

Complete integration-by-parts reductions of the non-planar hexagon-box via module intersections

Janko Böhm,^a Alessandro Georgoudis,^b Kasper J. Larsen,^c Hans Schönemann^a
and Yang Zhang^{d,e,f}

^aDepartment of Mathematics, University of Kaiserslautern,
67663 Kaiserslautern, Germany

^bDepartment of Physics and Astronomy, Uppsala University,
SE-75108 Uppsala, Sweden

^cSchool of Physics and Astronomy, University of Southampton,
Highfield, Southampton, SO17 1BJ, United Kingdom

^dPRISMA Cluster of Excellence, Johannes Gutenberg University,
55128 Mainz, Germany

^eInstitute for Theoretical Physics, ETH Zürich,
CH 8093 Zürich, Switzerland

^fKavli Institute for Theoretical Physics, University of California,
Santa Barbara, CA 93106, U.S.A.

E-mail: boehm@mathematik.uni-kl.de,

Alessandro.Georgoudis@physics.uu.se, Kasper.Larsen@soton.ac.uk,

hannes@mathematik.uni-kl.de, zhang@uni-mainz.de

ABSTRACT: We present the powerful module-intersection integration-by-parts (IBP) method, suitable for multi-loop and multi-scale Feynman integral reduction. Utilizing modern computational algebraic geometry techniques, this new method successfully trims traditional IBP systems dramatically to much simpler integral-relation systems on unitarity cuts. We demonstrate the power of this method by explicitly carrying out the complete analytic reduction of two-loop five-point non-planar hexagon-box integrals, with degree-four numerators, to a basis of 73 master integrals.

KEYWORDS: Differential and Algebraic Geometry, Scattering Amplitudes, Perturbative QCD

ARXIV EPRINT: [1805.01873](https://arxiv.org/abs/1805.01873)

Contents

1	Introduction	1
2	Simplification of IBP systems by the module-intersection method	3
2.1	Module intersection method without unitarity cuts	3
2.2	Module intersection method with cuts	6
2.3	Algorithm for computing individual modules	8
3	Determining the module intersection	9
4	Sparse row reduction	13
4.1	Selection of relevant and independent IBP identities	13
4.2	Sparse REF and RREF strategies	14
5	The non-planar hexagon-box diagram example	14
5.1	Module intersection on cuts	19
5.2	Reduction of IBP identities on cuts	22
5.3	Merging of IBP reductions and final result	23
5.4	Comparison with other IBP solvers	24
6	Conclusion and outlook	24

1 Introduction

High precision in the theoretical predictions for cross sections is necessary for the analysis of the Large Hadron Collider (LHC) Run II experiments. For many processes, the next-to-next-to-leading-order (NNLO) contributions are needed for precision-level predictions. NNLO in general requires the computation of a large number of two-loop (or, in some cases, higher-loop) Feynman integrals, which is often a bottleneck problem for particle physics.

Integration-by-parts (IBP) reduction is a key tool to reduce the large number of multi-loop Feynman integrals to a small integral basis of so-called master integrals. Schematically, for an L -loop integral in dimensional regularization with the inverse propagators D_1, \dots, D_m , we have

$$0 = \int \frac{d^D \ell_1}{i\pi^{D/2}} \cdots \frac{d^D \ell_L}{i\pi^{D/2}} \sum_{j=1}^L \frac{\partial}{\partial \ell_j^\mu} \frac{v_j^\mu}{D_1^{\nu_1} \cdots D_m^{\nu_m}}, \quad (1.1)$$

where the v_j^μ are vectors constructed from the external momenta and the loop momenta. Expanding the action of the derivative in eq. (1.1) leads to an IBP identity, which is a linear relation between multi-loop Feynman integrals. After collecting a sufficient set of IBP identities, one can carry out reduced row reduction in a specific integral ordering to express a target set of integrals as a linear combination of the master integrals.

The row reduction of IBP identities can be achieved by the Laporta algorithm [1, 2]. There are several publicly available implementations of IBP reductions: AIR [3], FIRE [4, 5], Reduze [6, 7], LiteRed [8], Kira [9], as well as various private implementations. In the standard Laporta algorithm and many of its variants, integrals with doubled propagators appear in the intermediate steps and also in the master integral basis. To trim the linear system of IBP identities for row reduction, one may impose the condition that no integrals with doubled propagators appear in the IBP identities [10]. This condition can be solved by the syzygy computation [10] or linear algebra [11]. In refs. [12, 13], methods for deriving IBP reductions on generalized-unitarity cuts was introduced. IBP reduction can also be sped up by the finite-field sampling method [14]. For several multiple-scale Feynman integrals, inspired choices of IBP generating vectors v_j^μ , leading to simple IBP identities, can be obtained from dual conformal symmetry [15]. In a recent development, IBP identities which target integrals with arbitrary-degree numerators [16] were introduced. An initial (and simpler) step to carrying out the IBP reductions is that of determining the integral basis itself. This can be done by the packages MINT [17] or AZURITE [18]. The dimension of the integral basis can be efficiently determined by the computation of an Euler characteristic which naturally arises in the D-module theory of polynomial annihilators [19].

IBP reductions, combined with generalized-unitarity methods [20–23] or integrand reduction methods [24–27], have led to successful computations of many complicated multi-loop integrands. Furthermore, IBP reductions are also important for setting up the differential equations for Feynman integrals [28–38], which has proven a highly efficient method for evaluating Feynman integrals. Recent developments in multi-loop integral reduction and the differential equation method have produced very impressive results for the two-loop amplitudes of massless $2 \rightarrow 3$ scattering processes with the all-plus-helicity configuration [39–41] and the generic-helicity configuration [42–45]. Furthermore, there has been important recent progress on the master integrals for in both the planar and non-planar sectors [46–48]. IBP reductions are also crucial for deriving dimension recursion relations [49–51].

For multi-loop, high-multiplicity or multi-scale amplitude computations, the IBP reductions frequently become the bottleneck which requires extensive computing resources and time. Schematically, the complexity of IBP reductions mainly originate from the following facts,

1. Multi-loop IBP identities involve many contributing integrals, and the complexity of row reduction grows quickly with the number of integrals.
2. There are many external invariants (i.e., Mandelstam variables and mass parameters), and the algebraic computation of polynomials or rational functions in these parameters requires a substantial amount of CPU time and RAM.

In this paper, utilizing our new powerful module-intersection IBP reduction method, we present solutions to the above problems:

1. With the module-intersection computation [52], we dramatically reduce the number of relevant IBP identities and integrals involved by imposing the no-doubled-propagator condition and applying unitarity cuts.

2. With the classical technique of treating parameters as new variables¹ and imposing a special monomial ordering, originating from primary decomposition algorithms [53] in commutative algebra, we can efficiently compute analytic IBP reductions with many parameters.

In this paper, we demonstrate our method by treating the cutting-edge example of the analytic IBP reduction of two-loop five-point non-planar hexagon-box integrals, with all degree-1, 2, 3, 4 numerators, to the basis of 73 master integrals. This computation was carried out using private implementations in SINGULAR [54] and MATHEMATICA of the ideas presented in this paper. Our result of fully analytic IBP reductions can be downloaded from,

https://github.com/yzhphy/hexagonbox_reduction/releases/download/1.0.0/hexagon_box_degree_4_Final.zip.

This paper is organized as follows. In section 2 we present our module-intersection method for trimming IBP systems and describe a simple way of obtaining the individual modules needed for our method. In section 3, the central part of our paper, we present a highly efficient new method for computing these intersections. We then explain the technical details of performing row reduction for the trimmed IBP systems from module intersections in section 4. In section 5, we present our example, the module-intersection IBP reduction of hexagon-box integrals, in details. The conclusions and outlook are given in section 6.

2 Simplification of IBP systems by the module-intersection method

In this section we present the details on trimming IBP systems by requiring the absence of integrals with doubled propagators [10], and by furthermore applying unitarity cuts. To demonstrate our method clearly, we first explain how to efficiently impose the no-doubled-propagator condition without applying unitarity cuts. Then we show that the same approach can be applied on unitarity cuts. In both cases, the algebraic constraints for trimming IBP systems are reformulated as the problem of computing the intersection of two modules over a polynomial ring [52]. At the end of this section, the algorithm for computing each individual module is introduced, while the intersection algorithm, the heart of this method, will be introduced in section 3.

2.1 Module intersection method without unitarity cuts

The objects under consideration are the multi-loop Feynman integrals,

$$I(\nu_1, \dots, \nu_m) = \int \frac{d^D l_1}{i\pi^{D/2}} \cdots \frac{d^D l_L}{i\pi^{D/2}} \frac{1}{D_1^{\nu_1} \cdots D_m^{\nu_m}}. \quad (2.1)$$

Here $\nu_i \in \mathbb{Z}$, l_1, \dots, l_L denote the loop momenta, and p_1, \dots, p_E the *independent* external momenta. By the procedure of integrand reduction,² we can set $m = L(L+1)/2 + LE$. The D_i denote inverse propagators.

¹We note that this technique has also been applied in ref. [43], cf. section III A of this reference.

²By “integrand reduction” we refer to a procedure applied to the loop integrand in which the number of propagators and powers of irreducible numerators in each contributing diagram are potentially reduced by making use of algebraic simplifications, without the use of IBP identities or discrete symmetries. For a systematic approach we refer to refs. [25–27].

To study IBP reduction of integrals of the form (2.1), we focus on the family of Feynman integrals associated with a particular Feynman diagram with k propagators (where $k \leq m$) and all its daughter diagrams, obtained by pinching propagators. Without loss of generality, the inverse denominators of this Feynman diagram can be labeled as D_1, \dots, D_k . Therefore, the family of integrals is,

$$\mathcal{F} = \{I(\nu_1, \dots, \nu_m) \mid \nu_j \leq 0 \text{ if } j > k\}. \quad (2.2)$$

Traditionally, IBP reduction is carried out within such a family. However, since the majority of the Feynman integrals that contribute to a quantity in perturbative QFT are integrals without doubled propagators, it is natural to consider the subfamily

$$\mathcal{F}_{\text{ndp}} = \{I(\nu_1, \dots, \nu_m) \mid \nu_j \leq 1 \text{ if } j \leq k, \quad \nu_j \leq 0 \text{ if } j > k\} \quad (2.3)$$

and the IBP relations for integrals in this subfamily [10].

We find that it is convenient to use the Baikov representation [55] of Feynman integrals for their integration-by-parts (IBP) reduction for several reasons: 1) the integrand reduction is manifest in this representation, 2) it is easy to apply unitarity cuts, 3) most importantly, it is surprisingly simple to trim IBP systems *analytically* in this representation. Here we briefly review the Baikov representation.

We collect the external and internal momenta as,

$$V = (v_1, \dots, v_{E+L}) = (p_1, \dots, p_E, \ell_1, \dots, \ell_L). \quad (2.4)$$

The Gram matrix S of these vectors is,

$$S = \left(\begin{array}{ccc|ccc} x_{1,1} & \cdots & x_{1,E} & x_{1,E+1} & \cdots & x_{1,E+L} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{E,1} & \cdots & x_{E,E} & x_{E,E+1} & \cdots & x_{E,E+L} \\ \hline x_{E+1,1} & \cdots & x_{E+1,E} & x_{E+1,E+1} & \cdots & x_{E+1,E+L} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{E+L,1} & \cdots & x_{E+L,E} & x_{E+L,E+1} & \cdots & x_{E+L,E+L} \end{array} \right), \quad (2.5)$$

where the elements are defined as $x_{i,j} = v_i \cdot v_j$. The upper-left $E \times E$ block is the Gram matrix of the external momenta, which is denoted as G . Defining $z_i \equiv D_i$ and integrating out solid-angle directions, the Feynman integral (2.1) takes the following form in Baikov representation,

$$I(\nu_1, \dots, \nu_m) = C_E^L U^{\frac{E-D+1}{2}} \int dz_1 \cdots dz_m P^{\frac{D-L-E-1}{2}} \frac{1}{z_1^{\nu_1} \cdots z_m^{\nu_m}}, \quad (2.6)$$

where $P \equiv \det S$, $U \equiv \det G$, and the factor C_E^L originates from the solid-angle integration and the Jacobian for the transformation $x_{i,j} \rightarrow z$. For the derivation of IBP identities, U and C_E^L are irrelevant, so we may ignore them in the following discussion. Note that in this representation, the inverse denominators D_i become free variables, and so the integrand reduction can be done automatically.

An IBP identity in this representation reads,

$$\begin{aligned}
 0 &= \int dz_1 \cdots dz_m \sum_{i=1}^m \frac{\partial}{\partial z_i} \left(a_i P^{\frac{D-L-E-1}{2}} \frac{1}{z_1^{\nu_1} \cdots z_m^{\nu_m}} \right) \\
 &= \int dz_1 \cdots dz_m \sum_{i=1}^m \left(\frac{\partial a_i}{\partial z_i} + \frac{D-L-E-1}{2P} a_i \frac{\partial P}{\partial z_i} - \frac{\nu_i a_i}{z_i} \right) P^{\frac{D-L-E-1}{2}} \frac{1}{z_1^{\nu_1} \cdots z_m^{\nu_m}}. \quad (2.7)
 \end{aligned}$$

Here the a_i denote polynomials in the ring $\mathbb{A} = \mathbb{Q}(\mathbf{s})[z_1, \dots, z_m]$. (We use \mathbf{s} to represent the independent Mandelstam variables and mass parameters collectively.) We remark that in the Baikov representation, the Baikov polynomial P vanishes on the boundary of the integration domain, and hence there is no surface term in the IBP identity. Note that the terms with the pole $1/P$ appearing inside the parenthesis in eq. (2.7) correspond to dimension-shifted integrals, and as such are not favorable in deriving simple IBP relations. To avoid such poles, we may impose the following constraints on the a_i [12, 13, 56],

$$\left(\sum_{i=1}^m a_i \frac{\partial P}{\partial z_i} \right) + bP = 0, \quad (2.8)$$

where b is also required to be a polynomial in \mathbb{A} . This constraint is known in computational commutative algebra as a “syzygy” equation [10]. In the following discussion, we only focus on the polynomials a_i , since once they are known it is straightforward to recover the polynomial b . The solutions of eq. (2.8), taking the form,

$$(a_1, \dots, a_m) \quad (2.9)$$

form a sub-module of the polynomial module \mathbb{A}^m , which we denote M_1 in the following.

Furthermore, to trim the IBP system, it is possible to work with integrals in \mathcal{F}_{ndp} , i.e. integrals without doubled propagators [10]. Note from the second line of eq. (2.7) that, even if the integral inside the differential operator has $\nu_i \leq 1$, the derivative will produce integrals with doubled propagators. This can also be prevented by a suitable choice of the a_i . For example, if we consider the integral of the parent diagram, $\nu_1 = \dots = \nu_k = 1$, $\nu_{k+1} \leq 0, \dots, \nu_m \leq 0$ inside eq. (2.7), we can avoid doubled propagators by requiring that a_i is divisible by z_i ,

$$a_i = b_i z_i, \quad i = 1, \dots, k. \quad (2.10)$$

Such (a_1, \dots, a_m) also form a sub-module of \mathbb{A}^m , which we denote M_2 . We need to solve eqs. (2.8) and (2.10) simultaneously to find IBP relations which involve neither dimension shifts nor doubled propagators in Baikov representation.

The strategy in ref. [13] to solve these conditions is to replace a_i by b_i in eq. (2.8) and then solve for $(b_1, \dots, b_k, a_{k+1}, \dots, a_m, b)$ as a single syzygy equation, employing Schreyer’s theorem. However, this approach, although it works well for simple two-loop four-point integrals, becomes less practical for more complicated kinematics. It was suggested in the reference [52] that a better strategy is to determine the generators of M_1 and M_2 individually, and then calculate the *module intersection*,

$$M_1 \cap M_2, \quad (2.11)$$

whose generators are solutions of eqs. (2.8) and (2.10). Geometrically, elements in $M_1 \cap M_2$ are the polynomial tangent vectors of the reducible hypersurface [52, 57]

$$z_1 \dots z_k P = 0. \tag{2.12}$$

In subsection 2.3, we will see that it takes no effort to find the generators of M_1 and M_2 . In section 3 we present a highly efficient algorithm for computing the intersection $M_1 \cap M_2$.

Once eqs. (2.8) and (2.10) are solved, we obtain the simplified IBP identities without doubled propagators. They take the following form,

$$0 = \int dz_1 \dots dz_m \left(\sum_{i=1}^m \frac{\partial a_i}{\partial z_i} - \sum_{i=1}^k b_i - \sum_{i=k+1}^m \frac{\nu_i a_i}{z_i} - \frac{D-L-E-1}{2} b \right) \frac{P^{\frac{D-L-E-1}{2}}}{z_1 \dots z_k z_{k+1}^{\nu_{k+1}} \dots z_m^{\nu_m}}. \tag{2.13}$$

Once the generators of $M_1 \cap M_2$ are obtained, we multiply them by monomials in the z_i in order to get a spanning set of IBP identities for the reduction targets. Alternatively, it is possible to apply D-module theory to get the IBP reductions directly from the generators of $M_1 \cap M_2$. We leave this direction for future research.

2.2 Module intersection method with cuts

In practice, for Feynman diagrams with high multiplicity or high loop order, instead of working with the Feynman integrals directly, it is more convenient to apply unitarity cuts [12, 13]. In this section we show how to apply the module intersection method in combination with unitarity cuts, thus simplifying the construction and subsequent Gaussian elimination of the IBP identities. We follow the notation of ref. [58].

Consider the c -fold cut of eq. (5.2) with $c \leq k$. Let \mathcal{S}_{cut} , $\mathcal{S}_{\text{uncut}}$ and \mathcal{S}_{ISP} denote the sets of indices of cut propagators, uncut propagators and irreducible scalar products (ISP) respectively. Explicitly,

$$\mathcal{S}_{\text{cut}} = \{\zeta_1, \dots, \zeta_c\}, \quad \mathcal{S}_{\text{uncut}} = \{r_1, \dots, r_{k-c}\}, \quad \mathcal{S}_{\text{ISP}} = \{r_{k-c+1}, \dots, r_{m-c}\}. \tag{2.14}$$

In Baikov representation, the c -fold cut of integrals with $\nu_1 = \dots = \nu_k = 1$ takes a simple form,

$$I(\nu_1, \dots, \nu_m) \propto \int dz_{r_1} \dots dz_{r_{m-c}} \frac{\tilde{P}^{\frac{D-L-E-1}{2}}}{z_{r_1} \dots z_{r_{k-c}} z_{r_{k-c+1}}^{\nu_{r_{k-c+1}}} \dots z_{r_{m-c}}^{\nu_{r_{m-c}}}}, \tag{2.15}$$

where,

$$\tilde{P} = P|_{z_{\zeta_1} \rightarrow 0, \dots, z_{\zeta_c} \rightarrow 0}. \tag{2.16}$$

We can derive IBP identities for the integrals on the cut \mathcal{S}_{cut} ,

$$\begin{aligned}
0 &= \int dz_{r_1} \dots dz_{r_{m-c}} \sum_{i=1}^{m-c} \frac{\partial}{\partial z_{r_i}} \left(\tilde{a}_{r_i} \frac{\tilde{P}^{\frac{D-L-E-1}{2}}}{z_{r_1} \dots z_{r_{k-c}} z_{r_{k-c+1}}^{\nu_{r_{k-c+1}}} \dots z_{r_{m-c}}^{\nu_{r_{m-c}}}} \right) \\
&= \int dz_{r_1} \dots dz_{r_{m-c}} \left(\sum_{i=1}^{m-c} \frac{\partial \tilde{a}_{r_i}}{\partial z_{r_i}} + \frac{D-L-E-1}{2\tilde{P}} \sum_{i=1}^{m-c} \tilde{a}_{r_i} \frac{\partial \tilde{P}}{\partial z_{r_i}} - \sum_{i=1}^{k-c} \frac{\tilde{a}_{r_i}}{z_{r_i}} - \sum_{i=k-c+1}^{m-c} \frac{\nu_{r_i} \tilde{a}_{r_i}}{z_{r_i}} \right) \\
&\quad \times \frac{\tilde{P}^{\frac{D-L-E-1}{2}}}{z_{r_1} \dots z_{r_{k-c}} z_{r_{k-c+1}}^{\nu_{r_{k-c+1}}} \dots z_{r_{m-c}}^{\nu_{r_{m-c}}}}, \tag{2.17}
\end{aligned}$$

where \tilde{a}_{r_i} , $i = 1, \dots, m-c$ are polynomials in $\tilde{\mathbb{A}} = \mathbb{Q}(\mathbf{s})[z_{r_1}, \dots, z_{r_{m-c}}]$. Once again, to derive simple IBP relations without dimension shifts or doubled propagators [12, 13], we may impose the conditions,

$$\left(\sum_{i=1}^{m-c} \tilde{a}_{r_i} \frac{\partial \tilde{P}}{\partial z_{r_i}} \right) + \tilde{b}\tilde{P} = 0, \tag{2.18}$$

$$\tilde{a}_{r_i} = \tilde{b}_{r_i} z_{r_i}, \quad i = 1, \dots, k-c. \tag{2.19}$$

Again, the module intersection method provides an efficient tool to solve these constraints simultaneously. However, for the purpose of automation, it is useful to make use of the following slight reformulation. Define

$$\tilde{a}_{\zeta_i} \equiv 0, \quad i = 1, \dots, c. \tag{2.20}$$

Then eqs. (2.18) and (2.19) are formally recast as,

$$\left(\sum_{i=1}^m \tilde{a}_i \frac{\partial \tilde{P}}{\partial z_i} \right) + \tilde{b}\tilde{P} = 0, \tag{2.21}$$

$$\tilde{a}_{r_i} = \tilde{b}_{r_i} z_{r_i}, \quad i = 1, \dots, k-c. \tag{2.22}$$

By comparing these equations and their counterparts without applied cuts, we observe the following shortcut. Recall that the modules M_1 and M_2 defined in the previous subsection are the solution sets for eqs. (2.8) and (2.10) respectively. We define

$$\tilde{M}_1 = M_1|_{z_{\zeta_1} \rightarrow 0, \dots, z_{\zeta_c} \rightarrow 0}, \quad \tilde{M}_2 = M_2|_{z_{\zeta_1} \rightarrow 0, \dots, z_{\zeta_c} \rightarrow 0}, \tag{2.23}$$

whereby the c -fold cut on the elements of M_1 and M_2 has been imposed. Then clearly,

$$\tilde{M}_1 \cap \tilde{M}_2 \tag{2.24}$$

solve the equations (2.21) and (2.22) simultaneously. Note that any element in $(q_1, \dots, q_m) \in M_2$ has the property $q_{\zeta_i} = h_{\zeta_i} z_{\zeta_i}$, $i = 1, \dots, c$. After imposing the cut we have,

$$q_{\zeta_i}|_{z_{\zeta_1} \rightarrow 0, \dots, z_{\zeta_c} \rightarrow 0} = 0, \tag{2.25}$$

which is consistent with the requirement (2.20). This automates the computation: to obtain \tilde{M}_1 and \tilde{M}_2 , we first compute M_1 and M_2 , and then simply apply the rules for the various cuts. An algorithm for computation of the intersection will be introduced in section 3.

Once the constraints (2.21) and (2.22) are solved, the IBP identities on the cut take the simple form,

$$0 = \int dz_{r_1} \cdots dz_{r_{m-c}} \left(\sum_{i=1}^{m-c} \frac{\partial \tilde{a}_{r_i}}{\partial z_{r_i}} - \frac{D-L-E-1}{2} \tilde{b} - \sum_{i=1}^{k-c} \tilde{b}_{r_i} - \sum_{i=k-c+1}^{m-c} \frac{\nu_{r_i} \tilde{a}_{r_i}}{z_{r_i}} \right) \times \frac{\tilde{P}^{\frac{D-L-E-1}{2}}}{z_{r_1} \cdots z_{r_{k-c}} z_{r_{k-c+1}}^{\nu_{r_{k-c+1}}} \cdots z_{r_{m-c}}^{\nu_{r_{m-c}}}}. \quad (2.26)$$

As for the uncut case, once the generators of $\tilde{M}_1 \cap \tilde{M}_2$ are obtained, we multiply them by monomials in the z_{r_i} , $i = 1, \dots, m - c$ to get a spanning set of IBP identities.

The cuts necessary for reconstructing the complete IBP identities can be determined from a list of master integrals [13, 18]: they are the maximal cuts of “uncollapsible” master integrals (master integrals which cannot be obtained from other integrals in the basis by adding propagators). In practice, the list of master integrals can be quickly determined by the packages MINT [59], AZURITE [18]. The total number of master integrals can also be determined by the D-module theory method of ref. [19].

Upon applying Gaussian elimination to the IBP identities evaluated on each cut, we can subsequently merge the obtained reductions to find the complete IBP reductions without applied cuts.

2.3 Algorithm for computing individual modules

The modules M_1 and M_2 defined in subsection 2.1, and hence \tilde{M}_1 and \tilde{M}_2 defined in the previous subsection, can all be determined without effort.

The condition (2.8) for M_1 is a syzygy equation for the Baikov polynomial P and its derivatives. Schreyer’s theorem [60] guarantees that solutions for syzygy equations can be obtained from Gröbner basis computations. However, for the Baikov polynomial P this is not necessary, owing to the special structure of P . A convenient way to find the solution, or equivalently the tangent vectors for the hypersurface $P = 0$, is to use the basic canonical IBP identities [12]. Here alternatively, we use the Laplace expansion method [58]³ of the Gram determinant S (2.5) to determine M_1 , since the results are manifestly expressed in the z variables.

Laplace expansion of $P = \det S$ yields,

$$\left(\sum_{k=1}^{E+L} (1+\delta_{i,k}) x_{j,k} \frac{\partial P}{\partial x_{i,k}} \right) - 2\delta_{i,j} P = 0, \quad (2.27)$$

where $E+1 \leq i \leq E+L$ and $1 \leq j \leq E+L$. These $L(E+L)$ relations are syzygy relations between P and its derivatives in the $x_{i,j}$ variables. It is straightforward to convert them to solutions of eq. (2.8),

$$\sum_{\alpha=1}^m (a_{i,j})_{\alpha} \frac{\partial P}{\partial z_{\alpha}} + b_{i,j} P = 0, \quad (2.28)$$

³We thank Roman Lee for introducing us to the Laplace expansion relations of symmetric matrices, also explained at his website <http://mathsketches.blogspot.ru/2010/07/blog-post.html> (in Russian).

where $a_{i,j}$ and b are obtained by applying the chain rule to the expressions in eq. (2.27). Explicitly, they are given by,

$$(a_{i,j})_\alpha = \sum_{k=1}^{E+L} (1 + \delta_{i,k}) \frac{\partial z_\alpha}{\partial x_{i,k}} x_{j,k} \quad \text{and} \quad b_{i,j} = -2\delta_{i,j}. \quad (2.29)$$

It is proven in ref. [58] via Józefiak complexes [61] that the $L(E+L)$ tuples of polynomials $(a_{i,j})_\alpha$ form a complete generating set of M_1 .

We remark that,

1. The generating set (2.29) is at most linear in the z_i .
2. The generating set (2.29) is homogenous in the z_i and the Mandelstam variables/mass parameters, as can be inferred from dimensional analysis.

The second property is crucial for our highly efficient algorithm for computing intersections of modules, to be explained in section 3.

The generating set for M_2 is trivial. There are m generators,

$$z_1 \mathbf{e}_1, \quad \dots, \quad z_k \mathbf{e}_k, \quad \mathbf{e}_{k+1}, \quad \dots, \quad \mathbf{e}_m. \quad (2.30)$$

Here \mathbf{e}_j is the j -th m -dimensional unit vector.

The cut cases, \tilde{M}_1 and \tilde{M}_2 defined in eq. (2.23), can then be obtained from eqs. (2.29) and (2.30) by simply setting $z_{\zeta_i} \rightarrow 0$, $i = 1, \dots, c$.

3 Determining the module intersection

In this section, we introduce a highly efficient algorithm for computing generators of the module intersections and an algorithm to trim the generating sets.

Given the two submodules $M_1, M_2 \subset A^t$ over the multivariate polynomial ring $A = F[z_1, \dots, z_n]$ with coefficients in the multivariate rational function field $F = \mathbb{Q}(c_1, \dots, c_r)$, our goal is to obtain a generating system of the module $M_1 \cap M_2$ (which is finitely generated since A is Noetherian). We apply Gröbner basis techniques to address this problem. Our algorithms are implemented in the computer algebra system SINGULAR, which focuses on polynomial computations with applications in commutative algebra and algebraic geometry [54].

We first recall some terminology: using the notation $z^\alpha = z_1^{\alpha_1} \dots z_n^{\alpha_n}$ for the monomials in A , we call $z^\alpha e_i$ with a unit basis vector $e_i \in A^t$ a *monomial* of A^t . An F -multiple of a monomial is called a *term*. A *monomial ordering* on A^t is a total ordering $>$ on the set of monomials in A^t , which respects multiplication, that is, $z^\alpha e_i > z^\beta e_j$ implies $z^\alpha z^\gamma e_i > z^\beta z^\gamma e_j$ for all $\alpha, \beta, \gamma, i, j$, and $z^\alpha e_i > z^\beta e_i \Leftrightarrow z^\alpha e_j > z^\beta e_j$ for all α, β, i, j . Then $>$ induces a monomial ordering on the monomials of A , which we again denote by $>$. In turn, any monomial ordering $>$ on A induces two canonical monomial orderings on A^t , *position over term*

$$z^\alpha e_i > z^\beta e_j \quad :\Leftrightarrow \quad i < j \text{ or } (i = j \text{ and } z^\alpha > z^\beta) \quad (3.1)$$

and analogously *term over position*. We call a monomial ordering on A^t *global* if the induced ordering on A is global, that is, $z^\alpha > 1$ for all α . Any $0 \neq f \in A^t$ can be written as $f = c \cdot z^\alpha e_i + g$ with $z^\alpha e_i > z^\beta e_j$ for all terms $\tilde{c} \cdot z^\beta e_j$ of $g \in A^t$. Then the term $LT_{>}(f) = c \cdot z^\alpha e_i$ is called the *lead term* of f , the constant $LC_{>}(f) = c$ is called the *lead coefficient* of f , and $L_{>}(f) = z^\alpha e_i$ the *lead monomial* of f . The monomials of A^t come with a natural partial order, which we call *divisibility*

$$z^\alpha e_i \mid z^\beta e_j \iff i = j \text{ and } z^\alpha \mid z^\beta. \tag{3.2}$$

For terms $c_1 z^\alpha e_i$ and $c_2 z^\beta e_j$ with $z^\alpha e_i \mid z^\beta e_j$ we define their quotient as $\frac{c_2 z^\beta e_j}{c_1 z^\alpha e_i} = \frac{c_2 z^\beta}{c_1 z^\alpha} \in A$. Moreover, we define the least common multiple $\text{lcm}(z^\alpha e_i, z^\beta e_j)$ as zero if $i \neq j$, and as $\text{lcm}(z^\alpha, z^\beta)$ otherwise. For a subset $G \subset A^t$, the *leading module* $L(G)$ is the module of all A -linear combinations of the lead monomials of the non-zero elements of G .

By iteratively canceling the lead term of f via multiples of lead terms of the divisors in $G = \{g_1, \dots, g_l\} \subset A^t$ with respect to a fixed global ordering, we obtain a notion of *division with remainder* yielding a division expression

$$f = \sum_i a_i g_i + \text{NF}_{>}(f, G) \tag{3.3}$$

with $a_i \in A$ such that $\text{NF}(0, G) = 0$, $\text{NF}_{>}(f, G) \neq 0$ implies that $L_{>}(\text{NF}_{>}(f, G)) \notin L(G)$, and the lead monomial of f is not smaller than that of any $a_i g_i$.

Let $U \subset A^t$ be a submodule and $>$ a global monomial ordering. A finite set $0 \notin G = \{g_1, \dots, g_l\} \subset U$ is called *Gröbner basis* of U with respect to $>$, if

$$L_{>}(G) = L_{>}(U). \tag{3.4}$$

Theorem 1 (Buchberger). *With notation as above, the following conditions are equivalent:*

1. G is a Gröbner basis of U ,
2. $f \in U \iff \text{NF}_{>}(f, G) = 0$,
3. $U = \langle G \rangle$ and $\text{NF}_{>}(\text{spoly}_{>}(g_i, g_j), G) = 0$ for all $i \neq j$, where

$$\text{spoly}(f, g) := \frac{\text{lcm}(L(f), L(g))}{LT(f)} f - \frac{\text{lcm}(L(f), L(g))}{LT(g)} g. \tag{3.5}$$

is the so-called *S-polynomial* (or *syzygy polynomial*) of f and g .

For a proof of this standard fact, see for example section 2.3 of ref. [62]. A generating set G of U can be extended to a Gröbner basis by means of Buchberger's algorithm, which according to the above criterion computes the remainder $r = \text{NF}_{>}(\text{spoly}_{>}(g_i, g_j), G)$ for all g_i, g_j in G , adds r to G if $r \neq 0$, and iterates this process with the updated G until all remainders vanish. This process terminates, since A^t is Noetherian and, hence, any ascending chain of submodules becomes stationary (section 2.1 of ref. [62]). Along this process, we can determine all relations between the g_i :

Algorithm 2 (Syzygies). Let $M_G = (g_1, \dots, g_l) \in A^{t \times l}$. If K is a Gröbner basis of the column space of

$$\begin{pmatrix} M_G \\ 1 & 0 \\ & \ddots \\ 0 & 1 \end{pmatrix} \tag{3.6}$$

with regard to the position over term order, k_1, \dots, k_m are the elements of K in $\bigoplus_{i=t+1}^{t+l} e_i$, and $\pi : A^{t+l} \rightarrow A^l$ is the projection onto the last l coordinates, then

$$\text{syz}(g_1, \dots, g_l) := \ker M_G \tag{3.7}$$

is generated as an A -module by $\pi(k_1), \dots, \pi(k_m)$.

So $\text{syz}(g_1, \dots, g_l) = \text{im}(\pi(k_1), \dots, \pi(k_m))$ is the image of the matrix with the $\pi(k_i)$ in the columns, in particular, $M_G \cdot (\pi(k_1), \dots, \pi(k_m)) = 0$. For a proof of correctness of the algorithm, see for example lemma 2.5.3 of ref. [62]. We can use this algorithm to compute module intersections:

Lemma 3 (Intersection). Let $M_1 = \langle v_1, \dots, v_l \rangle$ and $M_2 = \langle w_1, \dots, w_p \rangle$ be submodules of A^t , and

$$\text{syz}(v_1, \dots, v_l, w_1, \dots, w_p) = \text{im} \begin{pmatrix} G \\ H \end{pmatrix} \tag{3.8}$$

with $G = (g_{i,j}) \in A^{l \times m}$ and $H = (h_{i,j}) \in A^{p \times m}$ as obtained from Algorithm 2. Then the columns of

$$(v_1, \dots, v_l) \cdot G, \tag{3.9}$$

that is, the vectors $\sum_{i=1}^l g_{i,j} v_i$ with $j = 1, \dots, m$ generate $M_1 \cap M_2$.

Proof. Any element

$$s = \begin{pmatrix} s_1 \\ s_2 \end{pmatrix} \in \text{syz}(v_1, \dots, v_l, w_1, \dots, w_p) \tag{3.10}$$

with $s_1 = (a_j) \in A^l$ and $s_2 = (b_j) \in A^p$ yields an element

$$M_1 \ni \sum_{j=1}^l a_j v_j = -\sum_{j=1}^p b_j w_j \in M_2 \tag{3.11}$$

in $M_1 \cap M_2$. On the other hand, if $m \in M_1 \cap M_2$, then there are $a_j, b_j \in A$ with $m = \sum_{j=1}^l a_j v_j = -\sum_{j=1}^p b_j w_j$. Then the vertical concatenation of $s_1 = (a_j) \in A^l$ and $s_2 = (b_j) \in A^p$ is in $\text{syz}(v_1, \dots, v_l, w_1, \dots, w_p)$. \square

Algorithm 2 in conjunction with Lemma 3 can be used to determine a generating system of $M_1 \cap M_2$. For the module intersection problems arising from the non-planar hexagon-box diagram however, the performance of Buchberger's algorithm over F is not sufficient to yield a generating system in a reasonable time-frame. It turns out that a classical technique (which to our knowledge dates back to ref. [53]) to simulate computations over rational function fields via polynomial computations is much faster. We apply this technique to the Gröbner basis computation yielding the syzygy matrix used to determine the module intersection.

Definition 4. Given monomial orderings $>_1$ and $>_2$ on the monomials in z_1, \dots, z_n and c_1, \dots, c_r , respectively, a monomial ordering $>$ is given by

$$z^\alpha c^\beta > z^{\alpha'} c^{\beta'} \iff z^\alpha >_1 z^{\alpha'} \text{ or } (z^\alpha = z^{\alpha'} \text{ and } c^\beta >_2 c^{\beta'}) \quad (3.12)$$

We call $>$ the block ordering $(>_1, >_2)$ associated to $>_1$ and $>_2$.

Lemma 5 (Localization). Let $A = \mathbb{Q}(c_1, \dots, c_r)[z_1, \dots, z_n]$, let $B = \mathbb{Q}[z_1, \dots, z_n, c_1, \dots, c_r]$, let v_1, \dots, v_l be vectors with entries in B , and define

$$\begin{aligned} U &= \langle v_1, \dots, v_l \rangle \subset A^t \\ U' &= \langle v_1, \dots, v_l \rangle \subset B^t \end{aligned}$$

Let $G \subset B^t$ be a Gröbner basis of U' with respect to a global block ordering $(>_1, >_2)$ with blocks $z_1, \dots, z_n > c_1, \dots, c_r$. Then G is also a Gröbner basis of U with respect to $>_1$.

Proof. Denote the block ordering $(>_1, >_2)$ by $>$. Every $f \in U$ can be written as

$$f = \frac{1}{h} \sum_i \alpha_i v_i \quad (3.13)$$

with $\sum_i \alpha_i v_i \in U'$ and $h \in \mathbb{Q}[c_1, \dots, c_r]$. By $h \cdot f \in U'$ and G being a Gröbner basis of U' , we know that $\text{NF}(h \cdot f, G) = 0$. Hence, there is a $g \in G$ with $L_{>}(g) \mid L_{>}(h \cdot f)$ and $L_{>}(h \cdot f) = L_{>}(h) \cdot L_{>}(f)$. Since $L_{>}(h)$ is a unit (invertible) in A and $>$ is a block ordering, we obtain that $L_{>_1}(g) \mid L_{>_1}(f)$. This argument shows that $L_{>_1}(U) = L_{>_1}(G)$, that is, G is a Gröbner basis of U with respect to $>_1$. \square

Remark 6. The method of Lemma 5 turns out to be efficient since the input modules in our setting are homogeneous in the variables $z_1, \dots, z_n, c_1, \dots, c_r$, which allows for efficient sorting of the S -polynomials in Buchberger's algorithm by degree.

Remark 7. For our setting, modular techniques, which compute over finite fields, combine the results using the Chinese remainder theorem and apply rational reconstruction (as developed in a general setting in refs. [63] and [64]) seem not to be useful since very large constant coefficients occur. As a result, this approach would require considering a large number of primes for lifting.

The module intersection algorithm resulting from Lemma 3 and Lemma 5 produces generating sets which usually are not minimal in any sense. For a homogeneous module, a minimal generating system can be determined, however, the computation is very expensive. Another option is to determine the unique reduced Gröbner basis. A Gröbner basis g_1, \dots, g_l is called *minimal* if $L(g_i)$ does not divide $L(g_j)$ for all $i \neq j$. Such a minimal Gröbner basis is called *reduced* if none of the terms of the tails $g_i - \text{LT}(g_i)$ is divisible by some $L(g_j)$. It does, however, also not make much sense to pass to a minimal or reduced Gröbner basis, since Gröbner bases can be much larger than generating systems and usually cannot be obtained in a reasonable time in our setting. We hence employ a randomized algorithm for trimming the generating systems to remove extraneous generators. This algorithm is based on determining reduced Gröbner bases after passing to a finite field and specific values of parameters c_i :

Algorithm 8 (Trimming). *Given a generating system g_1, \dots, g_l of a submodule $U \subset A^t$ with polynomial coefficients in $\mathbb{Z}[c_1, \dots, c_r]$, we proceed as follows:*

1. *Substitute the c_i in the g_j by pairwise different large prime numbers p_i obtaining polynomials $h_j \in \mathbb{Z}[z_1, \dots, z_n]$.*
2. *Choose a large prime p different to the primes p_i . Apply the canonical map $\mathbb{Z}[z_1, \dots, z_n] \rightarrow \mathbb{F}_p[z_1, \dots, z_n]$ to the h_j obtaining $\overline{h}_1, \dots, \overline{h}_l$.*
3. *Choose an integer $j_0 \in \{1, \dots, l\}$.*
4. *Compute the reduced Gröbner bases G_1 of*

$$\langle \overline{h}_j \mid j = 1, \dots, l \rangle \tag{3.14}$$

and G_2 of

$$\langle \overline{h}_j \mid j = 1, \dots, l \text{ with } j \neq j_0 \rangle. \tag{3.15}$$

5. *If $G_1 = G_2$ return $\{g_j \mid j = 1, \dots, l \text{ and } j \neq j_0\}$.*

Multiple runs of the algorithm with different p and p_i reduce the chance of a bad prime or a bad parameter value. We apply Algorithm 8 iteratively to drop generators, starting with generators of large (byte) size.

4 Sparse row reduction

In this section we turn to discussing linear algebra techniques. Although the generation of IBP identities using eq. (2.26) is very fast, achieving the reduction of target integrals to linear combinations of master integrals is highly non-trivial. This is because the latter step requires computing the row reduced echelon form (RREF) of the IBP identities, which becomes computationally intensive in cases with multiple external invariants (i.e., Mandelstam variables and mass parameters), as these enter the IBP system as parameters. Therefore, analytic computation of the RREF requires sophisticated linear algebra techniques.

4.1 Selection of relevant and independent IBP identities

The IBP identities generated from eq. (2.26) usually contain linearly redundant identities, as well as identities which are irrelevant for reducing the target integrals. To speed up linear reduction in the subsequent step, we make use of the following methods to remove redundant and irrelevant identities.

1. *Removal of redundant linear identities.* This can be done with the standard linear algebra algorithm of picking up independent rows of a matrix. We construct the matrix of all requisite IBP identities, sort the rows by their density (i.e., number of non-vanishing entries) or their byte count, and then compute the RREF of the transposed matrix numerically. The pivot locations correspond to the linearly independent IBP identities, giving preference to sparser IBP identities, or IBP identities of smaller sizes.

2. *Removal of irrelevant linear identities.* Furthermore, given a target integral set, we can single out the relevant IBP identities which will ultimately reduce them to master integrals. This can also be done with a numeric RREF. We carry out the reduction numerically and record the rows used for reducing the targets in the computation (by recording the left-multiplying matrix of the row reduced matrix).

Regarding the numeric RREF above, in practice we assign generic integer values to all external invariants (Mandelstam variables and mass parameters) and the spacetime dimension and work with finite fields. The computation is powered by the highly efficient sparse finite-field linear algebra package SPASM [65].

4.2 Sparse REF and RREF strategies

We find that the IBP system that arises from eq. (2.26), after removing the linearly dependent and irrelevant IBP identities for targets is, in general, very sparse. To find the row echelon form (REF) and RREF efficiently, it is crucial to apply a sophisticated pivoting strategy to keep the linear system sparse in the intermediate steps.

First, we write the IBP identities in the form of a matrix, with the columns sorted according to some integral ordering, for example like that of AZURITE. Then the RREF will eventually reduce the target integrals to the AZURITE master integral basis. However, it is important to swap rows and columns during the REF computation, and find suitable pivots for row reduction, in order to keep the matrix sparse. This can be achieved by the heuristic Markowitz strategy [66], provided that all entries in the sparse matrix are of a similar size. However, in our cases, the entries are polynomials or rational functions in Mandelstam variables and mass parameters, and so a weighted pivot strategy, considering both the sparsity and the byte sizes, is used.

Note that we use a *total pivoting* strategy for which both row swaps and column swaps are used. The row swaps will not change the final result of the RREF, whereas the column swap *will* change the final result of the RREF. This means that the target integrals will typically not be reduced to the desired master integral basis, but a different integral basis. If we require the target integrals to be reduced to a specific pre-determined basis (say, the AZURITE basis), a basis change must be carried out after RREF computation. We find that it is more efficient to allow both row and column swaps, and then compute the basis change, than to allow row swap only (partial pivoting strategy).

The REF and RREF algorithm is implemented in our primitive MATHEMATICA code. A more efficient implementation in SINGULAR is in preparation and will become available soon.

5 The non-planar hexagon-box diagram example

To demonstrate the power of our new method, we consider a cutting-edge integration-by-parts reduction problem: the reduction of two-loop five-point non-planar massless hexagon-box integrals. Recently, this diagram has attracted a great deal of interest, and the hexagon-box integral with a chiral numerator has been analytically computed by use of the bootstrap method and by superconformal Ward identities [67, 68]. Here we consider the analytic IBP reduction of hexagon-box integrals with arbitrary numerators with the degree up to four.

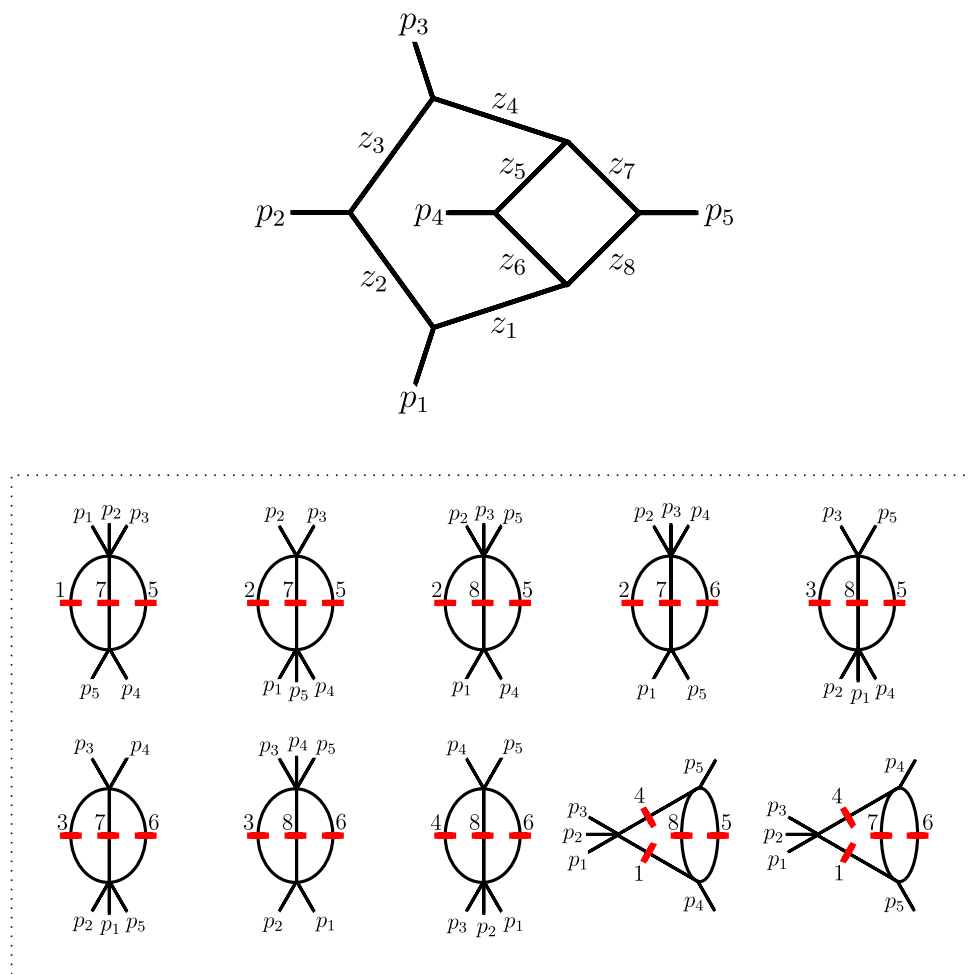


Figure 1. The fully massless non-planar hexagon-box diagram, along with our labelling conventions for its internal lines. The lower part shows the subset of the basis integrals with the property that their graphs cannot be obtained by adding internal lines to the graph of another basis integral. The corresponding cuts $\{1, 5, 7\}$, $\{2, 5, 7\}$, $\{2, 5, 8\}$, $\{2, 6, 7\}$, $\{3, 5, 8\}$, $\{3, 6, 7\}$, $\{3, 6, 8\}$, $\{4, 6, 8\}$, $\{1, 4, 5, 8\}$ and $\{1, 4, 6, 7\}$ are the cuts required for deriving complete IBP identities for the non-planar hexagon-box diagram.

The hexagon-box diagram, and the necessary cuts for deriving the IBP identities, are illustrated below in figure 1.

We define the inverse propagators as follows, setting $P_{i\dots j} \equiv p_i + \dots + p_j$,

$$\begin{aligned}
 3D_1 &= \ell_1^2, & D_2 &= (\ell_1 - p_1)^2, & D_3 &= (\ell_1 - P_{12})^2, \\
 D_4 &= (\ell_1 - P_{123})^2, & D_5 &= (\ell_1 + \ell_2 + p_4)^2, & D_6 &= (\ell_1 + \ell_2)^2, \\
 D_7 &= (\ell_2 - p_5)^2, & D_8 &= \ell_2^2, & D_9 &= (\ell_1 + p_5)^2, \\
 D_{10} &= (\ell_2 + p_1)^2, & D_{11} &= (\ell_2 + p_2)^2, & &
 \end{aligned}
 \tag{5.1}$$

and consider the family of Feynman integrals,

$$I(\nu_1, \dots, \nu_{11}) = \int \frac{d^D l_1}{i\pi^{D/2}} \frac{d^D l_2}{i\pi^{D/2}} \frac{1}{D_1^{\nu_1} \dots D_{11}^{\nu_{11}}}, \tag{5.2}$$

with $\nu_9 \leq 0$, $\nu_{10} \leq 0$ and $\nu_{11} \leq 0$. In terms of the notation of subsection 2.1, we have $L = 2$, $E = 4$, $k = 8$ and $m = 11$. We furthermore define the Baikov variables $z_i \equiv D_i$, set $s_{ij} = (p_i + p_j)^2$ and express the IBP identities in terms of the Mandelstam variables $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$.

Using AZURITE [18] we establish that, without applying global symmetries, there are 75 “pre”-master integrals — i.e., master integrals where global symmetry relations have not yet been imposed. In the notation of eq. (5.2), they take the form,

$$\begin{aligned}
 & \{\mathcal{I}_1, \dots, \mathcal{I}_{75}\} \equiv \\
 & \{I(1, 1, 1, 1, 1, 1, 1, 1, -2, 0, 0), I(1, 1, 1, 1, 1, 1, 1, 1, -1, 0, 0), I(1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0), \\
 & I(1, 1, 1, 0, 1, 1, 1, 1, -1, 0, 0), I(1, 1, 1, -1, 1, 1, 1, 1, 0, 0, 0), I(1, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0), \\
 & I(1, 1, -1, 1, 1, 1, 1, 1, 0, 0, 0), I(1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0), I(1, -1, 1, 1, 1, 1, 1, 1, 0, 0, 0), \\
 & I(1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0), I(0, 1, 1, 1, 1, 1, 1, 1, -1, 0, 0), I(-1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0), \\
 & I(0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0), I(1, 1, 1, 1, 1, -1, 0, 1, 0, 0, 0), I(1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0), \\
 & I(1, 1, 1, 1, -1, 1, 1, 0, 0, 0, 0), I(1, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0), I(1, 1, 1, -1, 1, 1, 1, 0, 0, 0, 0), \\
 & I(1, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0), I(1, 1, 1, -1, 1, 0, 1, 1, 0, 0, 0), I(1, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0), \\
 & I(1, -1, 1, 0, 1, 1, 1, 1, 0, 0, 0), I(1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0), I(1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0), \\
 & I(-1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0), I(0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0), I(-1, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0), \\
 & I(0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 0), I(0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 0), I(-1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0), \\
 & I(0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0), I(1, 1, 1, 0, 1, 0, 1, 0, 0, 0, 0), I(1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0), \\
 & I(1, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0), I(1, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0), I(1, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0), \\
 & I(1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 0), I(1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0), I(1, 0, 1, 1, 1, 0, 0, 1, 0, 0, 0), \\
 & I(1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0), I(1, -1, 1, 0, 1, 1, 1, 0, 0, 0, 0), I(1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0), \\
 & I(1, -1, 1, 0, 1, 0, 1, 1, 0, 0, 0), I(1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0), I(0, 1, 1, 1, 1, 0, 0, 1, 0, 0, 0), \\
 & I(0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 0), I(0, 1, 1, 1, 0, 1, 0, 1, 0, 0, 0), I(0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0), \\
 & I(0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0), I(0, 1, 1, 0, 1, 0, 1, 1, 0, 0, 0), I(0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0), \\
 & I(-1, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0), I(0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 0), I(-1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0), \\
 & I(0, 1, 0, 1, 0, 1, 1, 1, 0, 0, 0), I(0, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0), I(0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0), \\
 & I(0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0), I(0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0), I(1, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0), \\
 & I(1, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0), I(1, 0, 1, 0, 0, 1, 1, 0, 0, 0, 0), I(1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0), \\
 & I(1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0), I(0, 1, 0, 1, 1, 0, 0, 1, 0, 0, 0), I(0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 0), \\
 & I(0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 0), I(1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0), I(0, 1, 0, 0, 1, 0, 1, 0, 0, 0, 0), \\
 & I(0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0), I(0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0), I(0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 0), \\
 & I(0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0), I(0, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0), I(0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0)\}.
 \end{aligned}
 \tag{5.3}$$

The graphs of these integrals are shown in figure 2. We remark that AZURITE chooses master integrals which contain no doubled propagators.

As explained at the end of section 2.2, the cuts that are necessary to construct the complete IBP identities are the maximal cuts of the “uncollapsible” master integrals in eq. (5.3). From this list of master integrals we find that the following 10 cuts are necessary for computing the complete IBP identities,

$$\begin{aligned}
 & \{1, 5, 7\}, \{2, 5, 7\}, \{2, 5, 8\}, \{2, 6, 7\}, \{3, 5, 8\}, \{3, 6, 7\}, \{3, 6, 8\}, \{4, 6, 8\}, \\
 & \{1, 4, 5, 8\}, \{1, 4, 6, 7\}.
 \end{aligned}
 \tag{5.4}$$

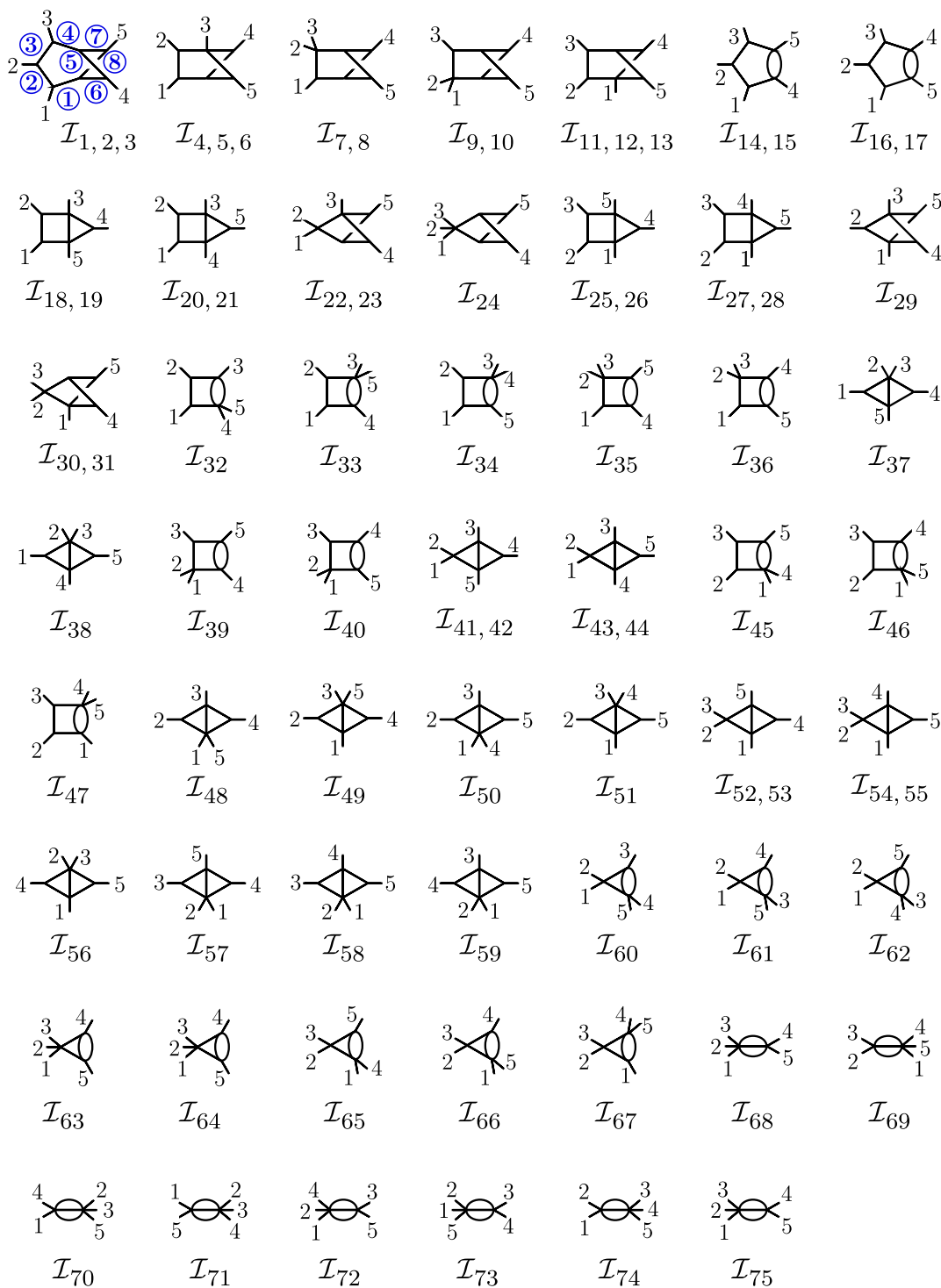


Figure 2. (Color online.) The 75 “pre”-master integrals found by AZURITE, with the global symmetry option turned off, for the non-planar hexagon-box family of eq. (5.2). By turning on the global symmetry option, AZURITE determines that there are 73 master integrals. These are the illustrated “pre”-master integrals excluding \mathcal{I}_{63} and \mathcal{I}_{68} . Our labeling convention for the propagators, corresponding to the indices recorded in eq. (5.3), are shown in the graph of the hexagon-box itself as the blue encircled numbers.

cut	“pre”-master integrals
{1, 5, 7}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{18}, \mathcal{I}_{19}, \mathcal{I}_{20}, \mathcal{I}_{21}, \mathcal{I}_{22}, \mathcal{I}_{23}, \mathcal{I}_{24}, \mathcal{I}_{32}, \mathcal{I}_{37}, \mathcal{I}_{38}, \mathcal{I}_{41}, \mathcal{I}_{42}, \mathcal{I}_{43}, \mathcal{I}_{44}, \mathcal{I}_{60}, \mathcal{I}_{68}$
{2, 5, 7}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{18}, \mathcal{I}_{19}, \mathcal{I}_{20}, \mathcal{I}_{21}, \mathcal{I}_{29}, \mathcal{I}_{30}, \mathcal{I}_{31}, \mathcal{I}_{32}, \mathcal{I}_{37}, \mathcal{I}_{38}, \mathcal{I}_{48}, \mathcal{I}_{50}, \mathcal{I}_{56}, \mathcal{I}_{69}$
{2, 5, 8}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{14}, \mathcal{I}_{15}, \mathcal{I}_{20}, \mathcal{I}_{21}, \mathcal{I}_{25}, \mathcal{I}_{26}, \mathcal{I}_{29}, \mathcal{I}_{30}, \mathcal{I}_{31}, \mathcal{I}_{33}, \mathcal{I}_{35}, \mathcal{I}_{38}, \mathcal{I}_{45}, \mathcal{I}_{49}, \mathcal{I}_{50}, \mathcal{I}_{52}, \mathcal{I}_{53}, \mathcal{I}_{56}, \mathcal{I}_{65}, \mathcal{I}_{70}$
{2, 6, 7}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{16}, \mathcal{I}_{17}, \mathcal{I}_{18}, \mathcal{I}_{19}, \mathcal{I}_{27}, \mathcal{I}_{28}, \mathcal{I}_{29}, \mathcal{I}_{30}, \mathcal{I}_{31}, \mathcal{I}_{34}, \mathcal{I}_{36}, \mathcal{I}_{37}, \mathcal{I}_{46}, \mathcal{I}_{48}, \mathcal{I}_{51}, \mathcal{I}_{54}, \mathcal{I}_{55}, \mathcal{I}_{56}, \mathcal{I}_{66}, \mathcal{I}_{71}$
{3, 5, 8}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{14}, \mathcal{I}_{15}, \mathcal{I}_{20}, \mathcal{I}_{21}, \mathcal{I}_{22}, \mathcal{I}_{23}, \mathcal{I}_{25}, \mathcal{I}_{26}, \mathcal{I}_{29}, \mathcal{I}_{33}, \mathcal{I}_{39}, \mathcal{I}_{43}, \mathcal{I}_{44}, \mathcal{I}_{45}, \mathcal{I}_{49}, \mathcal{I}_{50}, \mathcal{I}_{57}, \mathcal{I}_{59}, \mathcal{I}_{61}, \mathcal{I}_{72}$
{3, 6, 7}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{16}, \mathcal{I}_{17}, \mathcal{I}_{18}, \mathcal{I}_{19}, \mathcal{I}_{22}, \mathcal{I}_{23}, \mathcal{I}_{27}, \mathcal{I}_{28}, \mathcal{I}_{29}, \mathcal{I}_{34}, \mathcal{I}_{40}, \mathcal{I}_{41}, \mathcal{I}_{42}, \mathcal{I}_{46}, \mathcal{I}_{48}, \mathcal{I}_{51}, \mathcal{I}_{58}, \mathcal{I}_{59}, \mathcal{I}_{62}, \mathcal{I}_{73}$
{3, 6, 8}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4, \mathcal{I}_5, \mathcal{I}_6, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{22}, \mathcal{I}_{23}, \mathcal{I}_{25}, \mathcal{I}_{26}, \mathcal{I}_{27}, \mathcal{I}_{28}, \mathcal{I}_{29}, \mathcal{I}_{47}, \mathcal{I}_{49}, \mathcal{I}_{51}, \mathcal{I}_{57}, \mathcal{I}_{58}, \mathcal{I}_{59}, \mathcal{I}_{74}$
{4, 6, 8}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{11}, \mathcal{I}_{12}, \mathcal{I}_{13}, \mathcal{I}_{24}, \mathcal{I}_{25}, \mathcal{I}_{26}, \mathcal{I}_{27}, \mathcal{I}_{28}, \mathcal{I}_{30}, \mathcal{I}_{31}, \mathcal{I}_{47}, \mathcal{I}_{52}, \mathcal{I}_{53}, \mathcal{I}_{54}, \mathcal{I}_{55}, \mathcal{I}_{57}, \mathcal{I}_{58}, \mathcal{I}_{67}, \mathcal{I}_{75}$
{1, 4, 5, 8}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{14}, \mathcal{I}_{15}, \mathcal{I}_{24}, \mathcal{I}_{35}, \mathcal{I}_{39}, \mathcal{I}_{63}$
{1, 4, 6, 7}	$\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_7, \mathcal{I}_8, \mathcal{I}_9, \mathcal{I}_{10}, \mathcal{I}_{16}, \mathcal{I}_{17}, \mathcal{I}_{24}, \mathcal{I}_{36}, \mathcal{I}_{40}, \mathcal{I}_{64}$

Table 1. “Pre”-master integrals supported on each of the 10 cuts necessary to construct the complete IBP reductions.

Here for example the notation $\{1, 5, 7\}$ means $\mathcal{S}_{\text{cut}} = (\zeta_1, \zeta_2, \zeta_3) = (1, 5, 7)$; that is, the triple cut,

$$z_1 \rightarrow 0, \quad z_5 \rightarrow 0, \quad z_7 \rightarrow 0. \quad (5.5)$$

Thus, for the hexagon-box diagram, the set of necessary cuts thus includes 8 triple cuts and 2 quadruple cuts. The “pre”-master integrals supported on these cuts are listed in table 1.

The advantage of applying cuts is that the number of integrals in the IBP relations on each cut will be much less than when no cuts are applied.

If we switch on global symmetries in AZURITE, then the number of independent master integrals is found to be 73. This is due to the additional symmetry relations,

$$\begin{aligned} \mathcal{I}_{63} &\equiv I(1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0) = I(1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 0), \\ \mathcal{I}_{68} &\equiv I(1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0) = I(0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0). \end{aligned} \quad (5.6)$$

Hence the set of 73 master integrals is,

$$\mathcal{F}_{\text{MI}} = \{\mathcal{I}_1, \dots, \mathcal{I}_{75}\} \setminus \{\mathcal{I}_{63}, \mathcal{I}_{68}\}. \quad (5.7)$$

In this paper, we first reduce integrals to the linear combination of 75 “pre”-master integrals and then apply eq. (5.6) to further achieve the reduction to the linearly independent 73 master integrals.

To demonstrate the power of our method, we show how to reduce all the numerator-degree-4, 3, 2 and 1 hexagon-box integrals (our target integrals),

$$\begin{aligned}
 &\{I(1, 1, 1, 1, 1, 1, 1, 0, 0, -4), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -1, -3), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -2, -2), \\
 &I(1, 1, 1, 1, 1, 1, 1, 0, -3, -1), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -4, 0), \quad I(1, 1, 1, 1, 1, 1, 1, -1, 0, -3), \\
 &I(1, 1, 1, 1, 1, 1, 1, -1, -1, -2), \quad I(1, 1, 1, 1, 1, 1, 1, -1, -2, -1), \quad I(1, 1, 1, 1, 1, 1, 1, -1, -3, 0), \\
 &I(1, 1, 1, 1, 1, 1, 1, -2, 0, -2), \quad I(1, 1, 1, 1, 1, 1, 1, -2, -1, -1), \quad I(1, 1, 1, 1, 1, 1, 1, -2, -2, 0), \\
 &I(1, 1, 1, 1, 1, 1, 1, -3, 0, -1), \quad I(1, 1, 1, 1, 1, 1, 1, -3, -1, 0), \quad I(1, 1, 1, 1, 1, 1, 1, -4, 0, 0), \\
 &I(1, 1, 1, 1, 1, 1, 1, 0, 0, -3), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -1, -2), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -2, -1), \\
 &I(1, 1, 1, 1, 1, 1, 1, 0, -3, 0), \quad I(1, 1, 1, 1, 1, 1, 1, -1, 0, -2), \quad I(1, 1, 1, 1, 1, 1, 1, -1, -1, -1), \\
 &I(1, 1, 1, 1, 1, 1, 1, -1, -2, 0), \quad I(1, 1, 1, 1, 1, 1, 1, -2, 0, -1), \quad I(1, 1, 1, 1, 1, 1, 1, -2, -1, 0), \\
 &I(1, 1, 1, 1, 1, 1, 1, -3, 0, 0), \quad I(1, 1, 1, 1, 1, 1, 1, 0, 0, -2), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -1, -1), \\
 &I(1, 1, 1, 1, 1, 1, 1, 0, -2, 0), \quad I(1, 1, 1, 1, 1, 1, 1, -1, 0, -1), \quad I(1, 1, 1, 1, 1, 1, 1, -1, -1, 0), \\
 &I(1, 1, 1, 1, 1, 1, 1, 0, 0, -1), \quad I(1, 1, 1, 1, 1, 1, 1, 0, -1, 0)\}
 \end{aligned} \tag{5.8}$$

analytically to express them as linear combinations of the 73 master integrals.

5.1 Module intersection on cuts

In this section we show explicitly how to apply our module intersection method on cuts, in order to obtain simplified IBP systems (i.e., which do not involve integrals with doubled propagators) on unitarity cuts.

The module M_1 defined in subsection 2.1 for the hexagon-box diagram without cuts applied, is generated by the following $2 \times (4 + 2) = 12$ generators, cf. eq. (2.29),

$$\begin{aligned}
 &(z_1 - z_2, z_1 - z_2, -s_{12} + z_1 - z_2, -s_{12} - s_{13} + z_1 - z_2, s_{14} + z_1 - z_2 - z_8 + z_{10}, z_1 - z_2 - z_8 + z_{10}, \\
 &\quad 0, 0, -s_{12} - s_{13} - s_{14} + z_1 - z_2, 0, 0) \\
 &(0, 0, 0, 0, s_{14} + z_1 - z_2 - z_8 + z_{10}, z_1 - z_2 - z_8 + z_{10}, s_{12} + s_{13} + s_{14} - z_8 + z_{10}, z_{10} - z_8, 0, z_{10} - z_8, s_{12} - z_8 + z_{10}) \\
 &(s_{12} + z_2 - z_3, z_2 - z_3, z_2 - z_3, -s_{23} + z_2 - z_3, s_{12} + s_{24} + z_2 - z_3 - z_8 + z_{11}, s_{12} + z_2 - z_3 - z_8 + z_{11}, \\
 &\quad 0, 0, -s_{23} - s_{24} + z_2 - z_3, 0, 0) \\
 &(0, 0, 0, 0, s_{12} + s_{24} + z_2 - z_3 - z_8 + z_{11}, s_{12} + z_2 - z_3 - z_8 + z_{11}, s_{12} + s_{23} + s_{24} - z_8 + z_{11}, z_{11} - z_8, \\
 &\quad 0, s_{12} - z_8 + z_{11}, z_{11} - z_8) \\
 &(s_{13} + s_{23} + z_3 - z_4, s_{23} + z_3 - z_4, z_3 - z_4, z_3 - z_4, -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_3 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, \\
 &\quad -s_{12} + z_3 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, 0, 0, s_{12} + s_{13} + s_{14} + s_{23} + s_{24} + z_3 - z_4, 0, 0) \\
 &(0, 0, 0, 0, -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_3 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} + z_3 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, \\
 &\quad -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_4 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} - s_{13} - s_{23} + z_4 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, \\
 &\quad 0, -s_{12} - s_{23} + z_4 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} - s_{13} + z_4 - z_5 + z_6 + z_7 + z_8 - z_9 - z_{10} - z_{11}) \\
 &(-s_{12} - s_{13} - s_{23} + z_4 - z_9, -s_{12} - s_{13} - s_{14} - s_{23} + z_4 - z_9, -s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_4 - z_9, z_4 - z_9, z_5 - z_6, z_5 - z_6, \\
 &\quad 0, 0, z_4 - z_9, 0, 0) \\
 &(0, 0, 0, 0, z_5 - z_6, z_5 - z_6, -z_4 + z_5 - z_6 + z_9, s_{12} + s_{13} + s_{23} - z_4 + z_5 - z_6 + z_9, \\
 &\quad 0, s_{12} + s_{13} + s_{14} + s_{23} - z_4 + z_5 - z_6 + z_9, s_{12} + s_{13} + s_{23} + s_{24} - z_4 + z_5 - z_6 + z_9) \\
 &(2z_1, z_1 + z_2, -s_{12} + z_1 + z_3, -s_{12} - s_{13} - s_{23} + z_1 + z_4, -s_{12} - s_{13} - s_{23} + z_1 + z_4 + z_6 - z_8 - z_9, z_1 + z_6 - z_8, 0, 0, z_1 + z_9, 0, 0) \\
 &(0, 0, 0, 0, -s_{12} - s_{13} - s_{23} + z_1 + z_4 + z_6 - z_8 - z_9, z_1 + z_6 - z_8, z_6 - z_8 - z_9, -z_1 + z_6 - z_8, \\
 &\quad 0, -z_2 + z_6 - z_8, s_{12} - z_1 + z_2 - z_3 + z_6 - z_8) \\
 &(-z_1 + z_6 - z_8, -z_1 + z_6 - z_{10}, -z_1 + z_6 + z_8 - z_{10} - z_{11}, s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_5 - z_7 + z_9, \\
 &\quad s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_5 + z_8 + z_9, -z_1 + z_6 + z_8, 0, 0, -z_1 + z_6 - z_7, 0, 0) \\
 &(0, 0, 0, 0, s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_5 + z_8 + z_9, -z_1 + z_6 + z_8, z_7 + z_8, 2z_8, 0, z_8 + z_{10}, z_8 + z_{11}).
 \end{aligned} \tag{5.9}$$

Note that the generators are at most linear in the z_i , and always homogeneous in z_1, \dots, z_{11} and $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$.

The module M_2 for the hexagon-box diagram without applied cuts is generated by the following 11 generators, cf. eq. (2.30),

$$\begin{aligned}
 & (z_1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), (0, z_2, 0, 0, 0, 0, 0, 0, 0, 0, 0) \\
 & (0, 0, z_3, 0, 0, 0, 0, 0, 0, 0, 0), (0, 0, 0, z_4, 0, 0, 0, 0, 0, 0, 0) \\
 & (0, 0, 0, 0, z_5, 0, 0, 0, 0, 0, 0), (0, 0, 0, 0, 0, z_6, 0, 0, 0, 0, 0) \\
 & (0, 0, 0, 0, 0, 0, z_7, 0, 0, 0, 0), (0, 0, 0, 0, 0, 0, 0, z_8, 0, 0, 0) \\
 & (0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0) \\
 & (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1).
 \end{aligned} \tag{5.10}$$

We now proceed to consider the modules on the unitarity cuts given in eq. (5.4). For example, for the cut $\{2, 5, 7\}$, we apply the replacements

$$z_2 \rightarrow 0, \quad z_5 \rightarrow 0, \quad z_7 \rightarrow 0. \tag{5.11}$$

The propagator indices are thus classified as,

$$\mathcal{S}_{\text{cut}} = \{2, 5, 7\}, \quad \mathcal{S}_{\text{uncut}} = \{1, 3, 4, 6, 8\}, \quad \mathcal{S}_{\text{ISP}} = \{9, 10, 11\}. \tag{5.12}$$

From eq. (2.23) it follows that the generators of \tilde{M}_1 on the cut $\{2, 5, 7\}$ take the form,

$$\begin{aligned}
 & (z_1, z_1, z_1 - s_{12}, -s_{12} - s_{13} + z_1, s_{14} + z_1 - z_8 + z_{10}, z_1 - z_8 + z_{10}, 0, 0, -s_{12} - s_{13} - s_{14} + z_1, 0, 0) \\
 & (0, 0, 0, 0, s_{14} + z_1 - z_8 + z_{10}, z_1 - z_8 + z_{10}, s_{12} + s_{13} + s_{14} - z_8 + z_{10}, z_{10} - z_8, 0, z_{10} - z_8, s_{12} - z_8 + z_{10}) \\
 & (s_{12} - z_3, -z_3, -z_3, -s_{23} - z_3, s_{12} + s_{24} - z_3 - z_8 + z_{11}, s_{12} - z_3 - z_8 + z_{11}, 0, 0, -s_{23} - s_{24} - z_3, 0, 0) \\
 & (0, 0, 0, 0, s_{12} + s_{24} - z_3 - z_8 + z_{11}, s_{12} - z_3 - z_8 + z_{11}, s_{12} + s_{23} + s_{24} - z_8 + z_{11}, z_{11} - z_8, 0, s_{12} - z_8 + z_{11}, z_{11} - z_8) \\
 & (s_{13} + s_{23} + z_3 - z_4, s_{23} + z_3 - z_4, z_3 - z_4, z_3 - z_4, -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_3 + z_6 + z_8 - z_9 - z_{10} - z_{11}, \\
 & \quad -s_{12} + z_3 + z_6 + z_8 - z_9 - z_{10} - z_{11}, 0, 0, s_{12} + s_{13} + s_{14} + s_{23} + s_{24} + z_3 - z_4, 0, 0) \\
 & (0, 0, 0, 0, -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_3 + z_6 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} + z_3 + z_6 + z_8 - z_9 - z_{10} - z_{11}, \\
 & \quad -2s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_4 + z_6 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} - s_{13} - s_{23} + z_4 + z_6 + z_8 - z_9 - z_{10} - z_{11}, \\
 & \quad 0, -s_{12} - s_{23} + z_4 + z_6 + z_8 - z_9 - z_{10} - z_{11}, -s_{12} - s_{13} + z_4 + z_6 + z_8 - z_9 - z_{10} - z_{11}) \\
 & (-s_{12} - s_{13} - s_{23} + z_4 - z_9, -s_{12} - s_{13} - s_{14} - s_{23} + z_4 - z_9, -s_{12} - s_{13} - s_{14} - s_{23} - s_{24} + z_4 - z_9, z_4 - z_9, -z_6, -z_6, \\
 & \quad 0, 0, z_4 - z_9, 0, 0) \\
 & (0, 0, 0, 0, -z_6, -z_6, -z_4 - z_6 + z_9, s_{12} + s_{13} + s_{23} - z_4 - z_6 + z_9, \\
 & \quad 0, s_{12} + s_{13} + s_{14} + s_{23} - z_4 - z_6 + z_9, s_{12} + s_{13} + s_{23} + s_{24} - z_4 - z_6 + z_9) \\
 & (2z_1, z_1, -s_{12} + z_1 + z_3, -s_{12} - s_{13} - s_{23} + z_1 + z_4, -s_{12} - s_{13} - s_{23} + z_1 + z_4 + z_6 - z_8 - z_9, z_1 + z_6 - z_8, 0, 0, z_1 + z_9, 0, 0) \\
 & (0, 0, 0, 0, -s_{12} - s_{13} - s_{23} + z_1 + z_4 + z_6 - z_8 - z_9, z_1 + z_6 - z_8, z_6 - z_8 - z_9, -z_1 + z_6 - z_8, 0, z_6 - z_8, s_{12} - z_1 - z_3 + z_6 - z_8) \\
 & (-z_1 + z_6 - z_8, -z_1 + z_6 - z_{10}, -z_1 + z_6 + z_8 - z_{10} - z_{11}, s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_9, s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_8 + z_9, \\
 & \quad -z_1 + z_6 + z_8, 0, 0, z_6 - z_1, 0, 0) \\
 & (0, 0, 0, 0, s_{12} + s_{13} + s_{23} - z_1 - z_4 + z_8 + z_9, -z_1 + z_6 + z_8, z_8, 2z_8, 0, z_8 + z_{10}, z_8 + z_{11}),
 \end{aligned} \tag{5.13}$$

cut	time / sec	mem/GB
{1, 5, 7}	218	4.3
{2, 5, 7}	43	1.1
{2, 5, 8}	303	6.7
{2, 6, 7}	743	9.8
{3, 5, 8}	404	7.4
{3, 6, 7}	699	11.0
{3, 6, 8}	24	1.0
{4, 6, 8}	797	13.7
{1, 4, 5, 8}	53	1.7
{1, 4, 6, 7}	196	3.0

Table 2. Timings and RAM usages for the module intersection computations for the relevant cuts of the non-planar hexagon-box diagram.

whereas the generators of \tilde{M}_2 on the cut $\{2, 5, 7\}$ take the form,

$$\begin{aligned}
 & (z_1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0), (0, 0, z_3, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0) \\
 & (0, 0, 0, z_4, 0, 0, 0, 0, 0, 0, 0, 0, 0), (0, 0, 0, 0, 0, z_6, 0, 0, 0, 0, 0, 0, 0) \\
 & (0, 0, 0, 0, 0, 0, 0, z_8, 0, 0, 0, 0, 0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0) \\
 & (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0), (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1). \tag{5.14}
 \end{aligned}$$

The intersection of \tilde{M}_1 and \tilde{M}_2 , with the generators given above, is then computed by the method described in section 3. In the case at hand, we proceed as follows.

1. First we compute the generators $\tilde{M}_1 \cap \tilde{M}_2$ in the ring, with the algorithm described in Lemma 3,

$$\mathbb{B} = \mathbb{Q}[z_1, z_3, z_4, z_6, z_8, z_9, z_{10}, z_{11}, s_{12}, s_{13}, s_{14}, s_{23}, s_{24}] \tag{5.15}$$

using a block ordering with $[z_1, z_3, z_4, z_6, z_8, z_9, z_{10}, z_{11}] \succ [s_{12}, s_{13}, s_{14}, s_{23}, s_{24}]$.

2. Then we map the generators of $\tilde{M}_1 \cap \tilde{M}_2$ from the previous step, to the ring,

$$\tilde{\mathbb{A}} = \mathbb{Q}(s_{12}, s_{13}, s_{14}, s_{23}, s_{24})[z_1, z_3, z_4, z_6, z_8, z_9, z_{10}, z_{11}] \tag{5.16}$$

and simplify the generators.

3. Finally, we use the heuristic algorithm (Algorithm 8) to delete redundant generators in $\tilde{M}_1 \cap \tilde{M}_2$.

These steps are automated by our SINGULAR program. The generators of $\tilde{M}_1 \cap \tilde{M}_2$ for the other 9 cuts listed in eq. (5.4) were obtained in the same manner. In table 2 we provide timings for the computation of the module intersections for the relevant cuts of the non-planar hexagon-box diagram. The timings are in seconds on an Intel Xeon E5-2643 machine with 24 cores, 3.40 GHz and 384 GB of RAM.

cut	original size / MB	trimmed size / MB
{1, 5, 7}	68	10
{2, 5, 7}	25	1.4
{2, 5, 8}	49	3.1
{2, 6, 7}	100	2.8
{3, 5, 8}	97	3.7
{3, 6, 7}	80	3.6
{3, 6, 8}	10	1.6
{4, 6, 8}	21	1.6
{1, 4, 5, 8}	4.4	3.6
{1, 4, 6, 7}	9.4	4.1

Table 3. String sizes of the original and trimmed generating systems, given in megabytes.

Furthermore, using the heuristics as specified in Algorithm 8 we are able to reduce the size of the generating systems as specified in table 3. For instance, the trimmed generating system for the module intersection on the cut {2, 5, 7} consists of 24 generators, which are fully analytic in $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$. The generators can be downloaded from

https://raw.githubusercontent.com/yzhphy/hexagonbox_reduction/master/cut257/module_intersection_257.txt.

Each list in this file is in the format

$$(\tilde{b}_1, 0, \tilde{b}_3, \tilde{b}_4, 0, \tilde{b}_6, 0, \tilde{b}_8, \tilde{a}_9, \tilde{a}_{10}, \tilde{a}_{11}, \tilde{b}). \tag{5.17}$$

5.2 Reduction of IBP identities on cuts

After computing the module intersections on the 10 cuts, we use eq. (2.26) to generate IBP identities without doubled propagators on each cut in turn. As described in section 4, we use linear algebra techniques to select the relevant and independent IBP identities for reducing the target integrals to the master integrals on each cut. Characteristics of the resulting linearly independent linear systems, all analytic in $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$ and the spacetime dimension D , are presented in table 4. It is clear from this table that these linear systems are very sparse and of relatively small byte size.

For the purpose of demonstration, we have made the IBP relations on the cut {2, 5, 7} available at

https://raw.githubusercontent.com/yzhphy/hexagonbox_reduction/master/cut257/hexagonbox_257_deg4.txt.

We calculated the row reduced echelon form (RREF) of these linear systems using the *total pivoting* strategy, in order to retain the sparsity in intermediate steps. For the linear systems corresponding to some cuts, we are able to directly obtain the RREF analytically in $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$ and the spacetime dimension D . For linear systems corresponding

cut	# equations	# integrals	byte size / MB	density
{1, 5, 7}	1144	1177	1.2	1.4%
{2, 5, 7}	1170	1210	0.99	1.3%
{2, 5, 8}	1152	1190	1.1	1.5%
{2, 6, 7}	1118	1155	1.0	1.5%
{3, 5, 8}	1160	1202	1.2	1.5%
{3, 6, 7}	1173	1217	1.3	1.7%
{3, 6, 8}	1135	1176	0.77	1.2%
{4, 6, 8}	1140	1176	0.94	1.2%
{1, 4, 5, 8}	700	723	0.69	1.7%
{1, 4, 6, 7}	683	706	0.66	1.6%

Table 4. Linear systems of IBP identities on the 10 cuts given in eq. (5.4) for reducing the target hexagon-box integrals in eq. (5.8). The storage size is for the corresponding matrix and is measured in megabytes. The density refers to the percentage of non-vanishing elements in the matrices of these linear systems.

to other cuts, we calculate the RREF with integer values of one or two s_{ij} repeatedly. The fully analytic RREF is then readily obtained by our private heuristic interpolation algorithm. These computations are implemented in our primitive MATHEMATICA code. A much more efficient RREF code in SINGULAR is in preparation.

We also remark that for the computation of different cuts, a useful trick is to work with different choices of independent Mandelstam invariants. For a specific cut, a suitable choice of Mandelstam invariants can speed up the computation and also save the RAM usage. After the RREF is obtained, we use the program FERMAT [69] to replace the new Mandelstam variables by the original choice $s_{12}, s_{13}, s_{14}, s_{23}, s_{24}$.

The running time and resources required of our MATHEMATICA code depends on the size of the linear systems and also *the complexity of coefficients* in the reduced IBPs. For the smallest linear system, the one corresponding to the quadruple cut $\{1, 4, 6, 7\}$, our MATHEMATICA RREF code obtained the fully analytical RREF in 31 minutes with one core and 1.5 GB RAM usage on a laptop with 16 GB RAM. For the largest linear system, that corresponding to the triple cut $\{3, 6, 7\}$, we run our MATHEMATICA code and assign integer values to two Mandelstam invariants. This finished in 2.5 hours and used 1.8 GB RAM. We parallelize the running with various integer values, evaluating 440 points on the IRIDIS High Performance Computing Facility. The semi-analytic results are then interpolated to the fully analytic RREF result by our heuristic multivariate interpolation algorithm, requiring a CPU time of 23 minutes with one core and 15 GB RAM usage.

5.3 Merging of IBP reductions and final result

Having obtained the RREF of the 10 IBP systems on cuts, it is straightforward to merge the coefficients to obtain the complete IBP reductions without applied cuts [13]. For example, to determine the reduction coefficient of $\mathcal{I}_{42} = I(1, 0, 1, 0, 1, 1, 1, 0, 0, 0, 0)$, we search for \mathcal{I}_{42} in table 1 and find that it is supported on the cuts $\{1, 5, 7\}$ and $\{3, 6, 7\}$. We find that,

for every target integral, the reduction coefficient for \mathcal{I}_{42} on the cut $\{1, 5, 7\}$ equals the corresponding coefficient on the cut $\{3, 6, 7\}$, as must be the case. Thus, for each target integral in turn, we obtain its reduction coefficient of each \mathcal{I}_j as the reduction coefficient of \mathcal{I}_j on each cut supporting \mathcal{I}_j . In this way we obtain the complete reduction of the target integral *without applied cuts*.

After merging the 10 RREFs we reduce the 32 target integrals in eq. (5.8) to the 75 “pre”-master integrals $\{\mathcal{I}_1, \dots, \mathcal{I}_{75}\}$. We then apply the global symmetry in eq. (5.6) to eliminate the redundant integrals \mathcal{I}_{63} and \mathcal{I}_{68} , and then finally obtain the complete analytic reduction to the 73 master integrals. The IBP reduction result, as replacement rules (in a compressed file of the size 280 MB), can be downloaded from the link provided in the introduction.

5.4 Comparison with other IBP solvers

We have checked our results with FIRE5 [5] in the C++ implementation with LiteRed [8], and KIRA (v 1.1) [9]. We note that it is not an easy task to perform the analytic IBP reduction of hexagon-box integrals with degree-four numerators. Due to the RAM limit we faced, we have not yet been able to obtain the analytic IBP reduction for degree-four-numerator hexagon-box integrals from FIRE5 or KIRA with the Rackham cluster, on a node with two 10 core Intel Xeon V4 CPU and 256 GB of RAM, or on the Kebnekaise cluster large memory machines, four Intel Xeon E7-8860v4 with 18 cores each and 3072 GB of RAM. On the other hand, it is easy to obtain the numeric IBP reduction for degree-four-numerator hexagon-box integrals with FIRE5 and KIRA. For example, if all the five s_{ij} are taken to be integers, FIRE5 is able to generate the purely numeric IBP reductions in about 6.0 hours. We ran FIRE5 purely numerically for many sets of integer values, and all the numeric IBP reduction results are consistent with our analytic IBP reductions, after a basis change between the AZURITE integral basis and the FIRE basis.

6 Conclusion and outlook

In this paper we have presented a new and efficient method for computing integration-by-parts (IBP) reductions based on the ideas developed in refs. [13, 52]. We used a module intersection method [52] to trim IBP systems on unitarity cuts. The key idea is the efficient analytic computation of module intersections, which is achieved by the mathematical technique of treating the kinematical parameters as variables and using a block monomial ordering for which [variables] \succ [parameters]. This trick could also be helpful for other types of multi-loop multi-scale amplitude computations. After solving the module intersection problems, we find linear IBP systems on cuts which are of very small sizes. This part is implemented in our highly efficient and automated SINGULAR code. For example, the analytic IBP vectors for the hexagon-box integral with no doubled propagators on triple cuts can be computed in minutes.

Furthermore, we applied sophisticated sparse linear algebra techniques to compute the row reduced echelon form of the IBP identities on cuts. For example, we applied a weighted version of the Markowitz pivoting strategy to retain the sparsity in intermediate steps of

the row reduction. We have implemented the sparse linear algebra part of our algorithm as a preliminary `MATHEMATICA` code. A more efficient `SINGULAR` implementation will become available in the near future.

In this paper, we have solved a cutting-edge IBP reduction problem fully analytically: that of the reduction of non-planar five-point hexagon-box integrals, with numerators of degree four, to the basis of 73 master integrals. Our result has been verified (numerically) with the state-of-art IBP reduction programs.

We remark that increasing the maximal degree by one would require about twice as many linear equations and in turn require a factor of about $2^5 = 32$ times the current running time with the current implementations (taking also coefficient growth into account, see ref. [70]). However, with improvements of the implementation which are underway we expect to substantially reduce the CPU time in the near future. The planar five-gluon two-loop amplitude in QCD, which has a maximal numerator degree of 5 [44, 71], therefore appears within near reach. The non-planar amplitude involves degree-7 numerators for the double pentagon topology [71], while the maximal degree for the hexagon box is not currently known in the literature. Obtaining complete IBP reductions will therefore require significantly longer CPU time than the planar counterpart, but still appears feasible with the formalism presented in this paper.

We are currently preparing an automated implementation based on open-source software, such as `SINGULAR`, with which we expect to be able to solve yet more difficult IBP reduction problems. We expect that our method will boost the computation of NNLO cross sections for more complicated $2 \rightarrow 3$ scattering processes and higher-multiplicity cases.

Finally, we also expect that the ideas presented in this paper, such as the module intersection for trimming integral relations, the special ordering for variables and parameters, and sparse linear algebra techniques, can also be combined in various ways with other existing computational methods such as: the finite field sampling and reconstruction approach [14, 71], the dual conformal symmetry construction of IBP vectors [15] and the newly developed D-module method [19], in order to determine tailored optimal reduction strategies for various classes of Feynman integrals.

Acknowledgments

We thank Babis Anastasiou, Simon Badger, Zvi Bern, Rutger Boels, Christian Bogner, Jorrit Bosma, Charles Bouillaguet, Wolfram Decker, Lance Dixon, James Drummond, Ömer Gürdoğan, Johannes Henn, Enrico Herrmann, Harald Ita, Mikhail Kalmykov, David Kosower, Dirk Kreimer, Roman N. Lee, Hui Luo, Andreas von Manteuffel, Alexander Mitov, David Mond, Erik Panzer, Costas Papadopoulos, Tiziano Peraro, Gerhard Pfister, Robert Schabinger, Peter Uwer, Gang Yang and Mao Zeng for very enlightening discussions. The research leading to these results has received funding from Swiss National Science Foundation (Ambizione grant PZ00P2 161341), from the National Science Foundation under Grant No. NSF PHY17-48958, and from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 725110). The work of YZ is also partially supported by the Swiss Na-

tional Science Foundation through the NCCR SwissMap, Grant number 141869. The work of AG is supported by the Knut and Alice Wallenberg Foundation under grant #2015-0083. The work of KJL is supported by ERC-2014-CoG, Grant number 648630 IQFT. The work of JB and HS was supported by Project II.5 of SFB-TRR 195 “Symbolic Tools in Mathematics and their Application” of the German Research Foundation (DFG). The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton. Part of the computations were also performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at Uppmax and HPC2N. The authors moreover acknowledge the use of the Euler computing cluster, associated with ETH Zürich.

Open Access. This article is distributed under the terms of the Creative Commons Attribution License ([CC-BY 4.0](https://creativecommons.org/licenses/by/4.0/)), which permits any use, distribution and reproduction in any medium, provided the original author(s) and source are credited.

References

- [1] S. Laporta, *Calculation of master integrals by difference equations*, *Phys. Lett. B* **504** (2001) 188 [[hep-ph/0102032](https://arxiv.org/abs/hep-ph/0102032)] [[INSPIRE](https://inspirehep.net/literature/48107)].
- [2] S. Laporta, *High precision calculation of multiloop Feynman integrals by difference equations*, *Int. J. Mod. Phys. A* **15** (2000) 5087 [[hep-ph/0102033](https://arxiv.org/abs/hep-ph/0102033)] [[INSPIRE](https://inspirehep.net/literature/48107)].
- [3] C. Anastasiou and A. Lazopoulos, *Automatic integral reduction for higher order perturbative calculations*, *JHEP* **07** (2004) 046 [[hep-ph/0404258](https://arxiv.org/abs/hep-ph/0404258)] [[INSPIRE](https://inspirehep.net/literature/10250)].
- [4] A.V. Smirnov, *Algorithm FIRE — Feynman Integral REduction*, *JHEP* **10** (2008) 107 [[arXiv:0807.3243](https://arxiv.org/abs/0807.3243)] [[INSPIRE](https://inspirehep.net/literature/77616)].
- [5] A.V. Smirnov, *FIRE5: a C++ implementation of Feynman Integral REduction*, *Comput. Phys. Commun.* **189** (2015) 182 [[arXiv:1408.2372](https://arxiv.org/abs/1408.2372)] [[INSPIRE](https://inspirehep.net/literature/12770)].
- [6] C. Studerus, *Reduze-Feynman Integral Reduction in C++*, *Comput. Phys. Commun.* **181** (2010) 1293 [[arXiv:0912.2546](https://arxiv.org/abs/0912.2546)] [[INSPIRE](https://inspirehep.net/literature/8440)].
- [7] A. von Manteuffel and C. Studerus, *Reduze 2 — Distributed Feynman Integral Reduction*, [arXiv:1201.4330](https://arxiv.org/abs/1201.4330) [[INSPIRE](https://inspirehep.net/literature/10250)].
- [8] R.N. Lee, *Presenting LiteRed: a tool for the Loop InTEgrals REDuction*, [arXiv:1212.2685](https://arxiv.org/abs/1212.2685) [[INSPIRE](https://inspirehep.net/literature/11278)].
- [9] P. Maierhöfer, J. Usovitsch and P. Uwer, *Kira — A Feynman integral reduction program*, *Comput. Phys. Commun.* **230** (2018) 99 [[arXiv:1705.05610](https://arxiv.org/abs/1705.05610)] [[INSPIRE](https://inspirehep.net/literature/14700)].
- [10] J. Gluza, K. Kajda and D.A. Kosower, *Towards a Basis for Planar Two-Loop Integrals*, *Phys. Rev. D* **83** (2011) 045012 [[arXiv:1009.0472](https://arxiv.org/abs/1009.0472)] [[INSPIRE](https://inspirehep.net/literature/8800)].
- [11] R.M. Schabinger, *A New Algorithm For The Generation Of Unitarity-Compatible Integration By Parts Relations*, *JHEP* **01** (2012) 077 [[arXiv:1111.4220](https://arxiv.org/abs/1111.4220)] [[INSPIRE](https://inspirehep.net/literature/9900)].
- [12] H. Ita, *Two-loop Integrand Decomposition into Master Integrals and Surface Terms*, *Phys. Rev. D* **94** (2016) 116015 [[arXiv:1510.05626](https://arxiv.org/abs/1510.05626)] [[INSPIRE](https://inspirehep.net/literature/13700)].
- [13] K.J. Larsen and Y. Zhang, *Integration-by-parts reductions from unitarity cuts and algebraic geometry*, *Phys. Rev. D* **93** (2016) 041701 [[arXiv:1511.01071](https://arxiv.org/abs/1511.01071)] [[INSPIRE](https://inspirehep.net/literature/13700)].

- [14] A. von Manteuffel and R.M. Schabinger, *A novel approach to integration by parts reduction*, *Phys. Lett. B* **744** (2015) 101 [[arXiv:1406.4513](#)] [[INSPIRE](#)].
- [15] Z. Bern, M. Enciso, H. Ita and M. Zeng, *Dual Conformal Symmetry, Integration-by-Parts Reduction, Differential Equations and the Nonplanar Sector*, *Phys. Rev. D* **96** (2017) 096017 [[arXiv:1709.06055](#)] [[INSPIRE](#)].
- [16] D.A. Kosower, *Direct Solution of Integration-by-Parts Systems*, *Phys. Rev. D* **98** (2018) 025008 [[arXiv:1804.00131](#)] [[INSPIRE](#)].
- [17] R.N. Lee and A.A. Pomeransky, *Critical points and number of master integrals*, *JHEP* **11** (2013) 165 [[arXiv:1308.6676](#)] [[INSPIRE](#)].
- [18] A. Georgoudis, K.J. Larsen and Y. Zhang, *Azurite: An algebraic geometry based package for finding bases of loop integrals*, *Comput. Phys. Commun.* **221** (2017) 203 [[arXiv:1612.04252](#)] [[INSPIRE](#)].
- [19] T. Bitoun, C. Bogner, R.P. Klausen and E. Panzer, *Feynman integral relations from parametric annihilators*, [arXiv:1712.09215](#) [[INSPIRE](#)].
- [20] R. Britto, F. Cachazo and B. Feng, *Generalized unitarity and one-loop amplitudes in $N = 4$ super-Yang-Mills*, *Nucl. Phys. B* **725** (2005) 275 [[hep-th/0412103](#)] [[INSPIRE](#)].
- [21] R. Britto, F. Cachazo, B. Feng and E. Witten, *Direct proof of tree-level recursion relation in Yang-Mills theory*, *Phys. Rev. Lett.* **94** (2005) 181602 [[hep-th/0501052](#)] [[INSPIRE](#)].
- [22] D.A. Kosower and K.J. Larsen, *Maximal Unitarity at Two Loops*, *Phys. Rev. D* **85** (2012) 045017 [[arXiv:1108.1180](#)] [[INSPIRE](#)].
- [23] H. Johansson, D.A. Kosower and K.J. Larsen, *Two-Loop Maximal Unitarity with External Masses*, *Phys. Rev. D* **87** (2013) 025030 [[arXiv:1208.1754](#)] [[INSPIRE](#)].
- [24] G. Ossola, C.G. Papadopoulos and R. Pittau, *Reducing full one-loop amplitudes to scalar integrals at the integrand level*, *Nucl. Phys. B* **763** (2007) 147 [[hep-ph/0609007](#)] [[INSPIRE](#)].
- [25] S. Badger, H. Frellesvig and Y. Zhang, *Hepta-Cuts of Two-Loop Scattering Amplitudes*, *JHEP* **04** (2012) 055 [[arXiv:1202.2019](#)] [[INSPIRE](#)].
- [26] Y. Zhang, *Integrand-Level Reduction of Loop Amplitudes by Computational Algebraic Geometry Methods*, *JHEP* **09** (2012) 042 [[arXiv:1205.5707](#)] [[INSPIRE](#)].
- [27] P. Mastrolia, E. Mirabella, G. Ossola and T. Peraro, *Scattering Amplitudes from Multivariate Polynomial Division*, *Phys. Lett. B* **718** (2012) 173 [[arXiv:1205.7087](#)] [[INSPIRE](#)].
- [28] A.V. Kotikov, *Differential equations method: New technique for massive Feynman diagrams calculation*, *Phys. Lett. B* **254** (1991) 158 [[INSPIRE](#)].
- [29] A.V. Kotikov, *Differential equation method: The Calculation of N point Feynman diagrams*, *Phys. Lett. B* **267** (1991) 123 [Erratum *ibid.* **B 295** (1992) 409] [[INSPIRE](#)].
- [30] Z. Bern, L.J. Dixon and D.A. Kosower, *Dimensionally regulated pentagon integrals*, *Nucl. Phys. B* **412** (1994) 751 [[hep-ph/9306240](#)] [[INSPIRE](#)].
- [31] E. Remiddi, *Differential equations for Feynman graph amplitudes*, *Nuovo Cim. A* **110** (1997) 1435 [[hep-th/9711188](#)] [[INSPIRE](#)].
- [32] T. Gehrmann and E. Remiddi, *Differential equations for two loop four point functions*, *Nucl. Phys. B* **580** (2000) 485 [[hep-ph/9912329](#)] [[INSPIRE](#)].
- [33] J.M. Henn, *Multiloop integrals in dimensional regularization made simple*, *Phys. Rev. Lett.* **110** (2013) 251601 [[arXiv:1304.1806](#)] [[INSPIRE](#)].

- [34] C.G. Papadopoulos, *Simplified differential equations approach for Master Integrals*, *JHEP* **07** (2014) 088 [[arXiv:1401.6057](#)] [[INSPIRE](#)].
- [35] R.N. Lee, *Reducing differential equations for multiloop master integrals*, *JHEP* **04** (2015) 108 [[arXiv:1411.0911](#)] [[INSPIRE](#)].
- [36] J. Ablinger, A. Behring, J. Blümlein, A. De Freitas, A. von Manteuffel and C. Schneider, *Calculating Three Loop Ladder and V-Topologies for Massive Operator Matrix Elements by Computer Algebra*, *Comput. Phys. Commun.* **202** (2016) 33 [[arXiv:1509.08324](#)] [[INSPIRE](#)].
- [37] C.G. Papadopoulos, D. Tommasini and C. Wever, *The Pentabox Master Integrals with the Simplified Differential Equations approach*, *JHEP* **04** (2016) 078 [[arXiv:1511.09404](#)] [[INSPIRE](#)].
- [38] X. Liu, Y.-Q. Ma and C.-Y. Wang, *A Systematic and Efficient Method to Compute Multi-loop Master Integrals*, *Phys. Lett. B* **779** (2018) 353 [[arXiv:1711.09572](#)] [[INSPIRE](#)].
- [39] S. Badger, H. Frellesvig and Y. Zhang, *A Two-Loop Five-Gluon Helicity Amplitude in QCD*, *JHEP* **12** (2013) 045 [[arXiv:1310.1051](#)] [[INSPIRE](#)].
- [40] S. Badger, G. Mogull, A. Ochirov and D. O’Connell, *A Complete Two-Loop, Five-Gluon Helicity Amplitude in Yang-Mills Theory*, *JHEP* **10** (2015) 064 [[arXiv:1507.08797](#)] [[INSPIRE](#)].
- [41] 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 [Erratum *ibid.* **116** (2016) 189903] [[arXiv:1511.05409](#)] [[INSPIRE](#)].
- [42] S. Badger, C. Brønnum-Hansen, H.B. Hartanto and T. Peraro, *First look at two-loop five-gluon scattering in QCD*, *Phys. Rev. Lett.* **120** (2018) 092001 [[arXiv:1712.02229](#)] [[INSPIRE](#)].
- [43] S. Abreu, F. Febres Cordero, H. Ita, B. Page and M. Zeng, *Planar Two-Loop Five-Gluon Amplitudes from Numerical Unitarity*, *Phys. Rev. D* **97** (2018) 116014 [[arXiv:1712.03946](#)] [[INSPIRE](#)].
- [44] R.H. Boels, Q. Jin and H. Luo, *Efficient integrand reduction for particles with spin*, [arXiv:1802.06761](#) [[INSPIRE](#)].
- [45] H.A. Chawdhry, M.A. Lim and A. Mitov, *Two-loop five-point massless QCD amplitudes within the IBP approach*, [arXiv:1805.09182](#) [[INSPIRE](#)].
- [46] D. Chicherin, J.M. Henn and E. Sokatchev, *Implications of nonplanar dual conformal symmetry*, [arXiv:1807.06321](#) [[INSPIRE](#)].
- [47] T. Gehrmann, J.M. Henn and N.A. Lo Presti, *Pentagon functions for massless planar scattering amplitudes*, [arXiv:1807.09812](#) [[INSPIRE](#)].
- [48] S. Abreu, B. Page and M. Zeng, *Differential equations from unitarity cuts: nonplanar hexa-box integrals*, [arXiv:1807.11522](#) [[INSPIRE](#)].
- [49] O.V. Tarasov, *Connection between Feynman integrals having different values of the space-time dimension*, *Phys. Rev. D* **54** (1996) 6479 [[hep-th/9606018](#)] [[INSPIRE](#)].
- [50] R.N. Lee, *Space-time dimensionality D as complex variable: Calculating loop integrals using dimensional recurrence relation and analytical properties with respect to D* , *Nucl. Phys. B* **830** (2010) 474 [[arXiv:0911.0252](#)] [[INSPIRE](#)].
- [51] R.N. Lee, *Calculating multiloop integrals using dimensional recurrence relation and D -analyticity*, *Nucl. Phys. Proc. Suppl.* **205-206** (2010) 135 [[arXiv:1007.2256](#)] [[INSPIRE](#)].

- [52] Y. Zhang, *Lecture Notes on Multi-loop Integral Reduction and Applied Algebraic Geometry*, 2016, [arXiv:1612.02249](#) [INSPIRE].
- [53] P. Gianni, B. Trager and G. Zacharias, *Gröbner bases and primary decomposition of polynomial ideals*, *J. Symb. Comput.* **6** (1988) 149.
- [54] W. Decker, G.-M. Greuel, G. Pfister and H. Schönemann, *SINGULAR 4-1-1 — A computer algebra system for polynomial computations*, <http://www.singular.uni-kl.de> (2018).
- [55] P.A. Baikov, *Explicit solutions of the three loop vacuum integral recurrence relations*, *Phys. Lett.* **B 385** (1996) 404 [[hep-ph/9603267](#)] [INSPIRE].
- [56] R.N. Lee, *Modern techniques of multiloop calculations*, in *Proceedings, 49th Rencontres de Moriond on QCD and High Energy Interactions*, La Thuile, Italy, March 22–29, 2014, pp. 297–300 (2014) [[arXiv:1405.5616](#)] [INSPIRE].
- [57] H. Hauser and G. Müller, *Affine varieties and lie algebras of vector fields*, *Manuscripta Math.* **80** (1993) 309.
- [58] J. Böhm, A. Georgoudis, K.J. Larsen, M. Schulze and Y. Zhang, *Complete sets of logarithmic vector fields for integration-by-parts identities of Feynman integrals*, *Phys. Rev.* **D 98** (2018) 025023 [[arXiv:1712.09737](#)] [INSPIRE].
- [59] R.N. Lee and A.A. Pomeransky, *Normalized Fuchsian form on Riemann sphere and differential equations for multiloop integrals*, [arXiv:1707.07856](#) [INSPIRE].
- [60] D. Eisenbud, *Commutative algebra — With a view toward algebraic geometry*, *Graduate Texts in Mathematics*, vol. 150, Springer-Verlag, New York (1995).
- [61] T. Józefiak, *Ideals generated by minors of a symmetric matrix*, *Comment. Math. Helv.* **53** (1978) 595.
- [62] G.-M. Greuel and G. Pfister, *A Singular Introduction to Commutative Algebra*, 2nd ed., Springer Publishing Company, Incorporated (2007).
- [63] J. Böhm, W. Decker, C. Fieker and G. Pfister, *The use of Bad Primes in Rational Reconstruction*, *Math. Comp.* **84** (2015) 3013 [[arXiv:1207.1651](#)].
- [64] J. Böhm, W. Decker, C. Fieker, S. Laplagne and G. Pfister, *Bad Primes in Computational Algebraic Geometry*, *LNCS* **9725** (2016) 93 [[arXiv:1702.06920](#)].
- [65] The SpaSM group, *SpaSM: a Sparse direct Solver Modulo p*, v1.2 ed. (2017) [<http://github.com/cbouilla/spasm>].
- [66] H.M. Markowitz, *The elimination form of the inverse and its application to linear programming*, *Manage. Sci.* **3** (1957) 255.
- [67] D. Chicherin, J. Henn and V. Mitev, *Bootstrapping pentagon functions*, *JHEP* **05** (2018) 164 [[arXiv:1712.09610](#)] [INSPIRE].
- [68] D. Chicherin, J.M. Henn and E. Sokatchev, *Scattering Amplitudes from Superconformal Ward Identities*, *Phys. Rev. Lett.* **121** (2018) 021602 [[arXiv:1804.03571](#)] [INSPIRE].
- [69] R.H. Lewis, *Computer Algebra System Fermat*, v6.1.9 ed. (2018) [<http://home.bway.net/lewis/>].
- [70] H.R. Lee and B.D. Saunders, *Fraction Free Gaussian Elimination for Sparse Matrices*, *J. Symb. Comput.* **19** (1995) 393.
- [71] T. Peraro, *Scattering amplitudes over finite fields and multivariate functional reconstruction*, *JHEP* **12** (2016) 030 [[arXiv:1608.01902](#)] [INSPIRE].