde{W}_{16} ={}& \Delta_4^{(1)}=(s_{12}s_{15}-s_{12}s_{23}+s_{23}s_{34}-s_{34}s_{45})^2+4s_{12}s_{15}s_{34}s_{45}.\end{aligned} \tag{4.33}$$

[12]They are $W_{1\sim5}$, $W_{8\sim9}$, $W_{11\sim20}$ and W_{31} as defined in [14, 31].

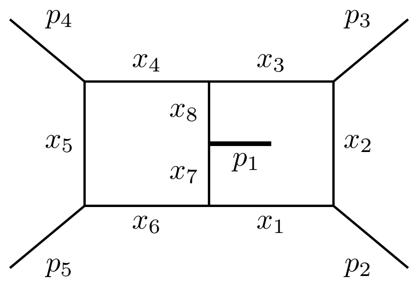

Figure 10. The two-loop nonplanar double pentagon family. p_1 is massive while all other lines are massless.

All the algebraic letters calculated are the 9 ones:

$$\widetilde{W}_{41}, \widetilde{W}_{46}, \widetilde{W}_{51}, \widetilde{W}_{76}, \widetilde{W}_{56\sim 60}. \tag{4.34}$$

While for family (b), except for the 18 rational letters[13] which already appear in two-loop planar cases, only one additional $\widetilde{W}_{16}$ is present. The algebraic letters calculated are the same as family (a).

As the Baikov representations become increasingly complicated, the method we use to identify spurious leading singularities will become less efficient. As one might expect, on the one hand, there will be more independent representations for one sector and we need to check every one of them. On the other hand, the number of remaining ISPs after the maximal cut in one representation will increase. We must pay a higher computational cost for the variable-by-variable solving strategy (see App. B.2) implemented in the package. Nevertheless, we also calculated some other three-loop families, as presented in [70].

4.5 Two-loop nonplanar family: double pentagon with one massive leg

Now we analyze a nonplanar family, DPzz, which has been studied in [21]. We will demonstrate that the methods for algebraic letters fail in this example. However, the calculation of leading singularities and rational letters is general and still works in this case.

The family is depicted in Fig. 10 and is defined as follows

$$\begin{aligned}
&x_1 = l_1^2,\ x_2 = (l_1+p_2)^2,\ x_3 = (l_1+p_2+p_3)^2,\ x_4 = (l_2-p_1-p_2-p_3)^2,\\
&x_5 = (l_2-p_1-p_2-p_3-p_4)^2,\ x_6 = l_2^2,\ x_7 = (l_1+l_2)^2,\ x_8 = (l_1+l_2-p_1)^2,\\
&x_9 = (l_1-p_1-p_2-p_3-p_4)^2,\ x_{10} = (l_2+p_3)^2,\ x_{11} = (l_2+p_2+p_3)^2.
\end{aligned} \tag{4.35}$$

In the following, we will define

$$\tilde{x}_{11} \equiv x_{11} + x_4 - 2x_6 = -2l_2\cdot p_1 + s_{23} + s_{45}, \tag{4.36}$$

[13] They are $W_{1\sim 5}$, $W_{6,8}$, $W_{11\sim 20}$ and W_{31} as defined in [14, 31].

and use $\tilde{x}_{11}$ in the Baikov representations in place of x_{11}. This simplifies the recursive structure of Baikov representations[14]. The kinematics are defined as

$$
\begin{aligned}
& p_1^2 = p_1^2,\, p_2^2 = p_3^2 = p_4^2 = p_5^2 = 0, \\
& (p_1+p_2)^2 = s_{12},\, (p_2+p_3)^2 = s_{23},\, (p_3+p_4)^2 = s_{34},\, (p_4+p_5)^2 = s_{45},\, (p_5+p_1)^2 = s_{15}.
\end{aligned}
\tag{4.37}
$$

By analyzing the leading singularities for all the sectors, we obtain a set of 45 candidates for the rational letters. Among them, 42 are those appearing in the CDE of this integral family. The remaining three spurious ones are[15]

$$
\begin{aligned}
& R_1 = p_1^2 s_{23} - s_{15}s_{23} + s_{15}s_{45},\, R_2 = s_{15}s_{23} - s_{23}^2 + p_1^2 s_{45} - s_{15}s_{45} + 2s_{23}s_{45} - s_{45}^2, \\
& R_3 = p_1^4 + 2p_1^2 s_{12} + s_{12}^2 - 4p_1^2 s_{23} + 4s_{12}s_{23} + 4s_{23}^2 - 4p_1^2 s_{45} - 4s_{12}s_{45} - 8s_{23}s_{45} + 4s_{45}^2.
\end{aligned}
\tag{4.38}
$$

Each of them originates from and only from one sector in the integral family. It can be found that R_1 and R_2 come from two symmetric slash-box subsector $\{1,3,5,6,8\}$ and $\{1,3,4,5,7\}$ respectively. They are symmetric to each other under the exchange of p_4 and p_5. $R_1(R_2)$ was not detected as a spurious leading singularity by our algorithm because it appears as a leading singularity in both independent Baikov representations of its sector. In our algorithms, we do not check leading singularities appearing in all the independent representations of a sector. Only those appearing in one representation but not another will be checked (see App. B.2 for more details). This is a trade-off between efficiency and accuracy. As a result, some spurious leading singularities remain unremoved. By manual verification, $R_1(R_2)$ does not induce a dimension drop for one of the two independent Baikov representations of this sector, confirming that it is a spurious leading singularity. R_3 was not detected as a spurious letter because the numerical solutions we generated for $R_3 = 0$ is not general enough in the package. We can find by hand a rational solution of $R_3 = 0$:

$$
\{p_1^2 \to 9,\, s_{12} \to 14,\, s_{23} \to 72,\, s_{45} \to 239/2\}
\tag{4.39}
$$

which does not induce a dimension drop for either of the two independent representations of sector $\{1,2,4,7,8\}$, where R_3 originates. Thus, it is indeed also a spurious letter.

In contrast to the rational ones, the generated set of square-root type leading singularities contains many spurious ones, although it does cover all the possible square-root type leading singularities appearing in the CDE. *We remove these spurious ones manually.* Now we focus on two genuine ones of the leading singularities:

$$
\begin{aligned}
\Sigma_5^{(4)} =& [(p_1^2 - s_{12} - s_{15} + s_{34})(s_{15} - s_{23} - s_{45}) + s_{15}(s_{15} - s_{23} - s_{34}) + s_{45}(p_1^2 - s_{12})]^2 \\
& - 4s_{23}s_{34}s_{45}(p_1^2 - s_{12} - s_{15} + s_{34}), \\
\Sigma_5^{(2)} =& [(p_1^2 - s_{15} - s_{12} + s_{34})(s_{12} - s_{45} - s_{23}) + s_{12}(s_{12} - s_{45} - s_{34}) + s_{23}(p_1^2 - s_{15})]^2 \\
& - 4s_{45}s_{34}s_{23}(p_1^2 - s_{15} - s_{12} + s_{34}).
\end{aligned}
\tag{4.40}
$$

[14] Note that, the definition of ISPs can be changed for a given family. The physical result will not be affected but the spurious part may depend on the choice of ISPs. This is something like the gauge choice for the physical results. For nonplanar family, we find that taking ISPs to be linear will simplify the calculation.

[15] As previously noted, if we had chosen a different definition of ISPs, we may get a different set of spurious letters.

square roots involved	number of independent algebraic letters
tr_5	12
$\sqrt{\Delta_3^{(1)}}$	4
$\sqrt{\Delta_3^{(3)}}$	4
$\sqrt{\Delta_3^{(1)}\mathrm{tr}_5}$	1
$\sqrt{\Delta_3^{(3)}\mathrm{tr}_5}$	1
total	22

Table 2. The number of independent algebraic letters of family DPzz classified by square roots calculated by `BaikovLetter`. It can be noticed that algebraic letters related to $\sqrt{\Sigma_5^{(2,4)}}$ are missed.

They are understood to appear as square roots. They come from two symmetric subsectors $\{1,2,4,5,7,8\}$ and $\{2,3,5,6,7,8\}$ respectively and are symmetric to each other under the permutation of external legs $3 \leftrightarrow 4, 2 \leftrightarrow 5$. We find that the algebraic letters related to these two square-root type leading singularities are missed by the package. The numbers of algebraic letters calculated by the package, classified by square roots, are listed in Tab. 2 where the square roots involved are defined as in [22]:

$$\mathrm{tr}_5 = 4\sqrt{G(p_1,p_2,p_3,p_4)},\ \sqrt{\Delta_3^{(1)}} = -\det G(p_1,p_2+p_3),\ \sqrt{\Delta_3^{(3)}} = -\det G(p_1,p_3+p_4). \tag{4.41}$$

We will first examine how these two leading singularities are generated from our analysis. Since they are symmetric, we will use $\Sigma_5^{(2)}$, which originates from $\{2,3,5,6,7,8\}$, as an example. There are two independent loop-by-loop representations for this sector. Both of them yield this leading singularity. We analyze only one of them, since they are symmetric to each other as well.

$$\begin{aligned} u \propto G(p_1,p_2,p_3,p_4)^{1/2+\epsilon} G(l_1,p_1,p_2+p_3,p_3,p_4)^{-1-\epsilon}[-G(l_1,p_1,p_2+p_3+p_4)]^{\epsilon} \\ \times G(l_1,l_2,p_1,p_2+p_3+p_4)^{-1/2-\epsilon}. \end{aligned} \tag{4.42}$$

Under the maximal cut $x_{2,3,5,6,7,8} \to 0$, $G(l_1,l_2,p_1,p_2+p_3+p_4)$ becomes a perfect square, thus providing a pole for Baikov variables:

$$\sqrt{G(l_1,l_2,p_1,p_2+p_3+p_4)}\Big|_{mc} = x_1(p_1^2 - s_{15} + s_{23} + s_{45} - \tilde{x}_{11}) - (s_{23} + s_{45} - \tilde{x}_{11})x_9. \tag{4.43}$$

We solve in the order $x_1, x_9, \tilde{x}_{11}$. First, we solve x_1 by (4.43) and substitute this solution into $G(l_1,p_1,p_2+p_3,p_3,p_4)$. The resulting expression is a quadratic polynomial in x_9. We then take the square root of the discriminant of x_9, which yields a quadratic polynomial for $\tilde{x}_{11}$. This operation, taking the square root of the discriminant, corresponds to the following process

$$\frac{1}{(x_9-c_1)(x_9-c_2)}\Big|_{x_9=c_1} = \frac{1}{c_1-c_2} = \frac{1}{\sqrt{\Delta}} \tag{4.44}$$

where Δ is the discriminant of the quadratic polynomial in x_9. Then,

$$\sqrt{\Delta} = 4\sqrt{G(p_1,p_2,p_3,p_4)}\Big[(-s_{12}+s_{34}+s_{45})\,\tilde{x}_{11}^2 + \tilde{x}_{11}\left(p_1^2 s_{45} - s_{12}p_1^2 - 2s_{45}^2 + s_{12}s_{15}\right.$$
$$\left.+3s_{12}s_{23} - 3s_{23}s_{34} + 2s_{12}s_{45} - s_{15}s_{45} - 2s_{23}s_{45} - s_{34}s_{45}\right) - p_1^2 s_{45}^2 + p_1^2 s_{12}s_{45} - p_1^2 s_{23}s_{45}$$
$$+2p_1^2 s_{12}s_{23} + s_{45}^3 - s_{12}s_{45}^2 + s_{15}s_{45}^2 + 2s_{23}s_{45}^2 + s_{23}^2 s_{45} - s_{12}s_{15}s_{45} - 3s_{12}s_{23}s_{45} + s_{15}s_{23}s_{45}$$
$$+2s_{23}s_{34}s_{45} - 2s_{12}s_{23}^2 - 2s_{12}s_{15}s_{23} + 2s_{23}^2 s_{34}\Big]. \tag{4.45}$$

We can then take the square root of the discriminant of $\tilde{x}_{11}$, yielding $\sqrt{\Sigma_5^{(2)}}$. Although we have obtained the leading singularities in this way, the relationship between leading singularities and Baikov Gram determinants is not direct. It is important to note that the projection map from the ansatz $A_{1,2,3}$ to specific algebraic letters is not necessarily directly related to the multivariate poles corresponding to the square-root type leading singularities involved in the algebaic letter. Thus, the above analysis only reveals the origin of $\Sigma_5^{(2)}$, but does not explain how certain Gram determinants can be projected into the desired leading singularities.

Now we try to analyze another integral family closely related to DPzz to gain insights into why algebraic letters related to $\Sigma_5^{(2,4)}$ are missed. There are two possible reasons: first, the ansatz $A_{1,2,3}$ in Sec. 3 may be insufficient and second, some projections from Baikov Gram determinants to leading singularities are missed. The following integral family suggests that the latter may be correct. This integral family is closely related to DPzz. It is the hexa-box family studied in [20] which is depicted in Fig. 11. It is defined as follows

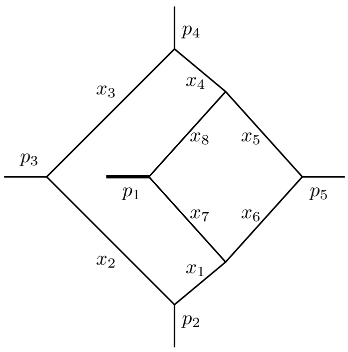

Figure 11. The hexabox family zzz with one massive external leg p_1. All other lines are massless.

$$\begin{aligned}
&x_1 = k_1^2,\ x_2 = (k_1+p_2)^2,\ x_3 = (k_1+p_2+p_3)^2, x_4 = (k_1+p_2+p_3+p_4)^2,\\
&x_5 = (k_1+k_2+p_1+p_2+p_3+p_4)^2,\ x_6 = (k_1+k_2)^2,\ x_7 = k_2^2,\ x_8 = (k_2+p_1)^2,\\
&x_9 = (k_2+p_2)^2,\ x_{10} = (k_1+p_1)^2, x_{11} = (k_2+p_2+p_3)^2.
\end{aligned} \tag{4.46}$$

We will redefine $\tilde{x}_{10} \equiv x_{10} + x_4 - 2x_1 = -2k_1 \cdot p_5 + p_1^2 + s_{15}$ due to the same reason as above. The kinematics are also defined as (4.37). Now, running the analysis for this integral family[16], we obtain 37 candidates for rational letters, which exactly match results

[16]The analysis of nonplanar families usually takes a very long time. For this family, it takes about 1 hour to get all the leading singularities and 17 hours to get all the algebraic ones on an M1 Mac mini.

square roots involved	number of independent algebraic letters
tr_5	11
$\sqrt{\Delta_3^{(1)}}$	4
$\sqrt{\Delta_3^{(3)}}$	4
$\sqrt{\Sigma_5^{(2)}}$	4
$\sqrt{\Delta_3^{(1)}\mathrm{tr}_5}$	1
$\sqrt{\Delta_3^{(3)}\mathrm{tr}_5}$	1
$\sqrt{\Sigma_5^{(2)}\mathrm{tr}_5}$	1
total	26

Table 3. The number of independent algebraic letters of hexabox family (4.46) classified by square roots calculated by `BaikovLetter` (we have removed a spurious one which is related to a spurious square-root type leading singularity from the result). It can be noticed that algebraic letters related to $\sqrt{\Sigma_5^{(2)}}$ are reproduced. And this result agrees with CDE.

in the CDE. The numbers of algebraic letters, classified by square roots, are list in Tab. 3. Interestingly, we obtain the algebraic letters related to one of the square-root type leading singularities, $\Sigma_5^{(2)}$. They are precisely what we miss in DPzz family, which are related to $\Sigma_5^{(2)}$. The hexabox family and the family DPzz share the same subsectors from which $\Sigma_5^{(2)}$ originates. Let us examine the ansatz and the corresponding projections that yield the algebraic letters related to $\Sigma_5^{(2)}$[17]:

$$\begin{aligned}
&\text{ansatz: } \log\frac{G(\{k_2,p_2,p_{1234}\},\{k_2,p_3,p_{1234}\})+\sqrt{-G(k_2,p_{1234})G(k_2,p_2,p_3,p_{1234})}}{G(\{k_2,p_2,p_{1234}\},\{k_2,p_3,p_{1234}\})-\sqrt{-G(k_2,p_{1234})G(k_2,p_2,p_3,p_{1234})}}\\
&\text{pole: } \{\{x_2\to 0,x_3\to 0,x_4\to 0,x_5\to 0,x_6\to 0,x_7\to 0,x_8\to 0\},\\
&\qquad\qquad\qquad\qquad\{x_9\to -s_{12},\tilde{x}_{10}\to 2p_1^2,x_{11}\to 2s_{23}-s_{45}\}\},\\
&\text{letter: } \log W_{166}=\frac{-p_1^2+2s_{12}(s_{12}+s_{15}+s_{23}-2s_{34}-2s_{45})-s_{23}s_{34}+(p_1^2-s_{15}+s_{34})s_{45}-\Sigma_5^{(2)}}{-p_1^2+2s_{12}(s_{12}+s_{15}+s_{23}-2s_{34}-2s_{45})-s_{23}s_{34}+(p_1^2-s_{15}+s_{34})s_{45}-\Sigma_5^{(2)}}
\end{aligned}\tag{4.47}$$

where $p_{1234}\equiv p_1+p_2+p_3+p_4$. The Grams are only related to k_2, but $\tilde{x}_{10}$ is also involved in the expression. This is a general feature of nonplanar integral families. The key point is that this pole originates from the subsector $\{2,3,4,5,6,7,8\}$ which contains the common subsector $\{2,3,5,6,7,8\}$ but is not present in the family DPzz. This is a intriguing, as it has been found in [87] that for all-multiplicity nonplanar MHV amplitudes in $\mathcal{N}=4$ SYM, the cancellation of the Calabi-Yau six-cut (which is the cut of $\{2,3,5,6,7,8\}$ we studied here) requires the participation of the hexabox B family and the double pentagon A family,

[17]There are many different ansatz and projections that yield the same algebraic letters related to $\Sigma_5^{(2)}$. However, they are similar to each other and what we present here is a typical example. The same ansatz can be written for DPzz, but the algebaic letters are missed due to the absence of the corresponding pole (projection).

which inculde the two integral families we study here. Another integral family included is the DPmz family from the same paper [21]. This cancellation suggests that proper $\mathrm{d}\log$ system for the more general hexabox B and double pentagon A families may not correspond to each integral family individually, but rather to their combinations. Although for the two families considered here, we can indeed construct CDE for each of them. In summary, the identities between these families suggest that the missing of algebraic letters in DPzz may be due to the absence of certain projections from Baikov Gram determinants to leading singularities, as such a projection map can indeed be found in a related family. However, we still do not know how to find such a map directly in DPzz or DPmz by now.

We also analyzed other hexabox families with one massive leg. There are three configurations of the external mass, denoted as zmz, mzz and zzz in [20]. zzz family is what we present above in (4.46). For the zmz (mzz) family, we obtain 33 (28) rational and 23 (11) algebraic letters, which exactly match those in [20]. However, it should be noted that for these two-loop five-point nonplanar families, computation takes much longer time, which is in the same order as time consumed by three-loop families in the previous section.

5 Conclusion

This paper studies the symbol alphabets of various Feynman integral families that admit canonical differential equations with $\mathrm{d}\log$ type singularities. It primarily develops and elaborates on the techniques and workflow established in [70]. All these ideas are implemented in the `Mathematica` package `BaikovLetter` which can be downloaded from the following site

https://github.com/windfolgen/Baikovletter.git

The symbol alphabets of an integral family characterize important singularity structures of the associated function basis. They are expressions of kinematic variables (functions defined in kinematic space) and are naturally connected to the Landau singularities of the integral system. However, they exhibit finer structures, which are reflected in the appearance of algebraic letters. Generally, they allow for the inclusion of square roots and require that the expressions of symbol letters exhibit definite parity under the sign change of these square roots. The parity-even part consists of rational letters which are log of polynomials of kinematic variables and are directly related to the Landau singularities. The parity-odd part consists of algebraic letters, which acquire a minus sign when the sign of square roots is changed.

Therefore, we study the structure of symbol alphabets in two steps. The first step is to analyze all the leading singularities of the family (which we conjecture to encompass all Landau singularities relevant to $\mathrm{d}\log$ forms in CDE) by Baikov representations. This analysis will provide all the rational letters in the symbol alphabets and also imposes constraints on the algebraic letters. Calculating leading singularities in the Baikov representation is typically easier due to the simplicity of performing maximal cut, and the usage of loop-by-loop Baikov representations further simplifies the problem. All loop-by-loop representations can be derived from standard representation, forming a recursive structure. The relations between Gram determinants in different Baikov representations also serve as the starting point for our ansatz of algebraic letters. The calculation of leading singularities is then translated into the computation of multivariate residues of Baikov variables in different loop-by-loop representations. The second step is to reconstruct all the algebraic letters using Gram determinants in Baikov representation. The basic idea is that square-root-type leading singularities are all related to certain Gram determinants in the Baikov representation. They are calculated from these Gram determinants through a series of specific operations that sequentially remove or fix Baikov variables. This approach is similar to the linear reduction method developed in [88–90]. This also motivates us to extend the kinematic space into a larger space by incorporating Baikov variables. In this extended space, the Gram determinants in Baikov representations serve as the natural functions, and the exchange relations between determinants (3.2) significantly constrain the possible forms of the algebraic-type ansatz. Thus, we construct an ansatz using Gram determinants and then project them back into kinematic space to obtain the algebraic letters. This projection is achieved by fixing Baikov variables to the poles obtained when calculating multivariate residues. In this way, we recover the possible algebraic letters in the symbol alphabets of the integral family.

In addition to what have been presented in the main text, the appendix provides a simple tutorial for using the package `BaikovLetter`, as well as discussions on several interesting topics, such as the 4d limit of Baikov representation and the multivariate residue calculations in Baikov representations. We also note that, to make the package more efficient, certain compromises have been made, which may sacrifice some accuracy for the sake of efficiency. All these aspects are further elaborated in the appendix. Finally, although the leading singularity analysis is general and applicable to both planar and nonplanar families, our methods for algebraic letters appear insufficient for general nonplanar integral families. However, based on studies using Schubert analysis [67, 69], the algebraic letters in nonplanar families share a similar origin with those in planar families. Therefore, a major future direction is to understand how the singularity structure of nonplanar families differs from that of planar ones and to develop methods to extend our approach to include nonplanar families in a general manner.

A The four dimension limit of Baikov representation

In this section, we study a special case where the external legs are specified to four dimensions. Since we have assumed the external legs are in general dimensions, a naive idea is that we can directly take the four-dimensional limit of leading singularities obtained from the method used in package `BaikovLetter`. However, we will show that this may miss some leading singularities in this family because of the reason stated at the beginning of Sec. 2.2. Nevertheless, the missing leading singularity can actually be derived by taking the four-dimensional limit of Baikov representation in d dimension, and some of them can be detected by including more general analysis of leading singularities, as we have explained in the main text.

The example we discuss here is the Pentagon-Triangle family in [24]. Here we slightly adjust the label of propagators:

$$\begin{aligned}x_1 =&k_1^2,\ x_2=(k_1-p_1)^2,\ x_3=(k_1-p_1-p_2)^2,\ x_4=(k_1-p_1-p_2-p_3)^2,\ x_5=(k_1+k_2)^2,\\ x_6 =&(k_2+p_1+p_2+p_3+p_4)^2,\ x_7=(k_2-p_6)^2,\ x_8=(k_2+p_1+p_2+p_3)^2,\ x_9=k_2^2,\\ x_{10} =&(k_1-p_1-p_2-p_3-p_4)^2,\ x_{11}=(k_1+p_6)^2,\ x_{12}=(k_2+p_1)^2,\ x_{13}=(k_2+p_1+p_2)^2.\end{aligned} \tag{A.1}$$

Running the package `BaikovLetter` and comparing the results in four dimensions (that is, directly taking the four-dimensional limit using the momentum-twistor parameterization provided in the paper), we find that except for the following leading singularity in the top sector, the rest matches perfectly:

$$\begin{aligned}&s_{123}s_{234}s_{34}-s_{123}s_{234}s_{345}+s_{12}s_{234}s_{45}-s_{23}s_{34}s_{56}+s_{23}s_{345}s_{56}\\&\qquad\qquad -s_{123}s_{34}s_{61}+s_{123}s_{345}s_{61}-s_{12}s_{45}s_{61}.\end{aligned} \tag{A.2}$$

To discover how this is missed in our analysis of d-dimensional Baikov representation, we present the Baikov representation of top sector under maximal cut.

$$\begin{aligned}u =&G(p_1,p_2,p_3,p_4,p_5)^{1+\epsilon}G(k_1-p_1-p_2-p_3-p_4,k_2+p_1+p_2+p_3+p_4,p_5)^{-\epsilon}\\&\times G(k_1-p_1-p_2-p_3-p_4,p_5)^{-1/2+\epsilon}G(k_1,p_1,p_2+p_3,p_3,p_4,p_5)^{-3/2-\epsilon},\end{aligned} \tag{A.3}$$

where we suppressed the constant coefficient in the representation which is irrelevant to our discussion[18]. It is obvious that this representation breaks down when the external legs are in four dimensions since we have one addition constraint in four dimensions

$$G(p_1,p_2,p_3,p_4,p_5)=0. \tag{A.4}$$

In general, we need to rewrite the Baikov representation using an independent basis of external momenta. However, here we want to discuss how to take the proper limit of above d-dimensional expression to recover the four-dimensional result. In above representation, there are two ISPs, x_{10} and x_{11}, remaining. Since we have integrated out x_8, x_9, x_{12} and x_{13}, these four must be assumed independent from the beginning. So we choose the independent external momenta as $\{p_1,p_2,p_3,p_4\}$ which includes p_1,p_2,p_3[19]. Therefore, x_{10} and x_{11} are not independent now, and the same applies to x_6 and x_7. Their relationship can be solved as

$$x_{10}-x_{11}=2k_1\cdot p_5+s_{56}=2\sum_{i=1}^{4}\alpha_i k_1\cdot p_i+s_{56}. \tag{A.5}$$

where α_i can be solved as

$$\begin{aligned}\alpha_1 =&\frac{G(\{p_5,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})}{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})},\ \alpha_2=\frac{G(\{p_1,p_5,p_3,p_4\},\{p_1,p_2,p_3,p_4\})}{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})}\\ \alpha_3 =&\frac{G(\{p_1,p_2,p_5,p_4\},\{p_1,p_2,p_3,p_4\})}{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})},\ \alpha_4=\frac{G(\{p_1,p_2,p_3,p_5\},\{p_1,p_2,p_3,p_4\})}{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})}.\end{aligned} \tag{A.6}$$

[18] Actually we also suppressed the minus sign before $G(k_1-p_1-p_2-p_3-p_4,p_5)$ for simplicity since here we don't care the positive property under square root.

[19] We can also choose $\{p_1,p_2,p_3,p_5\}$ as the independent set. If we choose in such a way, the expressions for x_{10} and x_{11} will be a little simpler.

Under maximal cut of propagators, above expression will be further simplied to a relation between x_{10} and x_{11}. This relation can also be interpreted as

$$G(\{k_1, p_1, p_2, p_3, p_4\}, \{p_1, p_2, p_3, p_4, p_5\}) = 0$$

in four dimension.

Now we discuss how to obtain the four-dimensional limit from the d-dimensional representation. Although p_5 is now a four-dimensional vector that is not independent from p_1, p_2, p_3, p_4, we can still introduce a transverse component $p_5^\perp$ which satisfies

$$p_5^\perp \cdot p_i = 0,\; i = 1, 2, 3, 4, 5, 6. \tag{A.7}$$

when deriving the Baikov representation. Therefore, the p_5 in (A.3) is actually $\tilde{p}_5 = (p_5, p_5^\perp)$. There is actually some freedom for us to introduce an additional vector to parameterize loop momenta since k_i are still in $4 - 2\epsilon$ dimension. Now, using the identity of general determinants, we have

$$G(p_1, p_2, p_3, p_4, \tilde{p}_5) = G(p_1, p_2, p_3, p_4, p_5 - \sum_{i=1}^{4} \alpha_i p_i) = G(p_1, p_2, p_3, p_4, p_5^\perp). \tag{A.8}$$

Similarly,

$$G(k_1, p_1, p_2 + p_3, p_3, p_4, \tilde{p}_5) = G(k_1, p_1, p_2 + p_3, p_3, p_4, p_5^\perp). \tag{A.9}$$

Then the representation becomes

$$\begin{aligned} u =& G(p_1, p_2, p_3, p_4, p_5^\perp)^{1+\epsilon} G(k_1 - p_1 - p_2 - p_3 - p_4, k_2 + p_1 + p_2 + p_3 + p_4, \tilde{p}_5)^{-\epsilon} \\ & \times G(k_1 - p_1 - p_2 - p_3 - p_4, \tilde{p}_5)^{-1/2+\epsilon} G(k_1, p_1, p_2 + p_3, p_3, p_4, p_5^\perp)^{-3/2-\epsilon}. \end{aligned} \tag{A.10}$$

Taking four-dimensional limit means $p_5^\perp \to 0$ and should be seperated from the representation. Apparently, the singular terms in this limit are $G(p_1, p_2, p_3, p_4, p_5^\perp)$ and $G(k_1, p_1, p_2 + p_3, p_3, p_4, p_5^\perp)$. $\tilde{p}_5$ in the other two Grams can be set to p_5 straightforwardly since $p_5 \gg p_5^\perp$. Using $p_5^\perp \cdot p_5^\perp = \eta$ as the regulator, the representation now becomes

$$\begin{aligned} u =& \eta^{1+\epsilon} G(p_1, p_2, p_3, p_4)^{1+\epsilon} G(k_1 - p_1 - p_2 - p_3 - p_4, k_2 + p_1 + p_2 + p_3 + p_4, p_5)^{-\epsilon} \\ & \times G(k_1 - p_1 - p_2 - p_3 - p_4, p_5)^{-1/2+\epsilon} G(k_1, p_1, p_2 + p_3, p_3, p_4, p_5^\perp)^{-3/2-\epsilon}. \end{aligned} \tag{A.11}$$

Note that we have suppressed terms of higher order in η. Now we can integrate $k_1 \cdot p_5^\perp$ out in the representation:

$$\begin{gathered} G(k_1, p_1, p_2 + p_3, p_3, p_4, p_5^\perp)^{-3/2-\epsilon} \\ \Big\downarrow \text{ integrating } k_1 \cdot p_5^\perp \text{ out} \\ G(p_1, p_2, p_3, p_4)^{-1/2} G(k_1, p_1, p_2, p_3, p_4)^{-1-\epsilon} \eta^{-1-\epsilon} \end{gathered} \tag{A.12}$$

The η term will be cancelled by the coefficient as expected and we arrive at the final expression for four-dimensional representation:

$$\begin{aligned} u =& G(p_1, p_2, p_3, p_4)^{1/2+\epsilon} G(k_1 - p_1 - p_2 - p_3 - p_4, k_2 + p_1 + p_2 + p_3 + p_4, p_5)^{-\epsilon} \\ & \times G(k_1 - p_1 - p_2 - p_3 - p_4, p_5)^{-1/2+\epsilon} G(k_1, p_1, p_2, p_3, p_4)^{-1-\epsilon} + \mathcal{O}(\eta). \end{aligned} \tag{A.13}$$

Finally, let us explain more about why the four-dimensional representation will generate new leading singularity that cannot be seen from the d-dimensional representation. We analyze the leading singularities step by step in the top sector. Let us take the maximal cut for the following two Grams (which is the same for d dimensions and four dimensions in this step):

$$
\begin{aligned}
&G(k_1-p_1-p_2-p_3-p_4,p_5)|_{x_{1,\dots,7}=0}=-\frac{1}{4}(x_{10}-x_{11})^2,\\
&G(k_1-p_1-p_2-p_3-p_4,k_2+p_1+p_2+p_3+p_4,p_5)|_{x_{1,\dots,6}=0}=-\frac{1}{4}x_7x_{10}(x_7+x_{10}-x_{11}),
\end{aligned}
\tag{A.14}
$$

where for the second Gram, we take a next-to-maximal cut first. This Gram will be 0 under maximal cut, which usually means that this sector is reducible. We can study such cases by the next-to-maximal cut, which is equivalent to the study of subsectors. Since we can further cut x_7, this is equivalent to studying the maximal cut of this sector. This is a general phenomenon when studying the cut of singular algebraic surface. Now, in d dimensions, we can identify poles $(x_{10},x_{11})=(0,0)$ from the cut of above two polynomials, as we assume the independence of x_{10} and x_{11}. However, in four dimensions, because there is only one independent variable, we can identify two poles for x_{10} in above expression. One is determined by $x_{10}=0$ and the other is determined by $x_{10}-x_{11}=0$. The latter one is nontrivial and only arises in four dimensions. This is the main reason why some 4d leading singularities cannot be derived from d-dimensional leading singularities.

Another important difference comes from the last Gram determinant. The maximal cut of the remaining Gram in the representation will result in a quadratic polynomial of x_{10} and x_{11} in both d dimensions and four dimensions, which we don't present here for simplicity. The important difference is their power. One is half integer in d dimensions, and the other is an integer in four dimensions. In general, when the polynomial is under a square root, we do not consider its cut since it is not a pole but a branch cut. However, the subtlety is that, in four dimensions, this polynomial under square root is a perfect square. So it is actually a pole, but we cannot see it in d dimensions! Now we can directly use the representation (A.13) to obtain this singularity. We localize x_{10} to the pole determined by $x_{10}=x_{11}$ in four dimensions and substitute x_{10} in $G(k_1,p_1,p_2,p_3,p_4)$ with this solution. Then we can obtain the leading singularity missed:

$$
\begin{aligned}
&G(k_1,p_1,p_2,p_3,p_4)|^{4d}_{x_{10}=x_{11}}\propto s_{12}s_{23}\big(s_{123}s_{234}s_{34}-s_{123}s_{234}s_{345}+s_{12}s_{234}s_{45}-s_{23}s_{34}s_{56}\\
&+s_{23}s_{345}s_{56}-s_{123}s_{34}s_{61}+s_{123}s_{345}s_{61}-s_{12}s_{45}s_{61}\big)\Big(s_{12}^2s_{23}^2+s_{23}^2s_{34}^2+s_{23}^2s_{56}^2+s_{12}^2s_{234}^2\\
&+s_{34}^2s_{123}^2+s_{123}^2s_{234}^2-2s_{23}s_{234}s_{12}^2-2s_{123}s_{234}^2s_{12}-2s_{23}^2s_{34}s_{12}-2s_{23}^2s_{56}s_{12}-4s_{23}s_{34}s_{56}s_{12}\\
&+2s_{23}s_{34}s_{123}s_{12}+2s_{23}s_{34}s_{234}s_{12}+2s_{23}s_{56}s_{234}s_{12}+2s_{23}s_{123}s_{234}s_{12}+2s_{34}s_{123}s_{234}s_{12}\\
&-2s_{23}^2s_{34}s_{56}-2s_{23}s_{34}^2s_{123}+2s_{23}s_{34}s_{56}s_{123}-2s_{34}s_{123}^2s_{234}+2s_{23}s_{34}s_{123}s_{234}-2s_{23}s_{56}s_{123}s_{234}\Big)
\end{aligned}
\tag{A.15}
$$

Interestingly, we can also discover this singularity directly from the d-dimensional representation without taking four-dimensional limit. That is, we include the singularity from the discriminant of polynomials under the square root. This is a pinch condition for the contour related to the corresponding Grams under square roots. However, this singularity does not arise from the multivariate residue in the d-dimensional representation (it is not a pole). Thus it is missed in our algorithm, which calculates leading singularities from multivariate residues. It is not difficult to include such singularities in the package, and we provide an option `"GeneralPinch"->True` in the function `PolesAnalyze[]` to include such leading singularities. The default value of `"GeneralPinch"` is `False` since we don't need it in most cases, and we provide another option `"SpecialKinematics"` to specify the special kinematic settings and consider the possible degeneracy. For example, in the above example, we can specify this option as the momentum twistor parameterization for external kinematics provided in [24] and set the independent variable z_i to a set of numbers:

```
"SpecialKinematics"→ {s12→3343, s123→14097431,
s56→-101144400648014180/6397, s23→21385171, s234→-883546069298350,
s61→43061762914859939321121/6397, s34→-5920029308115389839/5623,
s345→51462364442153514401640/6397, s45→-793269876153748169999/6397}
```

Then the algorithm will detect whether some polynomials become perfect squares with the special kinematics and consider their pinches further.

However, not all the singularities in four dimensions can be detected using the above trick. The major reason is that the relationship between ISPs in 4d kinematics will generate new cut conditions which can never be seen in d dimensions. Let us take the supersector of above Pentagon-Triangle family, Double-Pentagon family studied in [23] as an example. $x_{1\sim 9}$ are the propagators. The d dimensional Baikov representation for the top sector of double pentagon is

$$\begin{aligned} u =&G(p_1,p_2,p_3,p_4,p_5)^{1+\epsilon}G(k_1,k_2,p_1+p_2+p_3,p_4,p_5)^{-1-\epsilon}\\ &\times G(k_1,p_1+p_2+p_3,p_4,p_5)^{1/2+\epsilon}G(k_1,p_1+p_2,p_2,p_3,p_4,p_5)^{-3/2-\epsilon}, \end{aligned} \tag{A.16}$$

Then the 4d limit of above Baikov representation (keep in mind that the loop momenta are still in d dimensions) can be derived in exactly the same way as before and results in

$$\begin{aligned} u =&G(p_1,p_2,p_3,p_4)^{1/2+\epsilon}G(k_1,k_2,p_1+p_2+p_3,p_4,p_5)^{-1-\epsilon}\\ &\times G(k_1,p_1+p_2+p_3,p_4,p_5)^{1/2+\epsilon}G(k_1,p_1,p_2,p_3,p_4)^{-1-\epsilon}+\mathcal{O}(\eta). \end{aligned} \tag{A.17}$$

Let us consider such a leading singularity in 6 dimension, that is, $\epsilon=-1$, under maximal cut. The differential form in $6-2\epsilon$ dimension can be written as

$$\begin{aligned} \Omega =&C\frac{\mathrm{d}x_{11}}{\sqrt{G(p_1,p_2,p_3,p_4)G(k_1,p_1+p_2+p_3,p_4,p_5)|_{\text{maximal cut}}}}\\ =&\frac{C}{\sqrt{G(p_1,p_2,p_3,p_4)}}\times\\ &\frac{\mathrm{d}x_{11}}{\sqrt{[(s_{45}-s_{123})x_{10}+(s_{56}-s_{123})x_{11}-s_{45}s_{56}]^2+4s_{123}(s_{45}+s_{56}-s_{123})x_{10}x_{11}}}. \end{aligned} \tag{A.18}$$

where C is a Jacobian factor when we transform the d-dimensional variables x_{10},x_{11} to $x_{11},k_1\cdot p_5^{\perp}$. $k_1\cdot p_5^{\perp}$ is then integrated out as we have done before. Its explicit expression is

$$C=\frac{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4\})}{G(\{p_1,p_2,p_3,p_4\},\{p_1,p_2,p_3,p_4+p_5\})}. \tag{A.19}$$

It can be derived from the linear relation satisfied by x_{10} and x_{11}:

$$G(\{k_1, p_1, p_2, p_3, p_4\}, \{p_1, p_2, p_3, p_4, p_5\})|_{\text{maximal cut}} = k_1 \cdot p_5^{\perp} G(p_1, p_2, p_3, p_4). \tag{A.20}$$

In 4d limit, the right-hand-side will be 0, and we solve x_{10} in terms of x_{11}. To obtain the leading singularity, we must substitute this solution into (A.18). In d dimensions, since x_{10} and x_{11} are independent, no such leading singularity can be found. Therefore, in general, we need to incorporate the relationships between ISPs, which are generated from identities like (A.20), into our calculation of multivariate residues. After this step, the leading singularity of (A.18), which can be calculated from the coefficient of x_{11}^2 under square root, is equivalent to F_4 defined in [23]. This equivalence is seen only after we substitute the 4d kinematics. In d dimensions, the relationships between ISPs like (A.20) can be represented in different ways, and this introduces some ambiguity about how to implement this relation. For example, here we choose p_1, p_2, p_3, p_4 as the independent momenta and decompose k_5 into k_5 in 4d and $k_5^{\perp}$. We can also choose p_1, p_2, p_3, p_5 as independent momenta and decompose p_4. Then the Jacobian factor (A.19) will also change correspondingly, though it will cancel in the final expression of leading singularities in the 4d limit. How to implement this into the package will be left for future work.

B Techniques about the package `BaikovLetter`

In this section, we discuss the workings of the `BaikovLetter` package in detail. It can be conceptually divided into two main components. The first component is a module that computes all rational letters for a given integral family. Within this package, all rational letters of an integral family are identified through the leading singularities of that family. We systematically examine the integral family, starting from its top sector and proceeding through all subsectors using Baikov representations generated by the package `BaikovAll` [71], which we will also briefly introduce here.

B.1 The package `BaikovAll`: generating Baikov representations for an integral family

The package `BaikovAll` generates Baikov representations for integral families using the methodology outlined in [71]. Within the package, all Baikov variables are denoted as x_i, and the name `x` is protected. Similarly, the variables `y,G,j,`ϵ`,R,Rlist` are also protected. Their meanings are detailed in Tab. 4. The basic intput of one integral family consists of a list of propagators denominators. These denominators are defined as

```
PD[k,p,-m2]=(k+p)^2-m2,  LPD[k,p,-m2]=k*p-m2.
```

The first argument in the propagator denominators `PD[],LPD[]` should be linear combinations of loop momenta, while the second argument is linear combinations of external momenta. The final argument is the constant term (or the mass term). For example, the sunrise diagram depicted in Fig. 12 is defined using the following syntax:

variable	usage in package
x	name for Baikov variables Subscript[x,i]
y	(reserved) name for Baikov variables
G	name for Gram determinant like G[$\{p_1, p_2\}, \{p_1, p_2\}$]
j	name for Feynman integrals like j[family,0,1,0,0,0]
R	name for abbreviated square roots like R[1].
Rlist	list for all the abbreviated square roots and their expressions.

Table 4. Some reserved name for variables in BaikovAll and BaikovLetter.

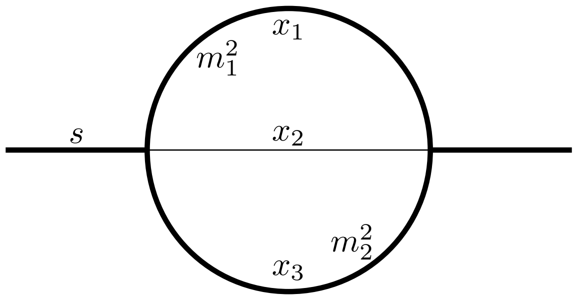

Figure 12. Sunrise diagram with two massive propagators of mass m_1 and m_2.

```
dlist = {PD[k1,0,-m12], PD[k1-k2,0,0], PD[k2,-p,-m22],
        PD[k2,0,0], PD[k1,-p,0]};
kinerep = {SProd[p,p] → s};
```

Usually, irreducible scalar products (ISPs) are appended to the end of dlist. The function SProd[p,q] defines the scalar product $p \cdot q$. Rreplacement rules for scalar products involving loop momenta can be calculated using BaikovTrans[dlist/.kinerep]. Typically, we define a comprehensive replacement rule for all scalar products as follows:

```
krep = Join[BaikovTrans[dlist/.kinerep][[2]],kinerep];
```

The standard Baikov representation for an integral family can then be computed using

```
intStd = BaikovRep[dlist, {k1,k2}, {p}, Abst→True];
```

The final option keep the Gram determinants in their abstract form G[...] without expanding them into explicit expressions. For instance, in the case of the sunrise, the output takes the form:

$$\{\{\{\texttt{G[\{p\},\{p\}]},-1+\epsilon\},\{\texttt{G[\{k1,k2,p\},\{k1,k2,p\}]},-\epsilon\}\},\frac{1}{8\pi\hat{}3/2\mathrm{Gamma}(1-\epsilon)\mathrm{Gamma}(3/2-\epsilon)}\}$$

To convert the abstract expression G[...] into polynomials of x_i, the following command can be used:

```
Gram2Poly[exp,krep]
```

which will replace all G[...] in exp to polynomials of x_i. The core function of this package is

```
allbaikov = AllSectorBaikovMat[intStd, krep, Exc→{}];
```

The input `intStd` should follow the format presented above. The option `Exc` specifies variables that **must not** be integrated out. This feature is useful when we want to obtain representations containing specific propagators. For instance, for the top sector of sunrise, we can specify

$$\texttt{Exc}\to\{x_1,x_2,x_3\}$$

which indicates x_1, x_2 or x_3 will never be integrated out and remain in the representations. Once the result `allbaikov`[20] is obtained, usually we want to extract the Baikov representation for specific sectors within the family. This extraction is accomplished using

```
rep = GetBaikovMatRep[allbaikov, sector, n];
```

The second argument `sector`, specifies the desired sector, while `n` is the total number of Baikov variables, which is the length of `dlist`. For example, to obtain representations for the top sector of sunrise, we would use

```
rep = GetBaikovMatRep[allbaikov, {1,2,3}, 5, "looporder"→ 2];
```

This command returns representations with as many Baikov variables integrated out as possible. Each returned representation represents the simplest form for this sector within different branches of the tree. The option `"looporder"` specifies the number of loops. For instance, in two-loop families, there are typically two inequivalent representations[21]. We will return both of them, even if one representation retains more Baikov variables (ISPs).

B.2 The rational letters: Leading singularity analysis in Baikov representation

One crucial module within `BaikovLetter` is dedicated to computing leading singularities in Baikov representation. The main methodology for calculating leading singularities in Baikov representations is outlined in Sec. 2.2. Here we mainly highlight certain subtleties in our algorithms and provide guidance on the usage of these functions.

The central function in this module is

```
polestructure = PolesAnalyze[allbaikov, top sector, krep, n];
```

The argument `allbaikov` is the output of `AllSectorBaikovMat[]`, as presented in the previous section. The argument `top sector` specifies the top sector within a given family. For instance, in the sunrise family, it will be specified as `{1,2,3}`. The argument `krep` is also defined as in the previous section, while `n` is the length of `dlist`, corresponding to the total number of Baikov variables. Within this funtion, we first generate all subsets of the top sector, which collectively represent all subsectors in this family. Subsequently, we systematically analyze the leading singularities of each subsector[22].

[20] All zero sectors in the result are identified by the scaling property of Baikov variables.

[21] They correspond to integrating variables out in different loop orders. While two-loop integrals generally yield two inequivalent representations, certain cases (reducible to products of one-loop integrals) may produce identical representations from different orders. In other cases, such as nonplanar ones, the number of inequivalent representations may exceed the naive loop-order count.

[22] It should be noted that all zero sectors are excluded.

```
for each sec
    Step 1: brep = GetBaikovMatRep[allbaikov, sec, n];
            perform maximal cut (of propagators) for brep;
```

For reducible sectors, a subtlety emerges as the polynomials (Gram determinants) in the Baikov representation vanish under maximal cuts. To address this issue, we first relax the maximal cut condition to a next-to-maximal cut in this case. Specifically, we cut all propagators except for one. This variable serves as a regulator. Under this next-to-maximal cut, the polynomials in the Baikov representations factorize for reducible sectors[23]. We can subsequently isolate the vanishing components and proceed with the analysis of the non-zero factorized polynomial parts as the final Baikov variable approaches zero. An illustrative example of this process is presented in (A.14).

```
    Step 2: for each rep in brep
                solve remaining Baikov variables (ISPs)
                by cutting Baikov polynomials (Jacobians);
```

The maximal cut in `Step 1` captures residues of propagators (i.e. $x_i = 0$ in the Baikov representation, where x_i appears in the denominator). In `Step 2`, we proceed to compute multivariate residues of ISPs[24] and fix all integration variables. As discussed in Sec. 2.2, the multivariate residues only capture leading singularities associated with simple poles. The pinching of polynomials under square roots may introduce additional leading singularities. Within the program, we provide one option in `PolesAnalyze[]`, `"GeneralPinch"`$\rightarrow$`True`, to account for pinches under square roots. However, since our primary focus is on studying the symbol alphabets of MPL integral families, of which singularities are associated with simple poles, this option is set to `False` by default. Under special kinematics, certain polynomials under square roots may degenerate into perfect squares, thereby becoming simple poles. To address such cases, we provide an additional option

`"SpecialKinematics"`$\rightarrow$`replacement rule for external kinematics`

where the replacement rule can be configured with numerical values that satisfy the special kinematic conditions. This option can be utilized to detect instances where polynomials under square roots degenerate. An illustrative example is provided at the end of App. A. All leading singularities computed by the package are derived from multivariate residues within the d-dimensional Baikov representations. Assuming the number of ISPs is m, when $m = 1$, the analysis is straightforward since residues of single variable are well-defined. However, for $m > 1$, certain subtleties arise. We will address these subtleties in the following discussion one by one.

First, we analyze the remaining ISPs in a specific order, under the assumption that genuine leading singularities are independent of this ordering. This is analogous to the

[23]The vanishing property induces the factorization for polynomials.

[24]In one-loop cases, ISPs are absent, and the maximal cut in `Step 1` directly yields the desired leading singularities. For higher loops, irreducible Feynman integrals will have ISPs, meaning the loop momenta cannot be fully determined by propagators alone.

linear reducibility analysis of polynomials in Feynman parameterization [88–90]. The distinction lies in our allowance for the presence of square roots of polynomials and quadratic polynomials. In practice, we choose such an order for the remaining ISPs that, in each step, there are as many linear polynomials of corresponding variable of this step as possible. The advantage of taking a specific order is that it allows us to treat this as a single-variable problem at each step. However, this approach also introduces certain drawbacks, such as the emergence of spurious poles in intermediate steps and other challenges, as we will discuss. The assumption that genuine leading singularities are independent of the integration variable order relies on the fact that, for well-defined multiple integrals[25], the integral's result should not depend on the order in which we integrate it. We now present an example where different analysis orders of Baikov variables yield distinct leading singularities. And the corresponding leading singularities are indeed spurious, as they do not appear in the canonical differential equations. Consider the following differential form in the integrand,

$$\begin{aligned}
\Omega =&\frac{\mathrm{d}x_9\wedge \mathrm{d}x_{10}}{P_1P_2\sqrt{P_3}}\\
P_1 =&[s_{12}s_{15}-s_{12}s_{23}-s_{15}s_{45}+s_{23}s_5+(s_{12}+s_{15}-s_5)x_9-(s_{12}+s_{23}-s_{45})x_{10}][s_{23}s_{34}+s_{15}x_9\\
&-s_{34}x_9-s_{23}x_{10}]+s_{12}(s_{23}-x_9)(s_{23}s_{34}+s_{15}s_4-s_{15}s_{34}-s_{34}x_9+s_{34}x_{10}-s_4x_{10}),\\
P_2 =&x_9\,x_{10},\\
P_3 =&s_4^2-2s_4x_9+x_9^2-2s_4x_{10}-2x_9x_{10}+x_{10}^2.
\end{aligned}\tag{B.1}$$

If we first perform a cut for x_9, taking the residue at $x_9=0$ in P_2, then $\sqrt{P_3}=x_{10}-s_4$ introduce a pole to x_{10}. After computing the residue around $x_{10}=s_4$, we obtain a leading singularity of the form

$$P_1(x_9=0,x_{10}=s_4)=s_{23}\left[s_{12}s_{23}s_4+(s_{34}-s_4)(s_{23}(s_5-s_4)-s_{45}(s_{15}-s_4))\right].\tag{B.2}$$

Conversely, if we first perform a cut at $x_{10}=0$, then $\sqrt{P_3}=x_9-s_4$, yielding a leading singularity of the form

$$P_1(x_9=s_4,x_{10}=0)=s_{12}s_{15}^2s_4+[s_{23}s_{34}+(s_{15}-s_{34})s_4][s_{15}(s_4-s_{45})+s_5(s_{23}-s_4)].\tag{B.3}$$

The presence of $\sqrt{P_3}$ is responsible for the different results. The spurious nature of these two "poles" $(x_9,x_{10})=(0,s_4)$ and $(x_9,x_{10})=(s_4,0)$, can be understood by noting that the pole $x_{10}=s_4$ (or $x_9=s_4$) exists only when $x_9=0$ ($x_{10}=0$). According to the definition of multivariate residue, along a small path encircling $x_9=0$, $\sqrt{P_3}$ is not a pole at all. This observation is further supported by variable transformations [52, 91].

$$x_9=-s_4u(v+1),\quad x_{10}=-s_4v(u+1).\tag{B.4}$$

Under this transformation, we obtain

$$\frac{\mathrm{d}x_9\wedge\mathrm{d}x_{10}}{\sqrt{P_3}}=s_4\mathrm{d}u\wedge\mathrm{d}v.\tag{B.5}$$

[25]In our context, all integrals are dimensionally regularized.

Thus, the pole arising from $\sqrt{P_3}$ is spurious, being merely an artifact due to that we follow a specific order in the analysis. By default, We retain such spurious leading singularities in intermediate steps and remove them using alternative methods, as discussed later. Nonetheless, we provide an option `SelectQ` within `PolesAnalyze[]`

`SelectQ`$\to$ `True`

This option can eliminate spurious poles of the aforementioned type during intermediate steps. However, it remains unclear whether all spurious leading singularities stem from this analysis, nor is it certain that such poles are always associated with spurious leading singularities. Thus, we default this option to `False`. It could be a nice try for examples with complicated representations, such as those encountered in certain three-loop cases. Finally, it is worthwhile noting that genuine leading singularities do depend on the order in which the denominators are grouped [92]. In our algorithms, we systematically explore all possibilities to ensure no leading singularities are overlooked. Let us consider the same example as in [92] to further illutrate this point:

$$\omega = \frac{z_1 \mathrm{d}z_1 \wedge \mathrm{d}z_2}{z_2(a_1 z_1 + a_2 z_2)(b_1 z_1 + b_2 z_2)}. \tag{B.6}$$

In our algorithms, we first choose an analysis order for the variables, in this case, $[z_2, z_1]$. Then we cut z_2 for each of the three propagators separately, yielding

$$\begin{aligned} z_2 = 0 &: \text{remaining denominator } a_1 z_1, b_1 z_1; \\ z_2 = -\frac{a_1 z_1}{a_2} &: \text{remaining denominator } a_1 z_1, (b_1 a_2 - a_1 b_2) z_1; \\ z_2 = -\frac{b_1 z_1}{b_2} &: \text{remaining denominator } b_1 z_1, (-b_1 a_2 + a_1 b_2) z_1; \end{aligned} \tag{B.7}$$

The analysis of z_1 is further carried out for each case. Since we are interested only in the factors of leading singularities, a_2, b_2 in the numerator are treated separately, resulting in the following factors of leading singularities

$$a_1, b_1, a_2, b_2, a_1 b_2 - b_1 a_2 \tag{B.8}$$

which is straightforwardly consistent with the multivariate residues of ω in [92].

Second, we elaborate on our approach to handling quadratic or higher-degree polynomials in intermediate step. As discussed in the main text, we partition the polynomials into two categories: those under square roots (algebraic set) and those not (rational set). When analyzing a Baikov variable x_i, if a polynomial in the rational has a degree in x_i greater than 2, we exclude it from further analysis, as it typically leads to cubic or higher roots, which are beyond the scope of our package[26]. For quadratic polynomial, we first attempt to factorize them into two linear polynomials in x_i[27]. Typically, these two linear

[26] In multivariate cases, the analysis order of x_i can usually be adjusted to avoid higher-degree polynomials in specific variables.

[27] It should be noted that, if x_i is the *only* remaining Baikov variable, this factorization is unnecessary, as the final leading singularity can be directly obtained from the discriminant of this polynomial.

polynomials will contain square roots. However, if the square root does not involve Baikov variables, (i.e., it is a square root of pure kinematic variables), we factorize the quadratic polynomial and abbreviate this square root to `R[i]`. If this square root involves remaining Baikov variables, then we refrain form factorization and instead use the discriminant of the polynomial for further analysis. For example, P_1 in (B.1) can be factorized as

```
 In:= EnhancedFactor[{P1},x9]
Out:= {(s12s15^2-s12s15s23+s15s23s34-s12s15s4-s15^2s45+s15s34s45+s15s23s5
       -2s23s34s5+s15R[1]+2s12s15x9+2s15^2x9-2s15s34x9-2s15s5x9+2s34s5x9
       -s12s15x10-s12s23x10-2s15s23x10+s23s34x10+s12s4x10+s15s45x10-s34s45x10
       +s23 s5 x10-R[1]x10)(s12s15^2-s12s15s23+s15s23s34-s12s15s4-s15^2s45
       +s15s34s45+s15s23s5-2s23s34s5-s15R[1]+2s12s15x9+2s15^2x9-2s15s34x9
       -2s15s5x9+2s34s5x9-s12s15x10-s12s23x10-2s15s23x10+s23s34x10+s12s4x10
       +s15s45x10-s34s45x10+s23s5x10+R[1]x10)}
```

Here, `R[1]` is actually a Gram determinant under square root and the explicit expressions of it can be found in `Rlist[[1]]` or by simply running `R[1]*R[1]`. If `R[i]` is permitted in the final result, it can be treated as a kinematic variable. However, leading singularities must exhibit definte parity under the exchange `R[i]`$\leftrightarrow$`-R[i]`, as the integral representation of Feynman integrals has definite parity. Therefore, the final leading singularities can only be monomials of `R[i]` or entirely free of `R[i]` ($\texttt{R[i]}^2$ will be replaced by its expressions automatically, so the power of `R[i]` can only be 0 or 1 modulo 2). In this example, the discriminant of P_1 is $\Delta = \texttt{R[1]}(x_{10} - s_{15})$. Therefore, if we perform a cut at $x_{10} = s_{15}$, the solution for x_9, x_{10} corresponds to the pinch condition for the polynomial P_1, which can also be directly solved as

$$\{\frac{\partial P_1}{\partial x_9} = 0,\, \Delta = 0\} \Rightarrow \{x_9 = s_{23}, x_{10} = s_{15}\}. \tag{B.9}$$

This solution simultaneously satisfies all pinch conditions

$$P_1 = 0, \quad \frac{\partial P_1}{\partial x_9} = 0, \quad \frac{\partial P_1}{\partial x_{10}} = 0, \tag{B.10}$$

This aligns with the necessary condition for singularity proposed in [72]. This demonstrates consistency between our variable-by-variable analysis and the approach that solves all variables simultaneously. The preceding discussion pertains to polynomials in the rational set. For quadratic or higher-degree polynomials under square roots (the algebraic part), if perfect squares are present, we factor them out and include them in the rational set. They may become simple poles. In other cases, we leave the polynomial under the square root and do not investigate its pinch, as previously mentioned. If only one Baikov variable remains and a cubic or quartic polynomial of this variable appears under square root, this indicates that elliptic integrals may be involved. In such cases, We output a warning message.

Finally, we address the most subtle case, where both a quadratic polynomial which can not be factorized (by `EnhancedFactor`) and a square root of the variable x_i appear in the denominator:

$$\frac{\mathrm{d}x_i}{P(x_i)\sqrt{Q(x_i)}} \propto \frac{\mathrm{d}x_i}{(x_i - c_1)(x_i - c_2)\sqrt{Q(x_i)}}. \tag{B.11}$$

where c_1, c_2 contain square roots in their expressions. If we naively compute the residue of x_i at c_1 or c_2, the leading singularities take the form,

$$\frac{1}{(c_1 - c_2)\sqrt{Q(c_1)}}, \quad \frac{1}{(c_1 - c_2)\sqrt{Q(c_2)}}. \tag{B.12}$$

where $Q(c_1)$ or $Q(c_2)$ lacks definite parity when the sign of the square roots in c_i is flipped. In general, we can combine these two leading singularities to obtain two leading singularities with definite parity. However, as noted in [51, 53], only when $P(x_i)$ has specific relations with $Q(x_i)$[28], does the linear combination yield reasonable overall leading singularities. This leading singularity analysis is equivalent to directly analyzing the multivariate residue of $\mathrm{d}x_i/P(x_i)$ while "ignoring" the presence of $\sqrt{Q(x_i)}$[29]. For general polynomials $P(x_i)$ and $Q(x_i)$, it is unclear whether this approach remains valid[30]. Thherefore, in our algorithms, we first detect any relations between $P(x_i)$ and $Q(x_i)$ in such cases. If no such relations detected, we skip the remaining analysis and output a warning message. In all cases we have encountered, scenarios where $P(x_i)$ has no relation with $Q(x_i)$ indeed can be overlooked.

Third, we discuss the removal of suprious leading singularities generated during intermediate steps.

```
Step 3: remove spurious leading singularities by criteria:
        1. genuine leading singularities can be detected
           in two inequivalent representations.
        2. spurious leading singularities can not induce the
           dimension drop for all the inequivalent representations.
```

We assume the validity of these two criteria. Let us elaborate on their implementation. First of all, for two-loop or higher-loop cases, there are usually more than one equivalent loop-by-loop representations of the same sector of an integral family. They arise from choosing specific orders to integrate out the loop momenta. In the recursive framework of Baikov representations, two representations are inequivalent if one cannot be derived from the other by integrating out certain Baikov variables, meaning they do not lie on the same path derived from the standard representations. Therefore, if two or more inequivalent representations exist for a sector, we first compute leading singularities from these different representations and take their intersection. However, this does not mean that leading singularities outside this intersection are not genuine, because they may be hidden. To decide whether they are genuine, we set them to 0[31] and calculate whether the dimension of other

[28] This relation ensures that $Q(c_1)$ becomes a perfect square and factors out of the square root.

[29] For example, if $\sqrt{Q(x_i)} = \sqrt{ax_i^2 + bx_i + c}$, only $\sqrt{a}$ as an overall factor contributes to the final leading singularity.

[30] It can happen that, after variable transformations, the leading singularities will be in a proper form.

[31] In practice, we use a set of rational numbers which satisfy the identity. For very complicated leading singularities (LS's), finding such non-zero rational numbers using function `FindInstance[]` in `Mathematica` can be challenging (Sometimes this function crashes randomly for reasons we don't know). So we implement a truncation: if rational solutions are not found in given time (120s), we include the singularity, though it may be spurious (a warning message will be generated in the same time). For almost all the cases, rational number solutions are found in given time. However, these numeric values may not be general enough,

representations drop. Here the dimension refers to the dimension of the cohomology group of a representation. This dimension can be computed using the method described in [81]. If a dimension drop occurs, we classify the singularity as genuine. This principle underlies the concept of principal Landau determinants in [79]. We examine all singularities outside the intersection set and add those genuine ones back. For efficiency, leading singularities within the intersection set are not subjected to dimension drop checks. They are pressumed to be genuine, although thiis may not always be the case. An example of this is provided in Sec. 4.5. However, such examples are rare, and thus the overall results are minimally affected.

Finally, we discuss certain compromises in the implementation of our algorithms. Given our variable-by-variable approach to the analysis, it is evident that a large number of remaining variables in the representation after cut of propagators can complicate the analysis a lot. Although we select an optimal order, as previously mentioned, the process can still be slow, particularly when 4 or more variables remain. Consequently, we must sacrifice some accuracy to obtain results within a reasonable timeframe. We impose a time constraint of 500 seconds for the leading singularity analysis of each representation. If the calculation does not complete within given time[32], we abandon the analysis of this representation. For a given sector, its independent loop-by-loop Baikov representations may be very different from each other. For example, one representation may have 2 remaining Baikov variables after all propagators are cut. In another representation, 4 variables may remain. In such cases, we still take the intersection of leading singularities calculated from these two representations and verify whether the remaining ones are genuine using methods in `Step 3`. However, but if the calculation for one representation is terminated due to time constraints, we disregard results from that representation.

Following the computation of `PolesAnalyze[]`, we provide a function to extract candidates for rational letters:

```
rationalset = AllRationalLetters[polestructure]
```

where `polestructure` is the output of `PolesAnalyze[]`. Among all the leading singularities computed by `PolesAnalyze[]`, the square-root type can be isolated using the function

```
permsqlist = ExtractSquareRoots[polestructure]
```

This provides a list of polynomials under square roots that may appear in algebraic letters.

B.3 The algebraic letters: constructing algebraic letter with Gram determinants

In this section, we outline the process of constructing an ansatz for algebraic letters and deriving their explicit expressions. Let us take the inner massive double box family in Sec. 4.1 as an example. First, after computing all the Baikov representations for this family

leading to the misidentification of some spurious leading singularities as genuine. Thus, some spurious letters may appear in the final results.

[32]For cases with at most 4 remaining variables, the analysis can be usually finished within this time.

using `AllSectorBaikovMat[]` (as detailed in App. B.1), we construct the ansatz A_1 and A_2 described in Sec. 3 using

```
algansatz = ExtractAlgLetter[allbaikov, sec, n]
```

where `allbaikov` is the output of `AllSectorBaikovMat[]`, and `sec` specifies the top sector under study. In this example, the top sector is `{1,2,3,4,5,6,7}`. The argument `n` is the number of all Baikov variables in the standard representation, which is `9` here. The ansatz A_3 is processed separately within the function `AllAlgLettersPL[]`, which will be introduced later. Next, we extract all linear poles from `polestructure`, the output of `PolesAnalyze[]` from the previous section.

```
poles = ExtractPoleInfo[polestructure,n]
```

The basic elements for linear poles take the form

$$\{\{x_2 \to 0, x_4 \to 0, x_5 \to 0, x_7 \to 0\}, \{x_1 \to 0, x_3 \to s\}\}$$

The first component specifies the cut condition, indicating the sector from which the solution originates. For example, the above solution comes from sector $\{2, 4, 5, 7\}$, a subsector with only four propagators. The second component provides solutions for the ISPs within this subsector. Since the first component is trivial, we will omit it in subsequent discussion for simplicity, focusing only on the second component. At this stage, we will simplify the poles calculated. The first simplification is related to our variable-by-variable solving strategy. It results in solutions such as

$$\{x_1 \to s + x_9,\ x_9 \to 0\} \tag{B.13}$$

We simplify this solution by substituting the solution of x_9 into x_1, yielding $\{x_1 \to s, x_9 \to 0\}$. However, subtleties arise when the solution for the latter variable causes the former variable to diverge. In such cases, the solving order may be important. For example, solutions may take the form:

$$\{x_1 \to \frac{-sx_9 + x_3x_9 - x_9^2}{x_3 - x_9}, x_9 \to 0, x_3 \to 0\} \tag{B.14}$$

An ambiguity arises regarding how the values of the last two variables are substituted into the first. If we first set $x_3 = 0$, we obtain $x_1 = s$, whereas setting $x_9 = 0$ first yields $x_1 = 0$. Whenever there is an ambiguity, we retain the original form of the solution[33]. In addition to the above solutions which correspond to the first type Landau singularities, we also have solutions corresponding to the second type Landau singularities. They are computed by first homogenizing Baikov polynomials via $x_i \to x_i/x_0$ and then setting $x_0 \to 0$. Such poles are recorded as follows

$$\{x_1 \to \infty, x_9 \to \infty, x_1 \to x_9, x_9 \to 0\} \tag{B.15}$$

[33]Interestingly, both solutions obtained from different approaches also appear in the solutions of its supersectors, such as $\{x_1 = s, x_9 = 0\}$ appearing in one of its supersectors.

This notation indicates that we first homogenize the polynomial with $x_1 \to x_1/x_0, x_9 \to x_9/x_0$ and then set $x_0 \to 0$. Due to the projective property of homogenized polynomials, only one degree of freedom exists between x_1 and x_9. The last term $x_9 \to 0$ does not imply setting x_9 to 0. It indicates that x_9 is the reserved variable, meaning all variables are rescaled by x_9. The second simplification is simply removing all duplicate solutions afterwards.

The next step involves substituting these linear poles into the ansatz we constructed. This is the most intricate step in our calculations. The subtleties mainly arise from three scenarios. The first one occurs when the ansatz degenerates upon substituting the solutions of Baikov variables into the ansatz. For example, consider the ansatz in the following expression

$$\log \frac{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) + \sqrt{-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}}{G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\}) - \sqrt{-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})}} \tag{B.16}$$

after substituting the solution, $G(\{\boldsymbol{q}, q_i\}, \{\boldsymbol{q}, q_j\})$ and $-G(\{\boldsymbol{q}\}, \{\boldsymbol{q}\})G(\{\boldsymbol{q}, q_i, q_j\}, \{\boldsymbol{q}, q_i, q_j\})$ may vanish simultaneously, but a common zero factor can cancel out, yielding a non-trivial algebraic letter. Thus, we preserve the order of variables in the solutions[34], substitute them sequentially and verify whether degenerate parts cancel in intermediate steps. This is consequently the most time-consuming one in our calculations.

The second subtlety stems from the compromise made in deriving the general projection from Baikov Gram determinants to leading singularities (Landau varieties). Our ansatz for algebraic letters may include Gram determinants in supersectors, which contain more Baikov variables than subsectors, as some Baikov variables have been integrated out. To project such Gram determinants, we should perform the leading singularities analysis in the representation of the supersector, such as the standard Baikov representation. However, our goal is to avoid direct analysis in the standard representation, which involves too many ISPs for subsectors. Instead, we start from different loop-by-loop representations, where distinct Baikov variables are integrated out, and then attempt to reconstruct the full solutions from these more manageable pieces. In summary, as discussed at the end of Sec. 2.1, solutions from independent representations are combined to provide the full solutions required. For example, in the top sector of inner massive double box, we obtain two distinct solutions from two independent loop-by-loop representations:

$$\begin{aligned} &\mathrm{S}_1 : \{\{x_1 \to 0, x_2 \to 0, x_3 \to 0, x_4 \to 0, x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_8 \to 0\}\}, \\ &\mathrm{S}_2 : \{\{x_1 \to 0, x_2 \to 0, x_3 \to 0, x_4 \to 0, x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_9 \to t\}\}. \end{aligned} \tag{B.17}$$

They can be combined to yield the solution

$$\mathrm{S}_{12} : \{\{x_1 \to 0, x_2 \to 0, x_3 \to 0, x_4 \to 0, x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_8 \to 0, x_9 \to t\}\}. \tag{B.18}$$

Here, S_{12} contains the full solutions for all 9 Baikov variables and provides a way to project the Gram determinant in standard Baikov representations, $G(k_1, k_2, p_1, p_2, p_3)$, onto kinematic varieties. S_{12} can actually also be solved by directly analyzing the multivariate

[34]The solving order of these variables is assumed to be the correct when substituting their values.

residues in the maximal cut of the standard representation, rather than the loop-by-loop representations:

$$u_{\text{std}}|_{\text{maximal cut}} \propto (m^2st^2 - 2m^2stx_9 - stx_8x_9 + sx_8^2x_9 + m^2sx_9^2 + sx_8x_9^2 + x_8^2x_9^2)^{-1-\epsilon} \quad \text{(B.19)}$$

We first solve for x_9 by cutting the quadratic polynomial in x_9. The square root of the discriminant of this polynomial

$$\sqrt{\Delta} = x_8\sqrt{s(4m^2st + 4m^2t^2 - st^2 + 2stx_8 - sx_8^2)}. \quad \text{(B.20)}$$

introduces a pole at $x_8 = 0$. The solution for x_9 then reduces to $m^2s(t - x_9)^2 = 0$, yielding $x_9 = t$. Alternatively, solving for x_8 first and then x_9 yields the same solution. This solution can also be derived using the pinch condition [72]

$$P = 0, \frac{\partial P}{\partial x_9} = 0, \frac{\partial P}{\partial x_8} = 0. \quad \text{(B.21)}$$

where $P \equiv m^2st^2 - 2m^2stx_9 - stx_8x_9 + sx_8^2x_9 + m^2sx_9^2 + sx_8x_9^2 + x_8^2x_9^2$ is the polynomial above. Thus, we successfully recover the solution S_{12} from the standard representation using solutions from loop-by-loop representations. However, this approach comes at a cost. First, if a spurious pole is identified in a loop-by-loop representation, then any combinations involving this pole are also spurious. That is, the combination may introduce additional spurious poles. Second, in the above example, the order in which S_1 and S_2 are combined is irrelevant. Both orders yield the same results, and they are all valid. However, there are cases where the order of combining S_1 and S_2 affects the final result. In such cases, since these two solutions are not derived in a single representation, we lack a preferred order and cannot determine which order corresponds to genuine solutions in higher representations. Therefore, all possible orders will be tried. Finally, we note that not all poles solved from loop-by-loop representations can be combined, only poles solved in the same sector (i.e., sharing the same cut) can be combined as above.

The third subtlety arises from the fact that there may be multiple ways to project the same Gram determinant onto the kinematic space. Returning to the earlier example, the Gram determinant $G(k_1, k_2, p_1, p_2, p_3)$ can be projected using the solution S_{12}. However, as discussed in Sec. 2.2.2, when x_9 has been integrated out from $G(k_1, k_2, p_1, p_2, p_3)$, it can be set to 0 for the ansatz related to this Gram determinant. This provides an alternative way to project the Gram determinant $G(k_1, k_2, p_1, p_2, p_3)$, which is supplementing the solution S_1 with $x_9 = 0$:

$$\mathrm{S}_1' : \{\{x_1 \to 0, x_2 \to 0, x_3 \to 0, x_4 \to 0, x_5 \to 0, x_6 \to 0, x_7 \to 0\}, \{x_8 \to 0, x_9 \to 0\}\}. \quad \text{(B.22)}$$

We employ the function `AllAlgLettersSupplementPL[]` to handle such projections separately. This function is incorporated into `AllAlgLettersPL[]`, which we will introduce soon.

All these calculations are performed by the function

```
algresult=AllAlgLettersPL[poles,algansatz,reform,krep,PermSq→permsqlist];
```

where `reform=ReformRep[allbaikov]` collects all the Gram determinants in the Baikov representations for the entire family. The argument `krep` specifies the replacement rules for scalar products

```
krep = Join[BaikovTrans[dlist/.kinerep][[2]],kinerep];
```

and the argument `permsqlist` contains the square-root type leading singularities computed earlier. This function utilizes the parallel evaluation in `Mathematica` and in general it will spead up the calculations. `algresult` is an intermediate output. To obtain all independent algebraic letters, we can use the function

```
algind = GetIndepAlgLetters[algresult, rationalset]
```

where `rationalset = AllRationalLetters[polestructure]` contains the leading singularities computed as candidates for rational letters. It is used to remove some algebraic letters which do not satisfy the constraints $\mathbb{T}_3$ outlined in Sec. 3. The output `algind` is organized according to the square roots presented in the algebraic letters. We also provide a function to identify linear relations between algebraic letters

```
FindLetterLinearRelation[{Log[...],...,Log[...]},Log[...]]
```

For instance,

$$\texttt{In:=FindLetterLinearRelation[\{Log[}\tfrac{4+v-\sqrt{16u+(4+v)^2}}{4+v+\sqrt{16u+(4+v)^2}}\texttt{],Log[}\tfrac{-4+v+\sqrt{16u+(4+v)^2}}{-4+v-\sqrt{16u+(4+v)^2}}\texttt{]\},}$$
$$\texttt{Log[}\tfrac{4+2u+v-\sqrt{16u+(4+v)^2}}{4+2u+v+\sqrt{16u+(4+v)^2}}\texttt{]]}$$

```
Out:={{1,1,-1}}
```

which implies

$$\log\left[\frac{4+v-\sqrt{16u+(4+v)^2}}{4+v+\sqrt{16u+(4+v)^2}}\right]+\log\left[\frac{-4+v+\sqrt{16u+(4+v)^2}}{-4+v-\sqrt{16u+(4+v)^2}}\right] \\ -\log\left[\frac{4+2u+v-\sqrt{16u+(4+v)^2}}{4+2u+v+\sqrt{16u+(4+v)^2}}\right]=0. \quad \text{(B.23)}$$

Finally, when both `algresult` and `polestructure` are computed, we provide a function to trace the origin of a specific letter.

```
LetterInfo[letter, algresult, polestructure]
```

If the `letter` is a rational letter, then `algresult` can be set to `{}`. Usually, a single letter can originate from multiple sources. However, certain special letters may be associated with unique sectors or poles.

Acknowledgments

Note added. This is also a good position for notes added after the paper has been written.

References

[1] A.V. Kotikov, *Differential equations method: New technique for massive Feynman diagrams calculation*, *Phys. Lett. B* **254** (1991) 158.

[2] A.V. Kotikov, *Differential equation method: The Calculation of N point Feynman diagrams*, *Phys. Lett. B* **267** (1991) 123.

[3] T. Gehrmann and E. Remiddi, *Differential equations for two loop four point functions*, *Nucl. Phys. B* **580** (2000) 485 [hep-ph/9912329].

[4] J.M. Henn, *Multiloop integrals in dimensional regularization made simple*, *Phys. Rev. Lett.* **110** (2013) 251601 [1304.1806].

[5] S. Caron-Huot and J.M. Henn, *Iterative structure of finite loop integrals*, *JHEP* **06** (2014) 114 [1404.2922].

[6] J.M. Henn, *Lectures on differential equations for Feynman integrals*, *J. Phys. A* **48** (2015) 153001 [1412.2296].

[7] K.G. Chetyrkin and F.V. Tkachov, *Integration by Parts: The Algorithm to Calculate beta Functions in 4 Loops*, *Nucl. Phys. B* **192** (1981) 159.

[8] F.V. Tkachov, *A Theorem on Analytical Calculability of Four Loop Renormalization Group Functions*, *Phys. Lett. B* **100** (1981) 65.

[9] S. Laporta, *High precision calculation of multiloop Feynman integrals by difference equations*, *Int. J. Mod. Phys. A* **15** (2000) 5087 [hep-ph/0102033].

[10] S. Laporta, *Calculation of master integrals by difference equations*, *Phys. Lett. B* **504** (2001) 188 [hep-ph/0102032].

[11] M.G. Kozlov and R.N. Lee, *One-loop pentagon integral in d dimensions from differential equations in ϵ-form*, *JHEP* **02** (2016) 021 [1512.01165].

[12] M.G. Kozlov, *One-loop pentagon integral with one off-shell leg in 6-2ε dimensions*, *Phys. Rev. D* **95** (2017) 036008 [1612.03565].

[13] J.M. Henn, A. Matijašić and J. Miczajka, *One-loop hexagon integral to higher orders in the dimensional regulator*, *JHEP* **01** (2023) 096 [2210.13505].

[14] T. Gehrmann, J.M. Henn and N.A. Lo Presti, *Analytic form of the two-loop planar five-gluon all-plus-helicity amplitude in QCD*, *Phys. Rev. Lett.* **116** (2016) 062001 [1511.05409].

[15] T. Gehrmann, J.M. Henn and N.A. Lo Presti, *Pentagon functions for massless planar scattering amplitudes*, *JHEP* **10** (2018) 103 [1807.09812].

[16] D. Chicherin, T. Gehrmann, J.M. Henn, P. Wasser, Y. Zhang and S. Zoia, *All Master Integrals for Three-Jet Production at Next-to-Next-to-Leading Order*, *Phys. Rev. Lett.* **123** (2019) 041603 [1812.11160].

[17] D. Chicherin and V. Sotnikov, *Pentagon Functions for Scattering of Five Massless Particles*, *JHEP* **20** (2020) 167 [2009.07803].

[18] C.G. Papadopoulos, D. Tommasini and C. Wever, *The Pentabox Master Integrals with the Simplified Differential Equations approach*, *JHEP* **04** (2016) 078 [1511.09404].

[19] S. Abreu, H. Ita, F. Moriello, B. Page, W. Tschernow and M. Zeng, *Two-Loop Integrals for Planar Five-Point One-Mass Processes*, *JHEP* **11** (2020) 117 [2005.04195].

[20] S. Abreu, H. Ita, B. Page and W. Tschernow, *Two-loop hexa-box integrals for non-planar five-point one-mass processes*, JHEP **03** (2022) 182 [2107.14180].

[21] S. Abreu, D. Chicherin, H. Ita, B. Page, V. Sotnikov, W. Tschernow et al., *All Two-Loop Feynman Integrals for Five-Point One-Mass Scattering*, Phys. Rev. Lett. **132** (2024) 141601 [2306.15431].

[22] S. Abreu, D. Chicherin, V. Sotnikov and S. Zoia, *Two-loop five-point two-mass planar integrals and double Lagrangian insertions in a Wilson loop*, JHEP **10** (2024) 167 [2408.05201].

[23] J. Henn, T. Peraro, Y. Xu and Y. Zhang, *A first look at the function space for planar two-loop six-particle Feynman integrals*, JHEP **03** (2022) 056 [2112.10605].

[24] J.M. Henn, A. Matijašić, J. Miczajka, T. Peraro, Y. Xu and Y. Zhang, *A computation of two-loop six-point Feynman integrals in dimensional regularization*, 2403.19742.

[25] J. Henn, A. Matijašića, J. Miczajka, T. Peraro, Y. Xu and Y. Zhang, *Complete function space for planar two-loop six-particle scattering amplitudes*, 2501.01847.

[26] S. Abreu, P.F. Monni, B. Page and J. Usovitsch, *Planar Six-Point Feynman Integrals for Four-Dimensional Gauge Theories*, 2412.19884.

[27] D.D. Canko and N. Syrrakos, *Planar three-loop master integrals for 2 → 2 processes with one external massive particle*, JHEP **04** (2022) 134 [2112.14275].

[28] J.M. Henn, J. Lim and W.J. Torres Bobadilla, *First look at the evaluation of three-loop non-planar Feynman diagrams for Higgs plus jet production*, JHEP **05** (2023) 026 [2302.12776].

[29] M.-M. Long, *Three-loop ladder diagrams with two off-shell legs*, JHEP **01** (2025) 018 [2410.15431].

[30] D. Canko and M. Pozzoli, *A first computation of three-loop master integrals for the production of two off-shell vector bosons with different masses*, 2412.06972.

[31] Y. Liu, A. Matijašić, J. Miczajka, Y. Xu, Y. Xu and Y. Zhang, *An Analytic Computation of Three-Loop Five-Point Feynman Integrals*, 2411.18697.

[32] K.-T. Chen, *Iterated path integrals*, Bull. Am. Math. Soc. **83** (1977) 831.

[33] A.B. Goncharov, *Multiple polylogarithms, cyclotomy and modular complexes*, Math. Res. Lett. **5** (1998) 497 [1105.2076].

[34] A.B. Goncharov, *Multiple polylogarithms and mixed Tate motives*, math/0103059.

[35] F.C.S. Brown, *Multiple zeta values and periods of moduli spaces M 0 ,n (R)*, *Annales Sci. Ecole Norm. Sup.* **42** (2009) 371 [math/0606419].

[36] A.B. Goncharov, M. Spradlin, C. Vergu and A. Volovich, *Classical Polylogarithms for Amplitudes and Wilson Loops*, Phys. Rev. Lett. **105** (2010) 151605 [1006.5703].

[37] A.B. Goncharov, *A simple construction of Grassmannian polylogarithms*, 0908.2238.

[38] C. Duhr, *Hopf algebras, coproducts and symbols: an application to Higgs boson amplitudes*, JHEP **08** (2012) 043 [1203.0454].

[39] M. Argeri, S. Di Vita, P. Mastrolia, E. Mirabella, J. Schlenk, U. Schubert et al., *Magnus and Dyson Series for Master Integrals*, JHEP **03** (2014) 082 [1401.2979].

[40] T. Gehrmann, A. von Manteuffel, L. Tancredi and E. Weihs, *The two-loop master integrals for $q\bar{q} \to VV$*, *JHEP* **06** (2014) 032 [1404.4853].

[41] R.N. Lee, *Reducing differential equations for multiloop master integrals*, *JHEP* **04** (2015) 108 [1411.0911].

[42] L. Adams, E. Chaubey and S. Weinzierl, *Simplifying Differential Equations for Multiscale Feynman Integrals beyond Multiple Polylogarithms*, *Phys. Rev. Lett.* **118** (2017) 141602 [1702.04279].

[43] R.N. Lee and A.A. Pomeransky, *Normalized Fuchsian form on Riemann sphere and differential equations for multiloop integrals*, 1707.07856.

[44] C. Dlapa, J. Henn and K. Yan, *Deriving canonical differential equations for Feynman integrals from a single uniform weight integral*, *JHEP* **05** (2020) 025 [2002.02340].

[45] O. Gituliar and V. Magerya, *Fuchsia: a tool for reducing differential equations for Feynman master integrals to epsilon form*, *Comput. Phys. Commun.* **219** (2017) 329 [1701.04269].

[46] M. Prausa, *epsilon: A tool to find a canonical basis of master integrals*, *Comput. Phys. Commun.* **219** (2017) 361 [1701.00725].

[47] C. Meyer, *Algorithmic transformation of multi-loop master integrals to a canonical basis with CANONICA*, *Comput. Phys. Commun.* **222** (2018) 295 [1705.06252].

[48] R.N. Lee, *Libra: A package for transformation of differential systems for multiloop integrals*, *Comput. Phys. Commun.* **267** (2021) 108058 [2012.00279].

[49] E. Herrmann and J. Parra-Martinez, *Logarithmic forms and differential equations for Feynman integrals*, *JHEP* **02** (2020) 099 [1909.04777].

[50] J. Henn, B. Mistlberger, V.A. Smirnov and P. Wasser, *Constructing d-log integrands and computing master integrals for three-loop four-particle scattering*, *JHEP* **04** (2020) 167 [2002.09492].

[51] J. Chen, X. Jiang, X. Xu and L.L. Yang, *Constructing canonical Feynman integrals with intersection theory*, *Phys. Lett. B* **814** (2021) 136085 [2008.03045].

[52] C. Dlapa, X. Li and Y. Zhang, *Leading singularities in Baikov representation and Feynman integrals with uniform transcendental weight*, *JHEP* **07** (2021) 227 [2103.04638].

[53] J. Chen, X. Jiang, C. Ma, X. Xu and L.L. Yang, *Baikov representations, intersection theory, and canonical Feynman integrals*, *JHEP* **07** (2022) 066 [2202.08127].

[54] M. Spradlin and A. Volovich, *Symbols of One-Loop Integrals From Mixed Tate Motives*, *JHEP* **11** (2011) 084 [1105.2024].

[55] N. Arkani-Hamed and E.Y. Yuan, *One-Loop Integrals from Spherical Projections of Planes and Quadrics*, 1712.09991.

[56] S. Abreu, R. Britto, C. Duhr and E. Gardi, *Diagrammatic Hopf algebra of cut Feynman integrals: the one-loop case*, *JHEP* **12** (2017) 090 [1704.07931].

[57] J. Chen, C. Ma and L.L. Yang, *Alphabet of one-loop Feynman integrals **, *Chin. Phys. C* **46** (2022) 093104 [2201.12998].

[58] S. Caron-Huot and A. Pokraka, *Duals of Feynman integrals. Part I. Differential equations*, *JHEP* **12** (2021) 045 [2104.06898].

[59] T. Dennen, M. Spradlin and A. Volovich, *Landau Singularities and Symbology: One- and Two-loop MHV Amplitudes in SYM Theory*, *JHEP* **03** (2016) 069 [1512.07909].

[60] I. Prlina, M. Spradlin and S. Stanojevic, *All-loop singularities of scattering amplitudes in massless planar theories*, *Phys. Rev. Lett.* **121** (2018) 081601 [1805.11617].

[61] H.S. Hannesdottir, A.J. McLeod, M.D. Schwartz and C. Vergu, *Implications of the Landau equations for iterated integrals*, *Phys. Rev. D* **105** (2022) L061701 [2109.09744].

[62] L. Lippstreu, M. Spradlin and A. Volovich, *Landau Singularities of the 7-Point Ziggurat I*, 2211.16425.

[63] C. Dlapa, M. Helmer, G. Papathanasiou and F. Tellander, *Symbol Alphabets from the Landau Singular Locus*, 2304.02629.

[64] Q. Yang, *Schubert problems, positivity and symbol letters*, *JHEP* **08** (2022) 168 [2203.16112].

[65] S. He, Z. Li, R. Ma, Z. Wu, Q. Yang and Y. Zhang, *A study of Feynman integrals with uniform transcendental weights and their symbology*, *JHEP* **10** (2022) 165 [2206.04609].

[66] R. Morales, A. Spiering, M. Wilhelm, Q. Yang and C. Zhang, *Bootstrapping Elliptic Feynman Integrals Using Schubert Analysis*, *Phys. Rev. Lett.* **131** (2023) 041601 [2212.09762].

[67] S. He, J. Liu, Y. Tang and Q. Yang, *The symbology of Feynman integrals from twistor geometries*, 2207.13482.

[68] S. He, X. Jiang, J. Liu and Q. Yang, *On symbology and differential equations of Feynman integrals from Schubert analysis*, 2309.16441.

[69] S. He, X. Jiang, J. Liu and Q. Yang, *Landau-based Schubert analysis*, 2410.11423.

[70] X. Jiang, J. Liu, X. Xu and L.L. Yang, *Symbol letters of Feynman integrals from Gram determinants*, 2401.07632.

[71] X. Jiang and L.L. Yang, *Recursive structure of Baikov representations: Generics and application to symbology*, *Phys. Rev. D* **108** (2023) 076004 [2303.11657].

[72] S. Caron-Huot, M. Correia and M. Giroux, *Recursive Landau Analysis*, 2406.05241.

[73] L.D. Landau, *On analytic properties of vertex parts in quantum field theory*, *Nucl. Phys.* **13** (1959) 181.

[74] R.J. Eden, P.V. Landshoff, D.I. Olive and J.C. Polkinghorne, *The analytic S-matrix*, Cambridge Univ. Press, Cambridge (1966).

[75] A.V.Z. Israel M. Gelfand, Mikhail M. Kapranov, *Discriminants, Resultants, and Multidimensional Determinants*, Birkhäuser Boston, MA (2009).

[76] S. Mizera and S. Telen, *Landau discriminants*, *JHEP* **08** (2022) 200 [2109.08036].

[77] R.P. Klausen, *Kinematic singularities of Feynman integrals and principal A-determinants*, *JHEP* **02** (2022) 004 [2109.07584].

[78] C. Fevola, S. Mizera and S. Telen, *Principal Landau Determinants*, 2311.16219.

[79] C. Fevola, S. Mizera and S. Telen, *Landau Singularities Revisited: Computational Algebraic Geometry for Feynman Integrals*, *Phys. Rev. Lett.* **132** (2024) 101601 [2311.14669].

[80] M. Helmer, G. Papathanasiou and F. Tellander, *Landau Singularities from Whitney Stratifications*, 2402.14787.

[81] R.N. Lee and A.A. Pomeransky, *Critical points and number of master integrals*, JHEP **11** (2013) 165 [1308.6676].

[82] P.A. Baikov, *Explicit solutions of the multiloop integral recurrence relations and its application*, Nucl. Instrum. Meth. A **389** (1997) 347 [hep-ph/9611449].

[83] V. Chestnov, H. Frellesvig, F. Gasparotto, M.K. Mandal and P. Mastrolia, *Intersection numbers from higher-order partial differential equations*, JHEP **06** (2023) 131 [2209.01997].

[84] S. Badger, M. Becchetti, E. Chaubey and R. Marzucca, *Two-loop master integrals for a planar topology contributing to $pp \to t\bar{t}j$*, JHEP **01** (2023) 156 [2210.17477].

[85] S. Badger, M. Becchetti, N. Giraudo and S. Zoia, *Two-loop integrals for $t\bar{t}$+jet production at hadron colliders in the leading colour approximation*, JHEP **07** (2024) 073 [2404.12325].

[86] D. Chicherin, J. Henn, J. Trnka and S.-Q. Zhang, *Positivity properties of five-point two-loop Wilson loops with Lagrangian insertion*, 2410.11456.

[87] J.L. Bourjaily, E. Herrmann, C. Langer, A.J. McLeod and J. Trnka, *All-Multiplicity Nonplanar Amplitude Integrands in Maximally Supersymmetric Yang-Mills Theory at Two Loops*, Phys. Rev. Lett. **124** (2020) 111603 [1911.09106].

[88] F. Brown, *The Massless higher-loop two-point function*, Commun. Math. Phys. **287** (2009) 925 [0804.1660].

[89] F.C.S. Brown, *On the periods of some Feynman integrals*, 0910.0114.

[90] E. Panzer, *Algorithms for the symbolic integration of hyperlogarithms with applications to Feynman integrals*, Comput. Phys. Commun. **188** (2015) 148 [1403.3385].

[91] M. Besier, P. Wasser and S. Weinzierl, *RationalizeRoots: Software Package for the Rationalization of Square Roots*, Comput. Phys. Commun. **253** (2020) 107197 [1910.13251].

[92] K.J. Larsen and R. Rietkerk, *MultivariateResidues: a Mathematica package for computing multivariate residues*, Comput. Phys. Commun. **222** (2018) 250 [1701.01040].