

Cluster Subalgebra-Constructibility I: Introduction and Review of Cluster Algebras for Integrated Amplitudes

John Golden^{1,2} and Andrew J. McLeod^{2,3,4}

¹ *Leinweber Center for Theoretical Physics and Randall Laboratory of Physics, Department of Physics, University of Michigan Ann Arbor, MI 48109, USA*

² *Kavli Institute for Theoretical Physics, UC Santa Barbara, Santa Barbara, CA 93106, USA*

³ *SLAC National Accelerator Laboratory, Stanford University, Stanford, CA 94309, USA*

⁴ *Niels Bohr International Academy, Blegdamsvej 17, 2100 Copenhagen, Denmark*

ABSTRACT: We review connections between cluster algebras and integrated amplitudes in $\mathcal{N} = 4$ supersymmetric Yang-Mills. After a simple, pedagogical introduction to cluster algebras and some of their applications to amplitudes, we investigate ways in which amplitudes can be written as sums over cluster *subalgebras*. As an example, we exhaustively search for all ways in which the ‘nonclassical’ part of the two-loop, seven-particle remainder function can be decomposed into the cluster subalgebras of the Grassmannian $\text{Gr}(4,7)$.

Contents

1	Introduction	2
2	A Brief Introduction to Cluster Algebras	4
2.1	Clusters, Mutations, and Cluster \mathcal{A} -coordinates	6
2.2	Cluster \mathcal{X} -coordinates	8
2.3	Subalgebras and Associahedra	10
2.4	Grassmannian Cluster Algebras and Planar $\mathcal{N} = 4$ SYM	13
2.5	Finite Cluster Algebras	17
2.6	Cluster Automorphisms	20
3	Cluster Polylogarithms and MHV Amplitudes	23
3.1	The Symbol and Cobracket	25
3.2	Cluster-Algebraic Structure in at Two Loops	27
3.3	Subalgebra Structure and Cluster Polylogarithms	31
4	Cluster Subalgebra-Constructibility	34
4.1	A_2 Functions are a Basis for Cluster Polylogarithms	34
4.2	A_2 -constructibility	36
4.3	A_2 -constructibility for Larger Algebras	38
4.4	A_3 -constructibility	39
5	Subalgebra-Constructibility of $R_7^{(2)}$	40
5.1	D_5 Decompositions of $R_7^{(2)}$	40
5.2	A_5 Decompositions of $R_7^{(2)}$	42
6	Conclusion	43
A	Integrability and Adjacency for A_2	44
B	Counting Subalgebras of Finite Cluster Algebras	44
C	Cobracket Spaces in Finite Cluster Algebras	46

1 Introduction

Multi-loop scattering amplitudes in the planar limit of $\mathcal{N} = 4$ supersymmetric Yang-Mills (SYM) theory exhibit a great deal of intriguing mathematical structure. Much of this structure, at least at low loops and particle multiplicity, seems to be intimately tied to the cluster algebras associated with the Grassmannian $\text{Gr}(4, n)$ []. This is especially true in the maximally-helicity-violating (MHV) sector, where amplitudes have been computed at high loop orders in six- and seven-particle kinematics [1–5], and algorithms exist for calculating two loop amplitudes for any number of particles [6, 7]. Remarkably, each of the branch cuts in these amplitudes ends at the vanishing locus of some cluster coordinate on $\text{Gr}(4, n)$ [8–11], and—even more strikingly—their iterated discontinuities vanishes unless sequentially taken in coordinates that appear together in a cluster of $\text{Gr}(4, n)$ [12, 13]. All known next-to-MHV (NMHV) amplitudes in this theory share these remarkable properties [3–5, 14–16], as do certain classes of Feynman integrals [12, 17–19], some of which have been computed to all loop orders [20]. While this collection of amplitudes and integrals represents the simplest this theory has to offer, it remains suggestive that cluster algebras combinatorially realize these salient aspects of their analytic structure, thereby encoding locality in a non-obvious way.

The fact that cluster algebras appear in this context is not totally surprising, given that the plabic graphs that describe the integrands of this theory to all loop orders are themselves dual to cluster algebras [21]. In particular, the boundaries of the positive Grassmannian, where it is known that these integrands can develop physical singularities, all lie on the vanishing loci of cluster coordinates on $\text{Gr}(4, n)$ [21]. Despite this, it’s far from obvious that the location of *all* physical singularities will be picked out by cluster coordinates in this way—and indeed, certain Feynman integrals contributing to eight- and higher-particle amplitudes have recently been found to have singularities outside the positive region, at points involving square roots when expressed in terms of cluster coordinates [18, 19, 22]. This obfuscates the general connection between amplitudes and cluster algebras, as does the eventual appearance of functions beyond polylogarithms []. Both complications point to the need for more general objects than cluster algebras to describe the analytic structure of this theory at higher loops and particle multiplicities.

There is reason, however, to be optimistic that an analogously simple characterization of this (more complicated) analytic structure might be found. This optimism stems from the observation that the infinite class of amplitudes we currently have access to—the two-loop MHV amplitudes—have properties beyond branch cuts that seem to be indelibly tied to cluster algebras. In particular, the ‘nonclassical’ part of each of these amplitudes (that is, the part that cannot be expressed in terms of classical polylogarithms) is uniquely determined by a small set of physical and cluster-algebraic properties [11]. Seemingly unrelated, a pair of functions can be associated with the simplest cluster algebras, related to the Dynkin diagrams A_2 and A_3 , in terms of which these nonclassical components can be decomposed into a sum over the A_2 or A_3 subalgebras of $\text{Gr}(4, n)$ [10]. Furthermore, the remaining ‘classical’ part of these amplitudes can always be written as products of classical polylogarithms involving only

negative cluster coordinates as arguments [7]. It can be hoped that the pervasiveness of such cluster-algebraic structure points to the existence of a deeper and more general combinatorial structure. If so, better understanding the many ways in which cluster algebras appear in these amplitudes can help us identify the features this structure must have.

The motivation for this work is to review the connections already mentioned between cluster algebras, as well as to further mine the case of two-loop MHV amplitudes for additional cluster algebraic connections. In particular, in this paper we elucidate the ways in which the subalgebra structure of cluster algebras can be used to build progressively more complicated polylogarithm functions. A central ingredient in this story is a proper understanding of the automorphism groups of cluster algebras, and the ways in which those automorphisms significantly restrain the space of “cluster polylogarithms,” or polylogarithm functions which *depend* on a cluster algebra (in a sense we will define precisely). Restraining our explorations to the case of finite cluster algebras, we systematically explore the space of cluster polylogarithms and show how they can be constructed to mirror the subalgebra structure of the attendant cluster algebra, an approach we term “cluster subalgebra-constructibility”. We then turn our attention to the two-loop seven-point MHV amplitude, itself a cluster polylogarithm associated with the Grassmannian $\text{Gr}(4, 7)$, and discuss ways in which it is subalgebra-constructible. The central result is a novel representation of $R_7^{(2)}$ as a sum over essentially all subalgebras of $\text{Gr}(4, 7)$.

In a forthcoming companion paper, we apply this technology to construct the full eight-point remainder function in terms of cluster polylogarithms. Specifically, we show that the nonclassical part of this function can be decomposed as a sum over a finite subset of the A_5 subalgebras of $\text{Gr}(4, 8)$. We there describe ways in which one can search for such decompositions of this nature despite the fact that $\text{Gr}(4, n)$ gives rise to an infinite number of cluster coordinates for $n > 7$. We also show that there exists a ‘minimal BDS-like ansatz’ that preserves all Steinmann and cluster adjacency relations.¹

This paper is organized as follows. In section 2 we provide a self-contained introduction to cluster algebras and the principle ways in which they have appeared in $\mathcal{N} = 4$ SYM theory. While this section will largely constitute review for those familiar with recent developments at the intersection of these topics, section 2.6 provides a new (to the physics literature) cataloguing of the automorphism generators for finite cluster algebras. In section 3 we discuss some of the tools relevant for working with polylogarithms, particularly their associated coproduct and cobracket. We then recap the ways in which the coproduct and cobracket of two-loop amplitudes can be seen to exhibit curious cluster-algebraic properties. This section concludes with an formal definition of the term *cluster polylogarithm*. Having discussed the broad ways in which polylogarithms can be connected to cluster algebras, we show how this connection can be enhanced through the notion of *subalgebra-constructibility*, which is defined in section 4. Using the framework of subalgebra-constructibility we catalog subalgebra-

¹There exists a unique DCI BDS-like ansatz for all n that aren’t a multiple of four, but obscured by the nonexistence of a natural BDS-like normalization that preserves all Steinmann relations as exists for particle multiplicities that are not a multiple of four ...

constructible cluster polylogarithms for a wide array of finite cluster algebras. Finally, in section 5 we systematically explore the ways in which the functions of the previous section can be used to the two-loop seven-point MHV amplitude. In particular, we show that there exists a single representation of the amplitude which makes manifest subalgebra structure at every co-rank of $\text{Gr}(4, 7)$, specifically subalgebras of type D_5 , A_5 , A_4 , A_3 , and A_2 .

We also include three appendices. Appendix A walks through the explicit construction of integrable and cluster-adjacent A_2 symbols through weight 4, illustrating how symbols can be constructed on any cluster algebra. Appendix B tabulates the subalgebras of each type that appear in the finite cluster algebras relevant to $R_7^{(2)}$, while appendix C tabulates the number of independent nonclassical degrees of freedom in each of these finite cluster algebras.

2 A Brief Introduction to Cluster Algebras

Cluster algebras were first introduced by Fomin and Zelevinsky [23] as a tool for identifying which algebraic varieties come equipped with a natural notion of positivity, and what quantities determine this positivity. As a consequence, they appear in the study of the positive Grassmannian $\text{Gr}_+(k, n)$, i.e. the space of $k \times n$ matrices where all ordered $k \times k$ minors are positive. This means they also appear in the study of planar $\mathcal{N} = 4$ SYM theory, where the integrands of amplitudes are encoded to all orders by $\text{Gr}_+(4, n)$ [21].

A simple example of the type of questions cluster algebras help address is: how many matrix minors are sufficient to specify a point in $\text{Gr}_+(k, n)$? In other words, given a $k \times n$ matrix, how many of its minors do we have to calculate to know if it is in $\text{Gr}_+(k, n)$? These minors are not all independent; they satisfy identities known as Plücker relations,

$$\langle abI \rangle \langle cdI \rangle = \langle acI \rangle \langle bdI \rangle + \langle adI \rangle \langle bcI \rangle, \quad (2.1)$$

where each Plücker coordinate $\langle i_1, \dots, i_k \rangle$ corresponds to the minor of columns i_1, \dots, i_k , and I is a multi-index object with $k - 2$ entries.

To gain some intuition for this problem, let us explore the case of $\text{Gr}_+(2, 5)$. The five cyclically adjacent minors $\langle 12 \rangle$, $\langle 23 \rangle$, $\langle 34 \rangle$, $\langle 45 \rangle$, and $\langle 15 \rangle$ cannot be eliminated in terms of each other by Plücker relations, so each gives rise to an independent positivity constraint. However, making use of the Plücker relations

$$\begin{aligned} \langle 24 \rangle &= (\langle 12 \rangle \langle 34 \rangle + \langle 23 \rangle \langle 14 \rangle) / \langle 13 \rangle, \\ \langle 25 \rangle &= (\langle 12 \rangle \langle 45 \rangle + \langle 24 \rangle \langle 15 \rangle) / \langle 14 \rangle, \\ \langle 35 \rangle &= (\langle 25 \rangle \langle 34 \rangle + \langle 23 \rangle \langle 45 \rangle) / \langle 24 \rangle, \end{aligned} \quad (2.2)$$

we can eliminate three of the nonadjacent minors—for instance, $\langle 24 \rangle$, $\langle 25 \rangle$, and $\langle 35 \rangle$ —in terms of the remaining $\binom{5}{2} - 3 = 7$ adjacent and nonadjacent ones. It can be checked that all further Plücker relations are implied by those in (2.2), telling us that seven minors must be computed to determine if a matrix is in $\text{Gr}_+(2, 5)$. However, as should be clear, it is not sufficient to check the positivity of *any* seven (ordered) minors; only certain triples of minors

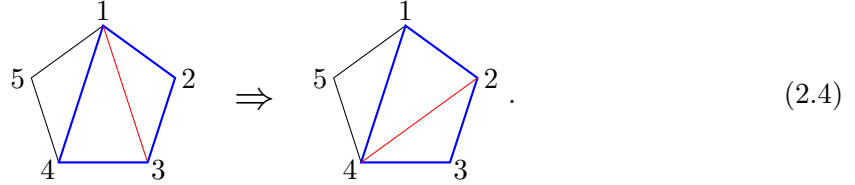
can be eliminated. It would therefore be advantageous to have a method for generating all sets of minors that are sufficient to answer this question.

To motivate how cluster algebras address this problem, consider the following triangulation of the pentagon:



We can associate the line connecting points i and j with the Plücker coordinate $\langle ij \rangle$; if we further assign these lines length $\langle ij \rangle$, the resulting pentagon is cyclic (in the sense that all its vertices reside on a common circle) due to Ptolemy's theorem. Conversely, all cyclic n -gons represent a point in $\text{Gr}_+(2, n)$ [?]. Note that the length of the three diagonals that are not present in this triangulation are determined by the length of the seven lines that are present (including edges); the problem has been reduced to geometry. This makes clear why these three diagonals—the three eliminated above—are redundant for the purpose of determining whether a matrix is in $\text{Gr}_+(2, 5)$.

From the first relation in (2.2) we see that we could have instead chosen to check the positivity of $\langle 24 \rangle$ rather than $\langle 13 \rangle$. This corresponds to choosing a different triangulation, which we get by trading the latter diagonal for the former:



We have highlighted in blue the fact that both diagonals are framed by the same quadrilateral face. More generally, we can pick any quadrilateral face and flip the diagonal it contains to generate a different triangulation. Repeatedly performing these flips generates all possible triangulations of the pentagon, as can be seen in figure 1. Each triangulation provides a set of edges/minors whose positivity ensures that a matrix is in $\text{Gr}_+(2, 5)$.

In an analogous way, cluster algebras answer questions about positivity for a larger class of algebraic varieties (and in particular for all $\text{Gr}_+(k, n)$) by considering *clusters* that can all be generated by an operation called *mutation*, just as all triangulations of the pentagon are generated by flipping the diagonals of quadrilateral faces. We now turn to the definition of these objects, by first considering how the example of $\text{Gr}_+(2, 5)$ can be rephrased in terms of cluster algebras.

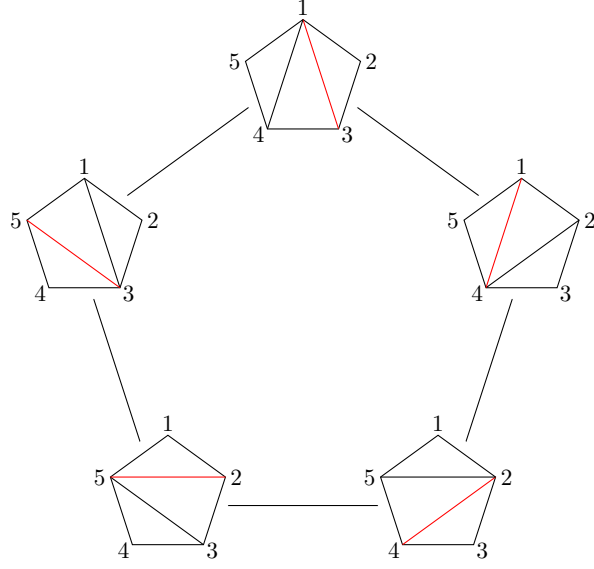


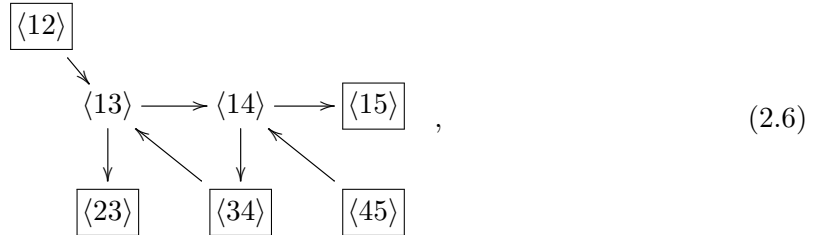
Figure 1: Mutating on the red chord moves you clockwise around the figure.

2.1 Clusters, Mutations, and Cluster \mathcal{A} -coordinates

Clusters can be thought of as quiver diagrams—namely, oriented graphs equipped with arrows connecting different nodes—in which each node is assigned a cluster coordinate.² We can form a cluster of $\text{Gr}(2, 5)$ out of our original triangulation (2.3) by assigning an orientation to the pentagon and all subtriangles, such as



The nodes of our cluster are then given by the lines of this triangulation (making the minors $\langle ab \rangle$ our cluster coordinates), where an arrow is assigned from $\langle ab \rangle$ to $\langle cd \rangle$ if the triangle orientations in (2.5) have segment (ab) flowing into segment (cd) . This gives us the quiver



²Here and throughout, we are implicitly restricting ourselves to skew-symmetric cluster algebras of geometric type [1].

where the boxes around the $\langle ii + 1 \rangle$ indicate that they are *frozen*—they can never change because they aren't in the interior of a quadrilateral face. The variables living at the frozen nodes can be thought of as parameterizing the boundary of our space, while the mutable nodes represent parameterizations of the interior. One can also draw arrows (with partial weight) connecting these frozen nodes (see, for instance, [21]), but we will not keep track of them here.

We have now drawn our first cluster (also sometimes called a seed). The Plücker coordinates in this cluster are referred to as cluster \mathcal{A} -coordinates (sometimes also \mathcal{A} -variables), and they come in two flavors: mutable (for example, $\langle 13 \rangle$ and $\langle 14 \rangle$ above) and frozen ($\langle ii + 1 \rangle$ above). In any quiver, the information encoded by the arrows can also be represented in terms of a skew-symmetric matrix

$$b_{ij} = (\# \text{ of arrows } i \rightarrow j) - (\# \text{ of arrows } j \rightarrow i) \quad (2.7)$$

called the exchange matrix (or the signed adjacency matrix).

The process of mutation, which we have described geometrically in terms of flipping diagonals, has a simple interpretation at the level of quivers. Given a quiver such as (2.6), we can choose any mutable node k to mutate on (this is equivalent to picking which diagonal to flip). Mutation gives us back a new quiver in which the \mathcal{A} -coordinate a_k has been sent to a'_k , where

$$a_k a'_k = \prod_{i|b_{ik}>0} a_i^{b_{ik}} + \prod_{i|b_{ik}<0} a_i^{-b_{ik}}, \quad (2.8)$$

(with the understanding that an empty product is set to one), while all other cluster \mathcal{A} -coordinates remain unchanged. The arrows connecting the nodes in this new quiver are also modified by carrying out the following operation:

- for each path $i \rightarrow k \rightarrow j$, add an arrow $i \rightarrow j$,
- reverse all arrows on the edges incident with k ,
- and remove any two-cycles (oppositely-oriented arrows) that may have formed.

This creates a new adjacency matrix b'_{ij} via

$$b'_{ij} = \begin{cases} -b_{ij}, & \text{if } k \in \{i, j\}, \\ b_{ij}, & \text{if } b_{ik}b_{kj} \leq 0, \\ b_{ij} + b_{ik}b_{kj}, & \text{if } b_{ik}, b_{kj} > 0, \\ b_{ij} - b_{ik}b_{kj}, & \text{if } b_{ik}, b_{kj} < 0. \end{cases} \quad (2.9)$$

Mutation is an involution, so mutating on a'_k will take you back to the original cluster (just as flipping the same diagonal twice will take you back to where you started).

In terms of these ingredients, a cluster algebra can be defined to be a set of clusters that is closed under mutation. Thus, mutating on any non-frozen node of any cluster will generate

a different cluster in the same cluster algebra. In practice, one therefore constructs cluster algebras by starting from a seed such as (2.6), and iteratively mutating on all available nodes until the set of clusters closes (or it becomes clear the cluster algebra is infinite).

It is common to refer to certain cluster algebras by particularly nice representative clusters, when the mutable nodes of the corresponding quiver form an oriented Dynkin diagram. For instance, the $\text{Gr}(2,5)$ cluster algebra is often referred to as A_2 , since the mutable part of the seed (2.6) is given by $\langle 13 \rangle \rightarrow \langle 14 \rangle$. Thus, we will often speak interchangeably of the cluster algebras for $\text{Gr}(2,5)$ and A_2 . This is a slight abuse of notation, as the $\text{Gr}(2,5)$ cluster algebra corresponds specifically to the cluster algebra generated by the collection of frozen and mutable nodes in eq. (2.6), whereas an A_2 cluster algebra can in principle be dressed with any number of frozen nodes. We will see why this language is useful in the next section.

2.2 Cluster \mathcal{X} -coordinates

Cluster algebras can also be formulated in terms of a different set of cluster coordinates, called Fock-Goncharov coordinates or \mathcal{X} -coordinates [24]. As we will see in future sections, cluster \mathcal{X} -coordinates play a crucial role in connecting cluster algebras to polylogarithms and scattering amplitudes. While it is always possible to phrase results involving cluster algebras directly in terms of \mathcal{A} -coordinates, \mathcal{X} -coordinates often allow for cluster-algebraic structure to be made more manifest.

Clusters formed out of \mathcal{X} -coordinates can be directly constructed out of clusters involving \mathcal{A} -coordinates. Given a quiver equipped with \mathcal{A} -coordinates and described by the exchange matrix b_{ij} , we can compute an \mathcal{X} -coordinate to assign to each mutable node by

$$x_i = \prod_j a_j^{b_{ij}}. \quad (2.10)$$

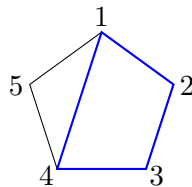
For example, the \mathcal{X} -coordinate cluster associated with (2.6) is formed by associating cluster \mathcal{X} -coordinates with all its mutable nodes, where these \mathcal{X} -coordinates are constructed by putting all Plücker coordinates that point to that node in the denominator, and all Plücker coordinates that are pointed to by that node in the numerator. That is, we get

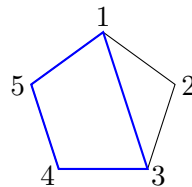
$$\frac{\langle 14 \rangle \langle 23 \rangle}{\langle 12 \rangle \langle 34 \rangle} \rightarrow \frac{\langle 15 \rangle \langle 34 \rangle}{\langle 13 \rangle \langle 45 \rangle}, \quad (2.11)$$

which takes the form of the generic A_2 quiver

$$x_1 \rightarrow x_2. \quad (2.12)$$

In the pentagon-triangulation picture, these \mathcal{X} -coordinates describe overlapping quadrilaterals, for instance


 $\sim \frac{\langle 14 \rangle \langle 23 \rangle}{\langle 12 \rangle \langle 34 \rangle},$


 $\sim \frac{\langle 15 \rangle \langle 34 \rangle}{\langle 13 \rangle \langle 45 \rangle},$

(2.13)

which come in one-to-one correspondence with the diagonals in a triangulation.

Mutation rules for \mathcal{X} -coordinates are different than for \mathcal{A} -coordinates, and are given by

$$x'_i = \begin{cases} x_k^{-1}, & i = k, \\ x_i(1 + x_k^{\text{sgn } b_{ik}})^{b_{ik}}, & i \neq k \end{cases}. \quad (2.14)$$

Mutation still changes the arrows in the quiver diagram as it did in the case of \mathcal{A} -coordinates. Given just a quiver diagram, it can sometimes be unclear whether a given quiver should be mutated using the \mathcal{A} -coordinate or \mathcal{X} -coordinate rules (2.8) or (2.14). We adopt the convention in this work that if a quiver is given with no frozen nodes, it should be thought of as equipped with \mathcal{X} -coordinates.

Just as with \mathcal{A} -coordinate clusters, we can generate all \mathcal{X} -coordinate clusters by mutation. Mutating on alternating nodes of our A_2 cluster (2.12) (starting with x_2), we generate the following sequence of clusters:

$$\begin{aligned} x_1 &\rightarrow x_2 \\ x_1(1+x_2) &\leftarrow \frac{1}{x_2} \\ \frac{1}{x_1(1+x_2)} &\rightarrow \frac{x_2}{1+x_1+x_1x_2} \\ \frac{x_1x_2}{1+x_1} &\leftarrow \frac{1+x_1+x_1x_2}{x_2} \\ \frac{1+x_1}{x_1x_2} &\rightarrow \frac{1}{x_1} \\ x_2 &\leftarrow x_1 \\ &\vdots \end{aligned} \quad (2.15)$$

The series then repeats, with all arrows reversed. Note that (specifically in the case of A_2), if we label these \mathcal{X} -coordinates by

$$\mathcal{X}_1 = 1/x_1, \quad \mathcal{X}_2 = x_2, \quad \mathcal{X}_3 = x_1(1+x_2), \quad \mathcal{X}_4 = \frac{1+x_1+x_1x_2}{x_2}, \quad \mathcal{X}_5 = \frac{1+x_1}{x_1x_2}, \quad (2.16)$$

the mutation rule in (2.14) takes the simple form

$$1 + \mathcal{X}_i = \mathcal{X}_{i-1}\mathcal{X}_{i+1}, \quad (2.17)$$

while all the clusters take the form $1/\mathcal{X}_i \rightarrow \mathcal{X}_{i+1}$. Eq. (2.17) is commonly referred to as the A_2 exchange relation. Putting this all together, we will generically refer to an A_2 cluster algebra as any set of clusters $1/\mathcal{X}_{i-1} \rightarrow \mathcal{X}_i$ for $i = 1 \dots 5$ where the \mathcal{X}_i satisfy eq. (2.17). We believe it is useful at this point to emphasize that one can take as input any x_1 and x_2 , and generate an associated A_2 .

A very useful feature of cluster \mathcal{X} -coordinates is that they come equipped with a natural Poisson bracket structure, making the Grassmannian $\text{Gr}(k, n)$ a cluster Poisson variety [].

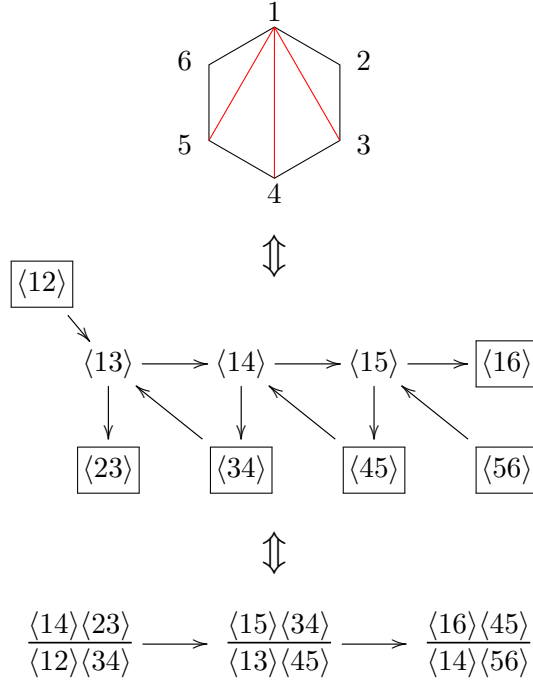


Figure 2: One triangulation of the hexagon, and its associated \mathcal{A} -coordinate and \mathcal{X} -coordinate seed quivers.

Namely, when two \mathcal{X} -coordinates appear together in a cluster of $\text{Gr}(k, n)$, there exists a Poisson bracket that evaluates to

$$\{x_i, x_j\} = b_{ij} x_i x_j . \quad (2.18)$$

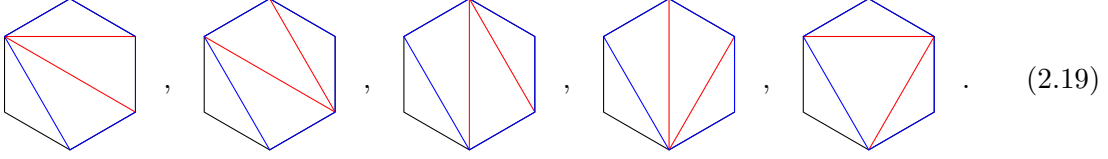
This structure respects mutation, implying that the entry b_{ij} (which counts the number of arrows from x_i to x_j in a given cluster's quiver) will be the same in all clusters containing both x_i and x_j . The Poisson bracket (and associated Sklyanin bracket) will play a larger role in a forthcoming companion paper [?], so we defer further discussion of this structure to there (for existing discussions in the literature, see also [? ?]).

2.3 Subalgebras and Associahedra

Cluster algebras contain a rich and intricate subalgebra structure, which will play a central role in our analysis. It is simple to illustrate how these subalgebras arise by considering $\text{Gr}(2, 6)$, which triangulates the hexagon. In figure 2 we give the seed cluster for $\text{Gr}(2, 6)$ in the triangulation, \mathcal{A} -coordinate, and \mathcal{X} -coordinate representations, respectively. Since the mutable nodes take the form of an A_3 Dynkin diagram, we often speak of $\text{Gr}(2, 6)$ and A_3 interchangeably, just as we did with $\text{Gr}(2, 5) \simeq A_2$.

The $\text{Gr}(2, 6)$ cluster algebra features 14 clusters, which can be grouped into multiple (overlapping) subalgebras. A simple example is the collection of all triangulations which

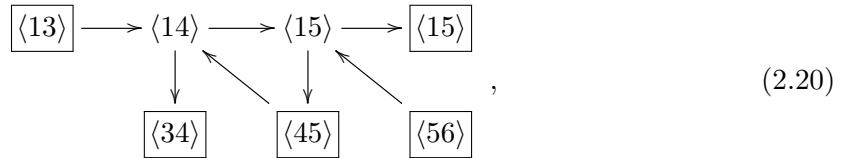
involve the chord $\langle 15 \rangle$. This set contains 5 clusters and is itself a cluster algebra, which can be generated by treating $\langle 15 \rangle$ as a frozen node (or in \mathcal{X} -coordinates, freezing the node $\frac{\langle 14 \rangle \langle 56 \rangle}{\langle 16 \rangle \langle 45 \rangle}$). This of course is the cluster algebra corresponding to the triangulations of the pentagon formed by points $1, \dots, 5$, outlined here in blue:



We therefore refer to the collection of clusters that involve this pentagon as an A_2 subalgebra of $\text{Gr}(2, 6)$.

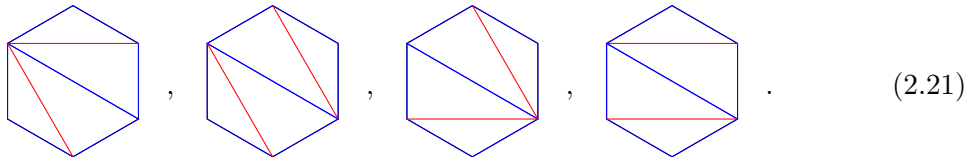
It should be clear, upon referring back to the mutation rules (2.8) and (2.14), that this A_2 subalgebra is truly identical to what we have been calling $\text{Gr}(2, 5)$. That is, neither the \mathcal{A} - or \mathcal{X} -coordinate mutation rule (or the rule for constructing the \mathcal{X} -coordinate cluster out of the \mathcal{A} -coordinate one) depends on nodes further than a single arrow away from the node on which one is mutating. Correspondingly, this subalgebra doesn't know about the existence of nodes involving point/column 6. (In the \mathcal{X} -coordinate case, the coordinate associated with the newly frozen node will change when one mutates on the node it is connected to, but the presence of this frozen node does not effect the coordinates appearing in the A_2 subalgebra itself.) We consider two subalgebras to be identical when the clusters they appear in only differ by nodes that have no effect on the mutable nodes of the subalgebra.

What if we instead disallow mutation on the chord $\langle 13 \rangle$ (and the corresponding \mathcal{X} -coordinate node $\frac{\langle 12 \rangle \langle 34 \rangle}{\langle 13 \rangle \langle 24 \rangle}$)? Dropping the nodes that play no role in any of the mutations that remain gives rise to the effective quiver



where we have put a box around $\langle 13 \rangle$ to make clear we are now treating it as frozen. We have also dropped the arrow from $\langle 34 \rangle$ to $\langle 13 \rangle$ since we are ignoring arrows between frozen nodes. The comparison to (2.6) should be clear; this just represents a re-labeled version of $\text{Gr}(2, 5)$.

Similarly, if we disallow mutation on the chord $\langle 14 \rangle$ (and $\frac{\langle 13 \rangle \langle 45 \rangle}{\langle 15 \rangle \langle 34 \rangle}$), we generate an $A_1 \times A_1$ subalgebra, since the chord $\langle 14 \rangle$ divides the hexagon in to two non-overlapping squares, each of which are triangulated by A_1 (or really $\text{Gr}(2, 4)$):



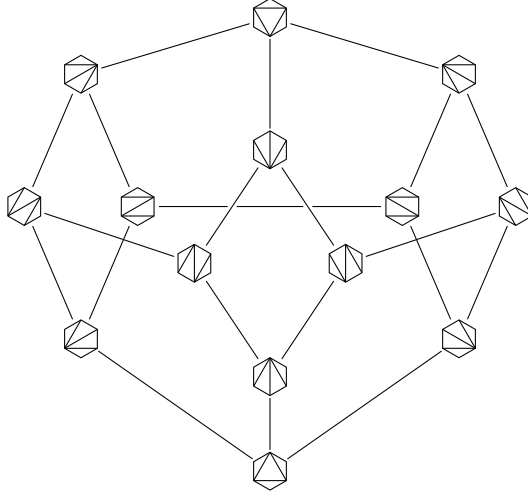


Figure 3: The associahedron for $A_3 \simeq \text{Gr}(2, 6)$, where each cluster is represented by a triangulation of the hexagon.

In appendix B we have tabulated the number of such A_2 and $A_1 \times A_1$ subalgebras in A_3 , as well as the subalgebras of other cluster algebras that are relevant to seven-particle scattering in planar $\mathcal{N} = 4$. There it will be found that there are in fact six A_2 subalgebras and three $A_1 \times A_1$ subalgebras of A_3 (the remaining subalgebras do not involve the cluster in figure 2).

This subalgebra structure can be nicely visualized by constructing an object known as the associahedron (or the Stasheff polytope) of a given cluster algebra. The vertices of this polytope each represent a cluster, while its edges represent the mutations that map these clusters into each other. For instance, figure 1 corresponds to the $\text{Gr}(2, 5) \simeq A_2$ associahedron, which also coincidentally takes the form of a pentagon. Note that every node has valency two since each cluster has two mutable vertices.

Similarly, the associahedron of $\text{Gr}(2, 6) \simeq A_3$ is given in figure 3. It has 14 vertices, corresponding to the 14 clusters of $\text{Gr}(2, 6)$, each with valency three. These vertices assemble into three square faces and six pentagonal faces—corresponding exactly to the three $A_1 \times A_1$ subalgebras and the six A_2 subalgebras of A_3 . This makes it easy to read off the subalgebra structure of A_3 directly.

In A_3 , it turns out that each of the faces corresponds to a distinct subalgebra. Associahedra for larger cluster algebras become quite complicated, and it is often the case that distinct faces (or higher-dimensional polytopes) correspond to identical subalgebras. As an example, the associahedron of $\text{Gr}(4, 7) \simeq E_6$, which will be the focus of much of the rest of this paper, is shown in figure 4. It has 833 vertices, each of valence 6, and 1071 pentagons corresponding to A_2 subalgebras; however, only 504 of these A_2 subalgebras are distinct.

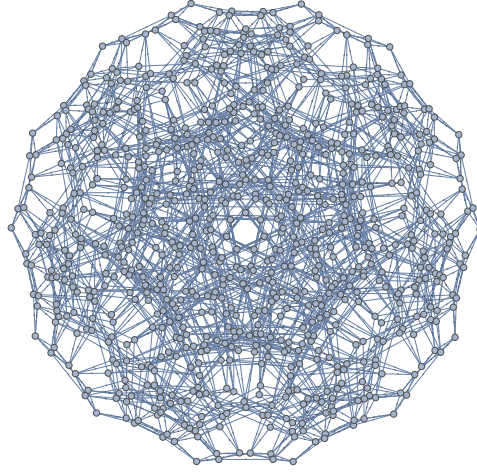


Figure 4: The associahedron for $E_6 \simeq \text{Gr}(4, 7)$.

2.4 Grassmannian Cluster Algebras and Planar $\mathcal{N} = 4$ SYM

So far we have leaned heavily on the correspondence between the triangulations of an n -gon and the cluster algebra for $\text{Gr}(2, n)$. Based on the examples of $\text{Gr}(2, 5)$ and $\text{Gr}(2, 6)$, it is not hard to write down a generic seed cluster for $\text{Gr}(2, n)$ corresponding to the triangulation consisting of all chords $\langle 13 \rangle, \dots, \langle 1n-1 \rangle$:

$$\begin{array}{c}
 \begin{array}{c} 1 \\ \diagup \quad \diagdown \\ n \quad \quad 2 \\ \diagdown \quad \diagup \\ n-1 \quad 3 \end{array} \Leftrightarrow \begin{array}{c} \boxed{\langle 12 \rangle} \\ \searrow \\ \langle 13 \rangle \longrightarrow \dots \longrightarrow \langle 1 \ n-1 \rangle \longrightarrow \boxed{\langle 1n \rangle} \\ \downarrow \quad \quad \quad \downarrow \quad \quad \quad \downarrow \\ \boxed{\langle 23 \rangle} \quad \quad \quad \dots \quad \quad \quad \boxed{\langle n-2 \ n-1 \rangle} \quad \boxed{\langle n-1 \ n \rangle} \end{array} . \quad (2.22)
 \end{array}$$

Here one sees that $\text{Gr}(2, n) \simeq A_{n-3}$.

For $\text{Gr}(k > 2, n)$, there is no longer a simple connection with triangulations or Dynkin diagrams. However, as shown by Scott [25] there exists a generalization of eq. (2.22) valid for all $\text{Gr}(k, n)$:

$$\begin{array}{c}
 \boxed{f_{11}} \leftarrow f_{12} \leftarrow f_{13} \leftarrow \dots \leftarrow f_{1l} \leftarrow \boxed{\langle 1, \dots, k \rangle} \\
 \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \\
 \boxed{f_{21}} \leftarrow f_{22} \leftarrow f_{23} \leftarrow \dots \leftarrow f_{2l} \\
 \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \\
 \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\
 \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \quad \nearrow \quad \downarrow \\
 \boxed{f_{k1}} \quad \boxed{f_{k2}} \quad \boxed{f_{k3}} \quad \dots \quad \boxed{f_{kl}}
 \end{array} , \quad (2.23)$$

where $l = n - k$ and

$$f_{ij} = \begin{cases} \langle i+1, \dots, k, k+j, \dots, i+j+k-1 \rangle, & i \leq l-j+1, \\ \langle 1, \dots, i+j-l-1, i+1, \dots, k, k+j, \dots, n \rangle, & i > l-j+1. \end{cases} \quad (2.24)$$

(Note that evaluating the above expression for $k = 2$ will give a rotated version of (2.22) and the instances of $\text{Gr}(2, n)$ in the previous sections.) The cluster algebra on $\text{Gr}(k, n)$ is therefore of rank $(n - k - 1)(k - 1)$, i.e. the number of mutable nodes in (2.23).

The cluster algebra on $\text{Gr}(4, n)$ naturally appears in planar $\mathcal{N} = 4$ SYM theory, where it parametrizes the space of n -particle kinematics. To make this connection, one first defines a set of dual coordinates x_i in terms of the external momenta p_i (which are endowed with a natural ordering in the planar limit) by

$$p_i^{\alpha\dot{\alpha}} = \lambda_i^\alpha \tilde{\lambda}_i^{\dot{\alpha}} = x_i^{\alpha\dot{\alpha}} - x_{i+1}^{\alpha\dot{\alpha}}, \quad (2.25)$$

where $x_{n+1}^{\alpha\dot{\alpha}} \equiv x_1^{\alpha\dot{\alpha}}$. These coordinates describe the cusps of a light-like Wilson loop dual to the amplitude; as a result, the original amplitude respects an additional superconformal symmetry that is associated with these dual coordinates (up to an anomaly associated with the cusps of the Wilson loop, which is accounted for by the BDS ansatz) [26–34]. In terms of the quantities in (2.25), we can further define momentum twistors Z_i

$$Z_i^R = (\lambda_i^\alpha, x_i^{\beta\dot{\alpha}} \lambda_{i\beta}), \quad (2.26)$$

where $R = (\alpha, \dot{\alpha})$ is an $SU(2, 2)$ index. Momentum twistors are invariant under the little group, which acts as an overall rescaling $Z_i^R \rightarrow t_i Z_i^R$, and as such represent points in \mathbb{CP}^3 .

If we assemble these momentum twistors into a $4 \times n$ matrix in which the i^{th} column corresponds to the four $SU(2, 2)$ components of Z_i^R , invariance under the dual conformal group becomes invariance under $\text{SL}(4)$. The overall rescaling symmetry of one of the momentum twistors can be combined with this $\text{SL}(4)$ invariance to identify this matrix as a point in the (not necessarily positive) Grassmannian $\text{Gr}(4, n)$, modulo the rescaling invariance of the remaining $n - 1$ columns. Thus, the kinematic data of an n -point scattering process is encoded in a momentum twistor matrix

$$Z_n \in \text{Gr}(4, n)/\text{GL}(1)^{n-1}. \quad (2.27)$$

For more details regarding this correspondence, see [8, 21].

To relate the dual-conformal invariants encoded in Z_n to more familiar kinematic quantities, we can translate the (cyclically ordered) Mandelstam invariants into squared differences of dual coordinates,

$$s_{i, \dots, j-1} \equiv (p_i + \dots p_{j-1})^2 = \det(x_i^{\alpha\dot{\alpha}} - x_j^{\alpha\dot{\alpha}}) \equiv x_{ij}^2. \quad (2.28)$$

Dual conformal invariants can be constructed out of these objects by putting together combinations that are invariant under the dual conformal inversion generator, which acts on these

coordinates as

$$I(x_i^{\alpha\dot{\alpha}}) = \frac{x_i^{\alpha\dot{\alpha}}}{x_i^2}, \quad I(x_{ij}^2) = \frac{x_{ij}^2}{x_i^2 x_j^2}. \quad (2.29)$$

Thus, (regulated) amplitudes in this theory depend only on ratios of squared differences in which the same dual indices appear in both the numerator and denominator. The quantities x_{ij}^2 can be translated into momentum twistors using the relation

$$x_{ij}^2 = \frac{\det(Z_{i-1}Z_iZ_{j-1}Z_j)}{(\epsilon_{\alpha\beta}\lambda_{i-1}^\alpha\lambda_i^\beta)(\epsilon_{\gamma\delta}\lambda_{j-1}^\gamma\lambda_j^\delta)}, \quad (2.30)$$

where $\epsilon_{\alpha\beta}$ is the Levi-Civita tensor. In dual-conformally invariant quantities, the spinor products $\epsilon_{\alpha\beta}\lambda_{i-1}^\alpha\lambda_i^\beta$ all cancel, leaving only determinants of four-tuples of momentum twistors. These are just minors of the momentum twistor matrix (2.27), which we recognize as the cluster \mathcal{A} -coordinates

$$\langle ijkl \rangle = \det(Z_iZ_jZ_kZ_l). \quad (2.31)$$

Note that the two-particle Mandelstams $s_{i,i+1}$ correspond to the frozen nodes of (2.23), while higher-particle Mandelstams and more general (polynomials of) Plücker coordinates can appear as mutable nodes.

By construction, the \mathcal{X} -coordinates on $\text{Gr}(4, n)$ derived from the seed (2.23) respect dual-conformal invariance. Both mutation rules (2.8) and (2.14) preserve this property (and commute with the translation (2.10)), ensuring that all \mathcal{X} -coordinates are dual conformal invariants. Such invariants cannot be formed in four- or five-particle kinematics, due to an insufficient number of non-lightlike separated points (since we are in massless kinematics, $x_{ii+1}^2 = 0$ for all i). This fact shows up in the seed (2.23) as $\text{Gr}(4, n < 6)$ having no mutable nodes (and therefore no \mathcal{X} -coordinates). For $n > 5$, there are $3(n - 5)$ mutable nodes in $\text{Gr}(4, n)$, matching the number of algebraically independent dual conformal invariants that can be formed out of n massless particles [].

The \mathcal{A} -coordinate and \mathcal{X} -coordinate seed clusters of the first nontrivial example, $\text{Gr}(4, 6)$, are shown in figure 5. As discussed above, the three \mathcal{X} -coordinates in this cluster furnish us with a chart that covers the space of (dual-conformally invariant) six-particle kinematics. Moreover, we can generate new charts by mutation—every \mathcal{X} -coordinate cluster of $\text{Gr}(4, n)$ provides a valid chart for n -particle kinematics. As explored in great depth in [21], these charts are especially well suited to describing the boundaries of the positive Grassmannian $\text{Gr}_+(4, n)$, where the integrands of n -particle amplitudes can develop physical singularities. In particular, every such boundary occurs at the vanishing locus of an \mathcal{A} -coordinate of $\text{Gr}(4, n)$, which implies it also occurs at the vanishing locus of some set of \mathcal{X} -coordinates.

This fact is especially propitious for loop-level amplitudes (and integrals) that only have branch points on the boundaries of the positive Grassmannian. In such cases, the symbol alphabet encoding the polylogarithmic part of these amplitudes is naturally given in terms of cluster coordinates. We defer discussion of the coaction and symbol alphabets to section 3.1, but here note that it is multiplicative independence, rather than algebraic independence,

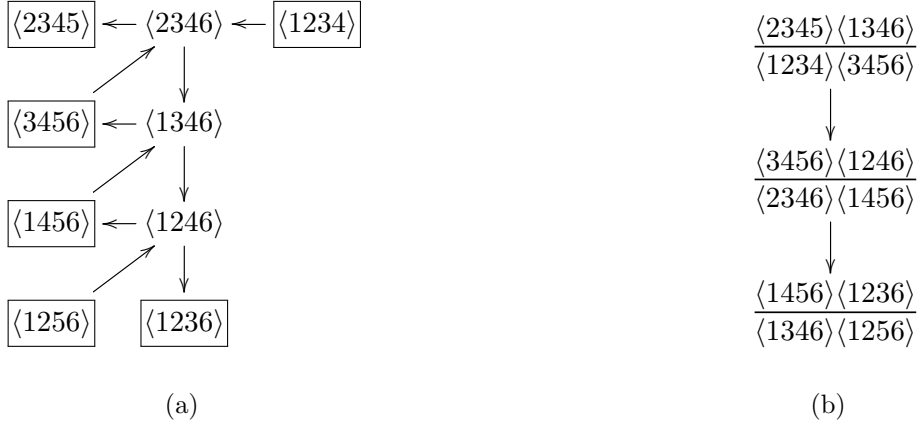


Figure 5: The \mathcal{A} -coordinate seed quiver (a) and \mathcal{X} -coordinate seed quiver (b) for $\text{Gr}(4, 6)$.

that is relevant in the context of symbol alphabets. Thus, while it is not possible to realize all boundaries of the positive Grassmannian as the vanishing loci of either type of cluster coordinate in a single chart [21], all boundaries are exposed as the vanishing of some symbol letter if cluster \mathcal{A} -coordinates or \mathcal{X} -coordinates on $\text{Gr}(4, n)$ are adopted as a symbol alphabet.

While amplitudes in planar $\mathcal{N} = 4$ are not generically expected to have this property (and indeed, certain Feynman integrals have been computed that do not [18, 19]), an infinite class of amplitudes do—namely, all two-loop MHV amplitudes [6], and all six- and seven-particle amplitudes computed to date [3–5, 14–16]. The significance of this property is illustrated by the two-loop, six-particle remainder function, which encodes the MHV amplitude. Namely, this function can be put in the form

$$R_6^{(2)} = \sum_{\text{cyclic}} \left[\text{Li}_4 \left(-\frac{\langle 2345 \rangle \langle 1346 \rangle}{\langle 1234 \rangle \langle 3456 \rangle} \right) - \frac{1}{4} \text{Li}_4 \left(-\frac{\langle 1246 \rangle \langle 1345 \rangle}{\langle 1234 \rangle \langle 1456 \rangle} \right) \right] + \dots, \quad (2.32)$$

[**check normalization**] where the cyclic sum is over all rotations of the four-bracket indices $i \rightarrow i + j$ for $0 \leq j < 6$, and the dots indicate this equality only holds up to products of lower-weight polylogarithms. This projection is well-defined and will be introduced, along with the n -particle remainder function, in section 3. Here we just emphasize the simplicity of this expression, which takes the form of classical polylogarithms with negative \mathcal{X} -coordinate arguments. (In particular, the argument of the first polylogarithm appears in figure 5b, while the argument of the second polylogarithm appears after mutating on the top node of this cluster.) Moreover, the part of the expression we have dropped in (2.32) can also be expressed entirely in terms of products of classical polylogarithms with negative \mathcal{X} -coordinate arguments [7].

This surprising property—of being expressible as polylogarithms with negative \mathcal{X} -coordinate arguments—is enjoyed by the remainder function at all n . However, for $n > 6$ the remainder function has a nonclassical component, so generalized polylogarithms (in addition to classical polylogarithms) with negative \mathcal{X} -coordinate arguments also appear []. Although this

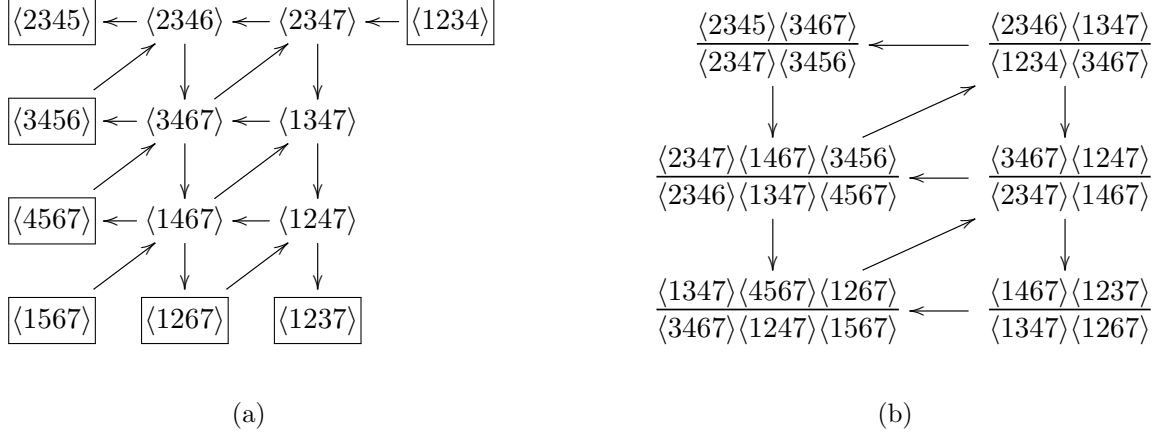


Figure 6: The \mathcal{A} -coordinate seed quiver (a) and \mathcal{X} -coordinate seed quiver (b) for $\text{Gr}(4, 7)$.

component represents the mathematically most complicated part of the remainder function, it was shown in [10] that it is decomposable into building blocks related to the A_2 and A_3 subalgebras of $\text{Gr}(4, n)$. This allows the all- n symbol computed in [6] to be systematically upgraded to a function, as was done for seven particles in [7]. In the later sections of this paper, we demonstrate the existence of further subalgebra structure in the nonclassical part of the seven-particle remainder function (we leave consideration of higher-point kinematics to future work [?]). The \mathcal{A} -coordinate and \mathcal{X} -coordinate seeds for $\text{Gr}(4, 7)$ are presented in figure 6.

Before turning to the remaining aspects of cluster algebras that we wish to develop, we note that plabic graphs—which are dual to the clusters we’ve been describing—encode a great deal more about planar $\mathcal{N} = 4$ than we have had reason to touch on. In particular, [boundaries—a subset of which are our subalgebras (?)]. We refer interested readers to the exposition of this rich structure given in [21].

2.5 Finite Cluster Algebras

The procedure of writing down an oriented quiver, dressing it with coordinates, and iteratively mutating on all non-frozen nodes using either the \mathcal{A} -coordinate or \mathcal{X} -coordinate mutation rule will always produce a cluster algebra. However, generic quivers give rise to exceedingly complicated cluster algebras—in fact, for a wide class of seeds, mutation will generate an infinite numbers of clusters. For the rest of this paper we will restrict our attention to finite cluster algebras, leaving the discussion of infinite algebras to a forthcoming companion paper [?] (see [] for discussions of infinite cluster algebras elsewhere in the literature).

Fortunately, Fomin and Zelevinsky classified all finite cluster algebras in [35]. In particular, they showed that a cluster algebra is of finite type if and only if the mutable part of at least one of its clusters takes the form of an oriented, simply-laced Dynkin diagram: A_n , D_n ,

or $E_{n \leq 8}$. As we will primarily be interested in subalgebras of the cluster algebra on $\text{Gr}(4, 7)$, we here focus on the cases where $n < 6$ (and on the case of $E_6 \simeq \text{Gr}(4, 7)$ itself).

As mentioned above, cluster algebras of type A_n can be generated by the seed

$$x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_n, \quad (2.33)$$

which corresponds to the cluster algebra on $\text{Gr}(2, n+3)$. Each of the clusters in these algebras can be thought as triangulating an $(n+3)$ -gon, where the \mathcal{A} -coordinates correspond to chords and the \mathcal{X} -coordinates to quadrilateral faces. This makes the counting easy: the number of clusters for A_n is given by the Catalan number $C(n+1)$, the number of distinct \mathcal{A} -coordinates is $\binom{n+3}{2} - n$, and the number of distinct \mathcal{X} -coordinates is $2\binom{n+3}{4}$. Any smaller polygon embedded into the $(n+3)$ -gon gives rise to a subalgebra; for example, there are $56 = \binom{8}{5}$ pentagonal embeddings in an octagon, so there are 56 A_2 subalgebras in A_5 .

Of particular interest is the cluster algebra generated by $A_3 \simeq \text{Gr}(4, 6)$, which describes six-particle scattering. By comparison with figure 5b, we see that the \mathcal{X} -coordinates in the quiver (2.33) act as coordinates on the space of momentum twistors, where they correspond to the functions

$$x_1 = \frac{\langle 2345 \rangle \langle 1346 \rangle}{\langle 1234 \rangle \langle 3456 \rangle}, \quad x_2 = \frac{\langle 3456 \rangle \langle 1246 \rangle}{\langle 2346 \rangle \langle 1456 \rangle}, \quad x_3 = \frac{\langle 1456 \rangle \langle 1236 \rangle}{\langle 1346 \rangle \langle 1256 \rangle}. \quad (2.34)$$

More generally, any \mathcal{A} -coordinate in $\text{Gr}(4, 6)$ can be expressed in terms of the variables x_1 , x_2 , and x_3 by evaluation on the momentum twistor matrix

$$Z_{A_3} = \begin{pmatrix} 1 & 1 & 0 & 0 & x_2 x_3 & 0 \\ 0 & 1 & 1+x_3 & x_3 & 0 & 0 \\ 1+x_1 & x_1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}. \quad (2.35)$$

The chief advantage of working directly in terms of cluster \mathcal{X} -coordinates such as x_1 , x_2 , and x_3 is that they trivialize all Plücker relations (2.1). Furthermore, cluster coordinates rationalize many of the square roots that appear when amplitudes and integrals are expressed in terms of dual conformally invariant cross ratios [18]. For instance, in this chart the dual conformal cross ratios commonly used to express the six-particle amplitude evaluate to

$$u = \frac{\langle 6123 \rangle \langle 3456 \rangle}{\langle 6134 \rangle \langle 2356 \rangle} = \frac{x_2 x_3}{1 + x_3 + x_2 x_3}, \quad (2.36)$$

$$v = \frac{\langle 1234 \rangle \langle 4561 \rangle}{\langle 1245 \rangle \langle 3461 \rangle} = \frac{1}{1 + x_2 + x_1 x_2}, \quad (2.37)$$

$$w = \frac{\langle 2345 \rangle \langle 5612 \rangle}{\langle 2356 \rangle \langle 4512 \rangle} = \frac{x_1 x_2}{(1 + x_2 + x_1 x_2)(1 + x_3 + x_2 x_3)}, \quad (2.38)$$

which rationalizes the square root that appears in six-particle kinematics

$$\sqrt{(1 - u - v - w)^2 - 4uvw} = \frac{x_2(1 - x_1 x_3)}{(1 + x_2 + x_1 x_2)(1 + x_3 + x_2 x_3)}. \quad (2.39)$$

Note that the cluster coordinate expressions (2.36) are rotated compared to those given elsewhere in the literature [8, 36] even though both arise from a seed of the form (2.33); this reflects a differing convention for the seed of $\text{Gr}(k, n)$.

The first nondegenerate Dynkin diagram of type D_n is D_4 , corresponding to the seed quiver

$$x_1 \rightarrow x_2 \begin{array}{l} \nearrow x_3 \\ \searrow x_4 \end{array} . \quad (2.40)$$

Note here that we no longer are treating x_1, x_2, x_3 as defined in eq. (2.34), the x_i in eq. (2.40) are all generic variables. This seed turns out to generate the same cluster algebra as $\text{Gr}(3, 6)$; in particular, starting from the seed in (2.23) and mutating on the nodes initially labeled by f_{13} and then f_{23} , one arrives at the \mathcal{X} -coordinate quiver (2.40), where

$$x_1 = \frac{\langle 234 \rangle \langle 135 \rangle}{\langle 123 \rangle \langle 345 \rangle}, \quad x_2 = \frac{\langle 125 \rangle \langle 356 \rangle}{\langle 156 \rangle \langle 235 \rangle}, \quad x_3 = \frac{\langle 156 \rangle \langle 345 \rangle}{\langle 135 \rangle \langle 456 \rangle}, \quad x_4 = \frac{\langle 123 \rangle \langle 156 \rangle}{\langle 126 \rangle \langle 135 \rangle}. \quad (2.41)$$

The corresponding momentum twistor matrix is given by

$$Z_{D_4} = \begin{pmatrix} x_1 & 1 & 0 & -1 & 0 & x_1 x_2 x_4 \\ 0 & -1 & 0 & 1 + x_1 & x_1(1 + x_4) & x_1(1 + x_4) \\ x_1 x_3 & x_3(1 + x_2) & 1 & 1 & 0 & 0 \end{pmatrix}. \quad (2.42)$$

Conversely, the cluster algebra generated by D_5 ,

$$x_1 \rightarrow x_2 \rightarrow x_3 \begin{array}{l} \nearrow x_4 \\ \searrow x_5 \end{array}, \quad (2.43)$$

is not equivalent to the cluster algebra on any Grassmannian (once again, ignore the notational overlap with eq. (2.41) and treat all of the x_i as generic variables). However, it appears as a subalgebra of any $\text{Gr}(k, n)$ with rank greater than five. The D_4 cluster algebra consists of 50 clusters, 16 \mathcal{A} -coordinates, and 104 \mathcal{X} -coordinates, while D_5 has 182 clusters, 25 \mathcal{A} -coordinates, and 260 \mathcal{X} -coordinates.

Finally, the cluster algebra E_6 is generated by the quiver

$$x_1 \rightarrow x_2 \rightarrow x_3 \begin{array}{c} \uparrow x_4 \\ \leftarrow x_5 \leftarrow x_6 \end{array}. \quad (2.44)$$

This cluster algebra is equivalent to $\text{Gr}(4, 7)$, as can be seen by mutating the seed (2.23) on

nodes $f_{12}, f_{13}, f_{23}, f_{12}, f_{22}$, and then f_{32} . By comparison with (2.44), we then have

$$\begin{aligned} x_1 &= \frac{\langle 1237 \rangle \langle 1246 \rangle}{\langle 1234 \rangle \langle 1267 \rangle}, & x_2 &= -\frac{\langle 4(12)(35)(67) \rangle}{\langle 1247 \rangle \langle 3456 \rangle}, \\ x_3 &= \frac{\langle 1245 \rangle \langle 1267 \rangle \langle 3456 \rangle}{\langle 1246 \rangle \langle 5(12)(34)(67) \rangle}, & x_4 &= -\frac{\langle 1234 \rangle \langle 4567 \rangle}{\langle 4(12)(35)(67) \rangle}, \\ x_5 &= -\frac{\langle 1567 \rangle \langle 4(12)(35)(67) \rangle}{\langle 1267 \rangle \langle 1345 \rangle \langle 4567 \rangle}, & x_6 &= \frac{\langle 5(12)(34)(67) \rangle}{\langle 1567 \rangle \langle 2345 \rangle}, \end{aligned} \quad (2.45)$$

where we have made use of the notation

$$\langle a(bc)(de)(fg) \rangle \equiv \langle abde \rangle \langle acfg \rangle - \langle abfg \rangle \langle acde \rangle. \quad (2.46)$$

Any \mathcal{A} -coordinate on $\text{Gr}(4, 7)$ can be expressed in terms of these \mathcal{X} -coordinates using the momentum twistor matrix

$$Z_{E_6} = \begin{pmatrix} -x_2 & 0 & -(1+x_4) & 0 & -x_2 & 1+x_{5,6} & (1+x_2)(1+x_{5,6}) \\ 1+x_{4,3,5} & 1 & x_3x_4x_5x_6 & 0 & x_3x_5x_6(1+x_2) & x_5x_6 & -x_4x_5x_6 \\ 0 & 0 & x_3x_4 & 0 & x_2x_3 & 1 & 1 \\ 0 & 0 & -x_1x_3x_4 & 1 & 1+x_{3,2} & 1 & 0 \end{pmatrix}, \quad (2.47)$$

where we have introduced the notation

$$x_{i_1, \dots, i_k} = \sum_{a=1}^k \prod_{b=1}^a x_{i_b} = x_{i_1} + x_{i_1}x_{i_2} + \dots + x_{i_1} \cdots x_{i_k}. \quad (2.48)$$

The associahedron of $\text{Gr}(4, 7) \simeq E_6$ was displayed in figure 4; it contains 833 clusters, 42 \mathcal{A} -coordinates, and 770 \mathcal{X} -coordinates. The subalgebras of E_6 , as well as those of its subalgebras, are tabulated in appendix B.

2.6 Cluster Automorphisms

Cluster algebras come equipped with an automorphism group that maps the set of cluster coordinates (but not necessarily the set of clusters) back to itself. We introduce here only what we need to elucidate the automorphisms of the cluster algebras introduced in the last section, and refer the interested reader to [37] for a more thorough mathematical introduction. Note that we describe automorphisms in terms of \mathcal{X} -coordinates, whereas [37] works in the \mathcal{A} -coordinate language.

The simplest example of a cluster automorphism is what we call a direct automorphism. Let \mathcal{A} be a cluster algebra equipped with a mutation rule $\mu(x_i, \mathbf{x})$ that mutates the cluster \mathbf{x} on node x_i . Then, we can define:

Direct Automorphism: The map $f : \mathcal{A} \rightarrow \mathcal{A}$ is a direct automorphism of \mathcal{A} if

- (i) for every cluster \mathbf{x} of \mathcal{A} , $f(\mathbf{x})$ is also a cluster of \mathcal{A} ,
- (ii) f respects mutations, i.e. $f(\mu(x_i, \mathbf{x})) = \mu(f(x_i), f(\mathbf{x}))$.

An example of a direct automorphism on A_2 is given by

$$\sigma_{A_2} : \mathcal{X}_i \rightarrow \mathcal{X}_{i+1}, \quad (2.49)$$

where we are using the coordinates introduced in (2.16). This automorphism cycles the five clusters $1/\mathcal{X}_i \rightarrow \mathcal{X}_{i+1}$ amongst themselves. The action of this automorphism can also be recast as

$$\sigma_{A_2} : x_1 \rightarrow \frac{1}{x_2}, \quad x_2 \rightarrow x_1(1 + x_2), \quad (2.50)$$

using the \mathcal{X} -coordinates x_1 and x_2 that appear in (2.33).

Cluster algebras are also endowed with what we call indirect automorphisms, which respect mutations but do not map the set of clusters back to itself. Instead, indirect automorphisms map the clusters in \mathcal{A} to clusters in \mathcal{A}' , where \mathcal{A}' is constructed from \mathcal{A} by multiplicatively inverting all cluster \mathcal{X} -coordinates and reversing the direction of all quiver arrows. Then we have:

Indirect Automorphism: The map $f : \mathcal{A} \rightarrow \mathcal{A}'$ is an indirect automorphism if

- (i) for every cluster \mathbf{x} of \mathcal{A} , $f(\mathbf{x})$ is a cluster of \mathcal{A}'
- (ii) f respects mutations, i.e. $f(\mu(x_i, \mathbf{x})) = \mu(f(x_i), f(\mathbf{x}))$.

A_2 is also equipped with an indirect automorphism generated by

$$\tau_{A_2} : \mathcal{X}_i \rightarrow \mathcal{X}_{6-i}, \quad (2.51)$$

where indices are understood to be mod 5. This can be recast in term of x_1 and x_2 as

$$\tau_{A_2} : x_1 \rightarrow \frac{1}{x_2}, \quad x_2 \rightarrow \frac{1}{x_1}. \quad (2.52)$$

To see that this is an indirect automorphism, consider

$$\tau_{A_2}(1/\mathcal{X}_1 \rightarrow \mathcal{X}_2) = 1/\mathcal{X}_5 \rightarrow \mathcal{X}_4. \quad (2.53)$$

Inverting the cluster coordinates on the right hand side and reversing the arrow, we get back to $\mathcal{X}_5 \leftarrow 1/\mathcal{X}_4$, which was one of the original clusters of A_2 . It is useful to think of indirect automorphisms as generating a “mirror” or “flipped” version of the original cluster algebra, where the total collection of \mathcal{X} -coordinates is the same, but their Poisson bracket has flipped sign. The existence of this flip then can be seen as resulting from the arbitrary choice of overall sign for the exchange matrix; picking the other sign would have generated the same cluster-algebraic structure, but with different labels for the nodes. Indirect automorphisms capture the superficiality of this notation change.

The automorphisms σ_{A_2} and τ_{A_2} generate the complete automorphism group for A_2 , namely the dihedral group D_5 (we apologize for the overloaded notation; unless explicitly noted otherwise, D_n refers to the Dynkin diagram throughout). More generally, cluster

algebras of type A_n have as their automorphism group the dihedral group D_{n+3} , which is generated by a cyclic (direct automorphism) generator

$$\sigma_{A_n} : \quad x_{k < n} \rightarrow \frac{x_{k+1}(1 + x_{1,\dots,k-1})}{1 + x_{1,\dots,k+1}}, \quad x_n \rightarrow \frac{1 + x_{1,\dots,n-1}}{\prod_{i=1}^n x_i}, \quad (2.54)$$

and a flip (indirect automorphism) generator

$$\tau_{A_n} : \quad x_1 \rightarrow \frac{1}{x_n}, \quad x_2 \rightarrow \frac{1}{x_{n-1}}, \quad \dots, \quad x_n \rightarrow \frac{1}{x_1}. \quad (2.55)$$

Both here and below we have made use of the notation introduced in (2.48).

The cluster algebra D_4 has automorphism group $D_4 \times S_3$ (where the first factor is the dihedral group of order eight, and the second is the symmetric group of order six). Each factor comes with a cyclic (direct automorphism) generator,

$$\begin{aligned} \sigma_{D_4}^{(4)} : \quad & x_1 \rightarrow \frac{x_2}{1 + x_{1,2}}, \quad x_2 \rightarrow \frac{(1 + x_1)x_1x_2x_3x_4}{(1 + x_{1,2,3})(1 + x_{1,2,4})}, \quad x_3 \rightarrow \frac{1 + x_{1,2}}{x_1x_2x_3}, \quad x_4 \rightarrow \frac{1 + x_{1,2}}{x_1x_2x_4}, \\ \sigma_{D_4}^{(3)} : \quad & x_1 \rightarrow \frac{1}{x_3}, \quad x_2 \rightarrow \frac{x_1x_2(1 + x_3)}{1 + x_1}, \quad x_3 \rightarrow x_4, \quad x_4 \rightarrow \frac{1}{x_1}, \end{aligned} \quad (2.56)$$

where $\sigma_{D_4}^{(4)}$ has length four and is associated with the dihedral group, and $\sigma_{D_4}^{(3)}$ has length three and is associated with the symmetric group. Then there are two flip generators

$$\tau_{D_4} : \quad x_2 \rightarrow \frac{1 + x_1}{x_1x_2(1 + x_3)(1 + x_4)}, \quad (2.57)$$

$$\mathbb{Z}_{2,D_4} : \quad x_3 \rightarrow x_4, \quad x_4 \rightarrow x_3,$$

where τ_{D_4} is associated with the dihedral group and generates an indirect automorphism, and \mathbb{Z}_{2,D_4} is associated with the symmetric group and generates a direct automorphism.

The cluster algebras on $D_{n>4}$, namely the quiver

$$x_1 \rightarrow x_2 \rightarrow \dots \rightarrow x_{n-2} \begin{array}{l} \nearrow x_{n-1} \\ \searrow x_n \end{array}, \quad (2.58)$$

have automorphism group $D_n \times \mathbb{Z}_2$ (the first factor representing the dihedral group), with generators σ_{D_n} (length n , direct), τ_{D_n} (length 2, indirect), and \mathbb{Z}_{2,D_n} (length 2, direct). In the case of D_5 , these generators can be chosen to be

$$\begin{aligned}
\sigma_{D_5} : \quad & x_1 \rightarrow \frac{x_2}{1+x_{1,2}}, \quad x_2 \rightarrow \frac{(1+x_1)x_3}{1+x_{1,2,3}}, \quad x_3 \rightarrow \frac{x_1x_2x_3x_4x_5(1+x_{1,2})}{(1+x_{1,2,3,4})(1+x_{1,2,3,5})}, \\
& x_4 \rightarrow \frac{1+x_{1,2,3}}{x_1x_2x_3x_4}, \quad x_5 \rightarrow \frac{1+x_{1,2,3}}{x_1x_2x_3x_5}, \\
\tau_{D_5} : \quad & x_1 \rightarrow x_1, \quad x_2 \rightarrow \frac{1+x_1}{x_1x_2(1+x_3x_5+x_{3,4,5})}, \quad x_3 \rightarrow \frac{x_3x_4x_5}{(1+x_{3,4})(1+x_{3,5})}, \\
& x_4 \rightarrow \frac{1+x_3x_5+x_{3,4,5}}{x_4}, \quad x_5 \rightarrow \frac{1+x_3x_5+x_{3,4,5}}{x_5},
\end{aligned} \tag{2.59}$$

$$\mathbb{Z}_{2,D_n} : \quad x_4 \rightarrow x_5, \quad x_5 \rightarrow x_4.$$

More generally, for D_n cluster algebras, the action of \mathbb{Z}_2 is always realized by the exchange $x_{n-1} \leftrightarrow x_n$.

Finally, the automorphism group of $E_6 \simeq \text{Gr}(4, 7)$ is the dihedral group D_{14} . This group has generators σ_{E_6} (length 7, direct), τ_{E_6} (length 2, indirect), and \mathbb{Z}_{2,E_6} (length 2, direct). In the coordinates of the defining quiver (2.44), these can be chosen to be

$$\begin{aligned}
\sigma_{E_6} : \quad & x_1 \rightarrow \frac{1}{x_6(1+x_{5,3,4})}, \quad x_2 \rightarrow \frac{1+x_{6,5,3,4}}{x_5(1+x_{3,4})}, \quad x_3 \rightarrow \frac{(1+x_{2,3,4})(1+x_{5,3,4})}{x_3(1+x_4)}, \\
& x_4 \rightarrow \frac{1+x_{3,4}}{x_4}, \quad x_5 \rightarrow \frac{1+x_{1,2,3,4}}{x_2(1+x_{3,4})}, \quad x_6 \rightarrow \frac{1}{x_1(1+x_{2,3,4})}, \\
\mathbb{Z}_{2,E_6} : \quad & x_i \rightarrow x_{7-i}, \\
\tau_{E_6} : \quad & x_1 \rightarrow \frac{x_5}{1+x_{6,5}}, \quad x_2 \rightarrow (1+x_5)x_6, \quad x_3 \rightarrow \frac{(1+x_{1,2})(1+x_{6,5})}{x_1x_2x_3x_5x_6(1+x_4)}, \\
& x_4 \rightarrow x_4, \quad x_5 \rightarrow x_1(1+x_2), \quad x_6 \rightarrow \frac{x_2}{1+x_{1,2}}.
\end{aligned} \tag{2.60}$$

In the language of $\text{Gr}(4, 7)$, these generators respectively correspond to cycling momentum twistor indices $Z_i \rightarrow Z_{i+1}$, flipping momentum twistor indices $Z_i \rightarrow Z_{8-i}$, and parity.

The dihedral and parity symmetries of the n -particle MHV amplitude are directly connected to equivalent automorphisms of $\text{Gr}(4, n)$. However—surprisingly—portions of these amplitudes can also be seen to respect the automorphism group of subalgebras of $\text{Gr}(4, n)$. Making clear precisely what we mean by this statement will be the focus of the remainder of this paper.

3 Cluster Polylogarithms and MHV Amplitudes

The BDS ansatz captures the infrared structure of planar $\mathcal{N} = 4$ SYM to all orders in the coupling [26]. In four- and five-particle kinematics it also furnishes the complete finite part of the amplitude, while for six or more particles it must be corrected by a finite dual-conformally invariant function [27–29]. In the case of the MHV amplitude, this correction is

often computed in the form of the n -particle remainder function R_n , defined by

$$\mathcal{A}_n^{\text{MHV}} = \mathcal{A}_n^{\text{BDS}} \times \exp(R_n), \quad (3.1)$$

where $\mathcal{A}_n^{\text{BDS}}$ is the BDS ansatz for n particles [26]. Like the amplitude, the remainder function can be expanded in the coupling

$$R_n = g^4 R_n^{(2)} + g^6 R_n^{(3)} + g^8 R_n^{(4)} + \dots, \quad (3.2)$$

where $g^2 = \frac{g_{\text{YM}}^2 N_c}{16\pi^2}$. In this expansion we have used the fact that $R_n^{(1)} = 0$, since the BDS ansatz encodes the complete one loop MHV amplitude at all n .

The remaining L -loop contributions to the remainder function are expected to be expressible in terms of generalized polylogarithms [38–40] of uniform transcendental weight $2L$. This space is spanned by (products of) the functions

$$G(a_1, \dots, a_k; z) \equiv \int_0^z \frac{dt}{t - a_1} G(a_2, \dots, a_k; t), \quad G(\underbrace{0, \dots, 0}_k; z) \equiv \frac{\log^k z}{k!}, \quad (3.3)$$

where $G(; z) \equiv 1$, and the transcendental weight of each function corresponds to its number of indices k . In particular, the remainder function is expected to be a pure function of this type, meaning that its kinematic dependence appears in the indices and arguments a_i and z , but not in the rational prefactors multiplying these functions. This is known to be true at two loops, due to an impressive all- n computation that leveraged the superconformal symmetry of this theory [6], as well as through six loops in six-particle kinematics [1, 2, 4, 41] and through four loops in seven-particle kinematics [3, 5].

In addition to being generalized polylogarithms, loop-level contributions to the remainder function exhibit a great deal of cluster-algebraic structure. In particular, they are members of the space of ‘cluster polylogarithms’ studied in [10], indicating that their symbol is naturally expressible in terms of cluster \mathcal{A} -coordinates, while their Lie cobracket is naturally expressible in terms of cluster \mathcal{X} -coordinates (in a way that will be made precise below). Their cobracket, moreover, has been shown to be decomposable into simple functions associated with their A_2 and A_3 subalgebras [10]. As will be shown in the next section, these functions (the ‘ A_2 function’ and the ‘ A_3 function’) are invariant under the automorphism group of the algebras on which they are defined, up to a sign. This expresses the fact that these functions are well-defined under coordinate relabelings (or, are well-defined functions of oriented graphs). We correspondingly propose that the space of cluster polylogarithms be refined to include only functions that respect the automorphism group of the cluster algebra on which they are defined. In addition to this subalgebra structure, the symbols of these amplitudes have been found to satisfy a ‘cluster adjacency’ principle [12], and their cobracket takes a similarly restricted form [7]. The rest of this section is devoted to making these properties precise, for which purpose we first describe the motivic structure of polylogarithms.

3.1 The Symbol and Cobracket

The space of generalized polylogarithms defined by (3.3) is colossally overcomplete. This is because a_i and z are allowed to be arbitrarily complicated algebraic functions, and because these polylogarithms satisfy a shuffle and stuffle algebra. The shuffle algebra represents the fact that unordered integrations can be triangulated into a sum over iterated integrals [42, 43]. In general, this means that when two polylogarithms share an argument z , their product can be re-expressed as the sum of functions

$$G(a_1, \dots, a_{k_1}; z) G(a_{k_1+1}, \dots, a_{k_1+k_2}; z) = \sum_{\sigma \in \Sigma(k_1, k_2)} G(a_{\sigma(1)}, \dots, a_{\sigma(k_1+k_2)}; z), \quad (3.4)$$

where $\Sigma(k_1, k_2)$ denotes the set of all shuffles between the sets of integers $\{1, \dots, k_1\}$ and $\{k_1 + 1, \dots, k_1 + k_2\}$ (that is, all ways of interleaving these two sets such that the ordering of the elements within each of the original sets is maintained). The stuffle algebra naturally arises when generalized polylogarithms are re-expressed as infinite sums,

$$\begin{aligned} \text{Li}_{n_1, \dots, n_d}(z_1, \dots, z_d) &\equiv \sum_{0 < m_1 < \dots < m_d} \frac{z_1^{m_1} \dots z_d^{m_d}}{m_1^{n_1} \dots m_d^{n_d}} \\ &= (-1)^d G(\underbrace{0, \dots, 0}_{n_d-1}, \frac{1}{z_d}, \dots, \underbrace{0, \dots, 0}_{n_1-1}, \frac{1}{z_1 \dots z_d}; 1) \end{aligned} \quad (3.5)$$

where d is called the depth of the polylogarithm. Stuffle identities represent the freedom to split up unordered summation indices (arising from products of polylogarithms) into nested sums where these indices are ordered, as in (3.5).

This overcompleteness gives rise to a rich space of identities. This can already be seen at the level of classical polylogarithms, which correspond to the instances of (3.5) with depth one. **[description of five-term dilog identity, refer to further weight three/four identities [8?]]**

Fortunately, *all* identities between polylogarithms are (conjecturally) generated by shuffle and stuffle relations [? ?]. These relations are trivialized (up to logarithmic identities) by the symbol map $\llbracket \cdot \rrbracket$, which can be defined in terms of total derivatives. For instance, taking the total derivative of (3.3), we have

$$dG(a_1, \dots, a_k; z) = \sum_{i=1}^k G(a_1, \dots, \hat{a}_i, \dots, a_k; z) d \log \left(\frac{a_{k-i+1} - a_{k-i}}{a_{k-i+1} - a_{k-i+2}} \right) \quad (3.6)$$

where $a_0 \equiv z$ and $a_{k+1} \equiv 0$, and the notation \hat{a}_i indicates this index should be omitted [?]. The symbol map is then defined recursively by

$$\mathcal{S}(G(a_1, \dots, a_k; z)) \equiv \sum_{i=1}^k \mathcal{S}(G(a_1, \dots, \hat{a}_i, \dots, a_k; z)) \otimes \left(\frac{a_{k-i+1} - a_{k-i}}{a_{k-i+1} - a_{k-i+2}} \right). \quad (3.7)$$

The entries of the resulting k -fold tensor product are referred to as symbol letters. These symbol letters inherit the distributive properties of (arguments of) logarithms, and can therefore be expanded into a multiplicatively independent basis of symbol letters (the ‘symbol alphabet’). Complicated polylogarithmic identities are thereby reduced to identities between logarithms, at the cost of losing information about the boundary of integration in (3.3).

The symbol also captures the analytic structure of polylogarithms, insofar as it encodes their (iterated) discontinuity structure. Namely, for generic indices a_i and argument z , these functions have nonzero monodromy only where the letters in the first entry of their symbol vanish or become infinite. These monodromies can themselves have branch cuts that the original function did not have, when new symbol letters appear in the second entry of the symbol, *ad infinitum*. For special values of a_i and z , some of this information is lost due to the fact that higher-weight transcendental constants (such as Riemann ζ values) are in the kernel of the symbol [40]. However, this information is retained by the coaction [44], and can be recovered by specifying an integration constant at each weight. We defer further consideration of these transcendental constants to a follow-up paper [].

In addition to the symbol, polylogarithms come equipped with a Lie cobracket structure [8]. The cobracket δ can be calculated using the symbol projection operator

$$\rho(s_1 \otimes \cdots \otimes s_k) = \frac{k-1}{k} \left(\rho(s_1 \otimes \cdots \otimes s_{k-1}) \otimes s_k - \rho(s_2 \otimes \cdots \otimes s_k) \otimes s_1 \right), \quad (3.8)$$

where $\rho(s_1) \equiv s_1$. This projects onto the component of a symbol that cannot be written as a product of lower-weight polylogarithms (for instance, via the shuffle relations (3.4)). The action of ρ can be lifted to a projection on functions, up to terms proportional to transcendental constants; since we will not be concerned with these terms in what follows, we will abuse notation by applying ρ to functions directly. The cobracket δ of a weight k polylogarithm f can then be calculated as

$$\delta(f) \equiv \sum_{i=1}^{k-1} (\rho_i \wedge \rho_{k-i}) \rho(f). \quad (3.9)$$

This notation indicates that the projection operator ρ is first applied to f , after which each term in the resulting sum is partitioned into a wedge product of weight i and weight $k-i$ functions (either by splitting up the symbol into its first i and last $k-i$ entries, or by taking the ‘ $i, k-i$ component’ of the coproduct); the projection operator is then applied to the entries in this wedge product separately. In general, the wedge product in (3.9) involves spaces of different weight. Without loss of generality, we can put the cobracket of any function into a form where the first factor of the wedge product has weight equal to or higher than that of the second factor (exchanging the order of factors when needed, at the cost of a minus sign). We denote by $\delta_{i,j}(f)$ the component of the cobracket of f that involves a wedge product of weight i and j functions, in that order—but we emphasize that this includes contributions from all terms in (3.9) that involve these weights in either order.

In the context of two-loop amplitudes, the salient property of the cobracket is that it isolates the component of weight four polylogarithms that cannot be written in terms of classical polylogarithms. Under the action of ρ , classical polylogarithms are mapped to elements of the Bloch group B_k [? ?], namely the algebra of polylogarithms modulo identities between classical polylogarithms. Following [8], we denote these elements by

$$\{z\}_k = \rho(-\text{Li}_k(-z)) \in B_k. \quad (3.10)$$

For instance, **[maybe show that this trivializes an identity?]**. In this language, the action of the cobracket on classical polylogarithms is given by

$$\delta(\text{Li}_k(-z)) = -\{z\}_{k-1} \wedge \{-1-z\}_1. \quad (3.11)$$

Note that, for the first time at weight four, there exists a component of the cobracket that is not mapped to by classical polylogarithms—namely, $\delta_{2,2}(\text{Li}_4(-z)) = 0$. This is not true of weight four polylogarithms in general, and in particular $\delta_{2,2}(R_n^{(2)})$ is nonzero for $n \geq 7$. However, it has been shown that any weight four function that *is* annihilated by $\delta_{2,2}$ can be written in terms of classical polylogarithms (with potentially complicated arguments) [].

Lastly, it is often quite useful to employ the fact that the trivial cohomology of δ gives us an integrability condition,

$$\delta^2(f) = 0, \quad (3.12)$$

for any polylogarithm f . This condition implies an intricate relationship between the arguments of the Bloch group elements appearing in $\delta(f)$. For example, any weight 4 function f satisfies

$$\delta(\delta_{2,2}(f)) + \delta(\delta_{3,1}(f)) = 0. \quad (3.13)$$

So, while $\delta_{2,2}$ serves the important and unique role of capturing the nonclassical contribution to a function, the $\delta_{3,1}$ component is important as well and can be thought of as serving as a bridge between the nonclassical and classical.

The symbol and cobracket naturally stratify the study of two-loop amplitudes and integrals that can be expressed as polylogarithms. The operator $\delta_{2,2}$ isolates the nonclassical component of these functions, while the symbol captures their analytic structure up to terms proportional to transcendental constants. In the case of MHV amplitudes in planar $\mathcal{N} = 4$ SYM, both objects turn out to distill intriguing cluster-algebraic structure that would otherwise be hard to see at the level of full functions. It is to this structure that we now turn.

3.2 Cluster-Algebraic Structure in at Two Loops

As outlined in the introduction, the two-loop MHV amplitudes of this theory exhibit different forms of cluster-algebraic structure at the level of their cobracket, their symbol, and as full functions [7, 8, 10, 11]. The first facet of this structure concerns the building blocks that appear at each level, which are found to lie within restricted classes:

- The cobracket of $R_n^{(2)}$ can be written in terms of elements of the Block group taking the form $\{\mathcal{X}_i\}_k$, where \mathcal{X}_i is an \mathcal{X} -coordinate on $\text{Gr}(4, n)$
- The symbol of $R_n^{(2)}$ can be expressed in terms of symbol letters drawn from the set of \mathcal{A} -coordinates on $\text{Gr}(4, n)$
- The function $R_n^{(2)}$ can be expressed entirely in terms of (products of) polylogarithms taking the form $\text{Li}_{n_1, \dots, n_d}(-\mathcal{X}_i, \dots, -\mathcal{X}_j)$, where each \mathcal{X}_p is again an \mathcal{X} -coordinate on $\text{Gr}(4, n)$

The physical meaning of the symbol alphabet restriction is clear, if unilluminating—the kinematic configurations in which these amplitudes are singular (due to internal propagators going on shell in the Feynman diagram expansion [**maybe this doesn't explain singularities in y_u -type letters?**]) coincide with the vanishing loci of certain cluster \mathcal{A} -coordinates. Even so, it is not clear how (or if) this restriction follows from physical principles. In this respect, the positive Grassmannian formulation of this theory is suggestive, insofar as the integrands of these amplitudes are seen to develop physical singularities only where \mathcal{A} -coordinates vanish [**or become infinite?**][**check**] [21]. However, this does not preclude the emergence of new branch cuts during integration, as has already been observed in Feynman integrals contributing to non-MHV amplitudes in this theory [18, 19]. The physical meaning of the restricted functional form and cobracket structure exhibited by these amplitudes remains even more obscure.

Cluster algebras also seem to play a role in how these building blocks are assembled in amplitudes. In particular, it was recently observed that—when these amplitudes are normalized appropriately—the only cluster \mathcal{A} -coordinates that appear in adjacent entries of their symbol are those that appear together in at least one cluster of $\text{Gr}(4, n)$ [12]. This ‘cluster adjacency’ principle is not enjoyed by the remainder function (3.1), but by BDS-normalized amplitudes \mathcal{E}_n [4, 5, 45, 46]. These are defined by

$$\mathcal{A}_n^{\text{MHV}} = \mathcal{A}_n^{\text{BDS-like}} \times \mathcal{E}_n, \quad (3.14)$$

where $\mathcal{A}_n^{\text{BDS-like}}$ is related to $\mathcal{A}_n^{\text{BDS}}$ by the cusp anomalous dimension [47]

$$\Gamma_{\text{cusp}} = 4g^2 - 8\zeta_2 g^4 + \mathcal{O}(g^6) \quad (3.15)$$

and a simple weight two polylogarithm Y_n via

$$\mathcal{A}_n^{\text{BDS-like}} = \mathcal{A}_n^{\text{BDS}} \times \exp\left(\frac{\Gamma_{\text{cusp}}}{4} Y_n\right). \quad (3.16)$$

The function Y_n corresponds to the part of the one-loop MHV amplitude that depends on three- and higher-particle Mandelstam invariants, where these invariants have been assembled into dual-conformally-invariant cross ratios (with the help of two-particle invariants). The BDS-like ansatz that remains only depends on two-particle invariants, yet accounts for the full infrared structure of these amplitudes.

The motivation for switching to the BDS-like normalization is precisely this restricted kinematic dependence. Since the BDS-like ansatz depends only on two-particle invariants, the functions \mathcal{E}_n directly inherit the Steinmann relations between three- and higher-particle invariants that are obeyed by the full amplitude [4, 5, 48–50]. These relations tell us that [check]

$$\left. \begin{aligned} \text{Disc}_{s_j, \dots, j+p+q} [\text{Disc}_{s_i, \dots, i+p}(\mathcal{E}_n)] &= 0, \\ \text{Disc}_{s_i, \dots, i+p} [\text{Disc}_{s_j, \dots, j+p+q}(\mathcal{E}_n)] &= 0, \end{aligned} \right\} \quad 0 < j - i \leq p \quad \text{or} \quad q < i - j \leq p + q, \quad (3.17)$$

where all indices should be considered mod n . Formulated in terms of symbol entries, this implies that the cluster \mathcal{A} -coordinates $\langle j-1, j, j+p+q-1, j+p+q \rangle$ and $\langle i-1, i, i+p-1, i+p \rangle$ never appear next to each other in the first two entries of the symbol. In fact, it is believed that these constraints can be applied at all depths in the symbol, as these letters are never seen to appear next to each other [4, 5, 13, 20, 41]. These generalized constraints have been termed the extended Steinmann relations, as they amount to applying the relations (3.17) to all discontinuities of the amplitude in addition to the amplitude itself.

The fact that the functions \mathcal{E}_n also obey the cluster adjacency principle is far more surprising. The constraints that follow from this principle take a form similar to the extended Steinmann relations (insofar as they restrict which symbol letters can appear in adjacent entries), and in fact turn out to be equivalent in six-particle kinematics when applied to functions with physical branch cuts. It is therefore tempting to believe cluster adjacency follows from some set of physical principles that includes the extended Steinmann relations. However, it is not yet known whether these two conditions are equivalent at all n .

Seeming to complicate this question of equivalence is the fact that the BDS-like ansatz (3.16) only exists when n is not a multiple of four. This is because no function satisfying the above description of Y_n exists for these particle multiplicities [5, 51]. (Such a function not only exists for all other n , but is uniquely picked out by this description.) Stated another way, when n is a multiple of four, any normalization that absorbs the infrared-divergent part of the amplitude either depends on some set of three- or higher-particle Mandelstam invariants, or spoils the dual conformal invariance of the resulting normalized amplitude. However, this shows this complication is superficial—for the purpose of understanding the relationship between the extended Steinmann relations and the cluster adjacency principle, we need merely choose the latter horn of this dilemma, and give up dual conformal invariance.

This issue first arises in eight-particle kinematics. There it can be seen—by direct computation—that both sets of constraints are obeyed when the amplitude is normalized by a generalized BDS-like ansatz whose kinematic dependence is restricted to two-particle Mandelstam invariants [?]. This provides further evidence that the extended Steinmann relations and the cluster adjacency principle are encoding the same information.

While cluster adjacency was first observed in \mathcal{A} -coordinates, it can also be formulated in terms of \mathcal{X} -coordinates. Unlike \mathcal{A} -coordinates, which are multiplicatively independent, \mathcal{X} -coordinates satisfy numerous multiplicative identities. Thus, there will exist many representations of the same symbol in terms of \mathcal{X} -coordinates, whereas its representation in terms

of \mathcal{A} -coordinates is unique. In general, only a subset of these \mathcal{X} -coordinate representations will satisfy cluster adjacency (even if the \mathcal{A} -coordinate representation does). Moreover, while \mathcal{X} -coordinate adjacency trivially implies \mathcal{A} -coordinate adjacency by the relation (2.10), the converse can not be true in general. This is because \mathcal{A} -coordinates can express a larger class of functions than \mathcal{X} -coordinates, insofar as the latter necessarily respect dual conformal symmetry while the former do not. However, it can be checked that every dual-conformal-invariant symbol that satisfies \mathcal{A} -coordinate adjacency in $\text{Gr}(k \leq 4, n \leq 7)$ has an \mathcal{X} -coordinate adjacent representation. Although we do not have a general proof of this, we conjecture that this remains true for all dual-conformally-invariant symbols constructed on $\text{Gr}(k, n)$, for general k and n . **[need to actually check that DCI + A-adjacency implies X-adjacency for all $\text{Gr}(k \leq 4, n \leq 7)$, tho that may well be too much work, instead we can check for as many cases as are computationally easy and leave the rest as conjecture – john]**

The cobracket $\delta(R_n^{(2)})$ also satisfies its own form of cluster adjacency [11]. Namely, $\delta(R_n^{(2)})$ can be expressed as a linear combination of terms $\{\mathcal{X}_i\}_2 \wedge \{\mathcal{X}_j\}_2$ and $\{\mathcal{X}_k\}_3 \otimes \mathcal{X}_l$ where $\{\mathcal{X}_i, \mathcal{X}_j\}$ and $\{\mathcal{X}_k, \mathcal{X}_l\}$ appear together in a cluster of $\text{Gr}(4, n)$. For example, $\delta_{2,2}(R_n^{(2)})$ can be expressed as a sum of terms of the form $\{v_{ijk}\}_2 \wedge \{z_{pqr}^\pm\}_2$, where (using the notation defined in (2.46))

$$v_{ijk} = -\frac{\langle i+1(i, i+2)(j, j+1)(k, k+1) \rangle}{\langle i, i+1, k, k+1 \rangle \langle i+1, i+2, j, j+1 \rangle}, \quad (3.18)$$

and

$$\begin{aligned} z_{ijk}^+ &= \frac{\langle i, j-1, j, j+1 \rangle \langle i+1, k-1, k, k+1 \rangle - \langle i+1, j-1, j, j+1 \rangle \langle i, k-1, k, k+1 \rangle}{\langle i, k-1, k, k+1 \rangle \langle i+1, j-1, j, j+1 \rangle}, \\ z_{ijk}^- &= \frac{\langle i, i+1, j, k \rangle \langle i-1, i, i+1, i+2 \rangle}{\langle i-1, i, i+1, k \rangle \langle i, i+1, i+2, j \rangle}. \end{aligned} \quad (3.19)$$

As long as $i < j < k$ (considered mod n), the quantities v_{ijk} , z_{ijk}^+ , and z_{ijk}^- each constitute \mathcal{X} -coordinates, where z_{ijk}^+ and z_{ijk}^- are also parity conjugate to each other. The coordinates v_{ijk} were originally motivated by consideration of the location of physical branch cuts, while the z_{ijk}^\pm were motivated by consideration of the final symbol entry of the remainder function, which is constrained to take the form $\langle i-1, i, i+1, j \rangle$ by dual superconformal symmetry [14]. However, for the present discussion, the pertinent point is that $\delta_{2,2}(R_n^{(2)})$ can be expressed as a sum of terms $\{v_{ijk}\}_2 \wedge \{z_{pqr}^\pm\}_2$ in which v_{ijk} and z_{pqr}^\pm also occur together in a cluster. Importantly, this form of cobracket-level cluster adjacency isn't implied by the cluster \mathcal{A} -coordinate adjacency observed in [12], as can be seen in the A_2 function we will define in the next subsection—this function satisfies cluster \mathcal{A} -coordinate adjacency, but the \mathcal{X} -coordinates appearing in its $\delta_{2,2}$ cobracket component cannot be chosen to satisfy cobracket-level cluster adjacency.

While this cobracket-level structure was originally observed in the remainder function, the same characterization carries over to the BDS-like normalized amplitude. This can be

seen by combining equations (3.1), (3.14), and (3.16) to express \mathcal{E}_n in terms of R_n :

$$\begin{aligned}\mathcal{E}_n &= \exp\left(R_n - \frac{\Gamma_{\text{cusp}}}{4}Y_n\right) \\ &= 1 - Y_n g^2 + \left(R_n^{(2)} + 2\zeta_2 Y_n + \frac{1}{2}Y_n^2\right)g^4 + \mathcal{O}(g^6).\end{aligned}\tag{3.20}$$

Thus, $\rho(\mathcal{E}_n^{(2)}) = \rho(R_n^{(2)})$, since these two functions differ only by products and terms involving transcendental constants. Correspondingly, the cobracket structure of the remainder function and the BDS-like normalized amplitude are identical. Moreover, the function Y_n can always be expressed in terms of classical polylogarithms taking negative \mathcal{X} -coordinates as arguments [check]. Thus, at the level of its cobracket, its symbol, and as a full function, $\mathcal{E}_n^{(2)}$ can be expressed in terms of the same restricted building blocks as $R_n^{(2)}$. To these observations we can now add:

- The cobracket $\delta(\mathcal{E}_n^{(2)})$ can be expressed as a linear combination of terms $\{\mathcal{X}_i\}_2 \wedge \{\mathcal{X}_j\}_2$ and $\{\mathcal{X}_k\}_3 \otimes \mathcal{X}_l$ where $\{\mathcal{X}_i, \mathcal{X}_j\}$ and $\{\mathcal{X}_k, \mathcal{X}_l\}$ appear together in a cluster of $\text{Gr}(4, n)$.
- Pairs of \mathcal{A} -coordinates only appear in adjacent entries of the symbol $\mathcal{S}(\mathcal{E}_n^{(2)})$ when they also appear together in at least one cluster of $\text{Gr}(4, n)$.

As discussed above, the last statement can also be applied at n that are multiples of four by going to a generalized BDS-like normalization in which only two-particle invariants appear in the normalizing function.

As it turns out, there exists yet more structure in $\delta_{2,2}(R_n^{(2)}) = \delta_{2,2}(\mathcal{E}_n^{(2)})$. In particular, it was shown in [10] that this cobracket component can be decomposed into a sum over various A_2 subalgebras of $\text{Gr}(4, n)$, by defining an A_2 function that can be evaluated on each of these subalgebras. Moreover, this A_2 function can be assembled into an A_3 function, in terms of which this cobracket component can similarly be decomposed. As this subalgebra decomposability will play a central role in what follows, we devote the next subsection to its description.

3.3 Subalgebra Structure and Cluster Polylogarithms

As we saw in section 2.3, cluster algebras are endowed with subalgebras that can be generated by mutating on restricted sets of nodes. This motivates looking for physically relevant cluster polylogarithms on algebras other than $\text{Gr}(4, n)$, when these algebras appear as subalgebras of the latter. Before we do so, let us return to the definition of these objects, which we are now in a position to make precise. Following [10], we define cluster polylogarithms (at least through weight four) to have the following properties:

Cluster Polylogarithm: A generalized polylogarithm f is a cluster polylogarithm on a cluster algebra \mathcal{A} if

- (i) the symbol alphabet of f is composed of only \mathcal{A} -coordinates on \mathcal{A} ,

- (ii) the cobracket of f can be expressed in terms of Bloch group elements with arguments drawn from the \mathcal{X} -coordinates of \mathcal{A} ,
- (iii) the function f is invariant under the automorphisms of \mathcal{A} , up to a sign.

Property (iii) can be thought of as the requirement that cluster polylogarithms be well-defined functions on the cluster algebra, or more specifically, on an oriented graph representing a cluster inside that algebra. In other words, since cluster algebras can be generated by any individual cluster inside the algebra, functions corresponding to a cluster algebra should not depend on which cluster one uses as input. For instance, if we wish to define a function related to the A_2 cluster algebra, as we shall do shortly, it should satisfy the property that $f_{A_2}(1/\mathcal{X}_1 \rightarrow \mathcal{X}_2) = \pm f_{A_2}(1/\mathcal{X}_2 \rightarrow \mathcal{X}_3)$, and similarly when the function is evaluated on the other clusters. This ambiguity in the overall sign, which will be discussed below, reflects the fact that some automorphisms flip the orientation of cluster algebras while others do not.

The only polylogarithm that can be formed on the rank one cluster algebra A_1 is $\log^n(x)$, which is trivially a cluster polylogarithm but too simple to be of interest. However, a nontrivial cluster polylogarithm can already be defined on A_2 . As A_2 subalgebras are generated by all pairs of connected nodes, it appears ubiquitously in $\text{Gr}(4, n)$.

A central fact about the ‘ A_2 function,’ also referred to as f_{A_2} , is that it is uniquely determined by the cluster polylogarithm conditions, modulo products of classical polylogarithms of weight ≤ 3 (see [10] for an in-depth discussion of the A_2 function). The analytic properties by adding and subtracting classical polylogarithms (which again must respect the automorphisms of A_2), and in this work choose to define the A_2 function (which can be thought of as a ‘function on an oriented graph’) as

$$\begin{aligned}
f_{A_2}(x_1 \rightarrow x_2) = \sum_{\text{skew-dihedral}} & \left[\text{Li}_{2,2} \left(-\frac{1}{\mathcal{X}_{i-1}}, -\frac{1}{\mathcal{X}_{i+1}} \right) - \text{Li}_{1,3} \left(-\frac{1}{\mathcal{X}_{i-1}}, -\frac{1}{\mathcal{X}_{i+1}} \right) \right. \\
& + 6 \text{Li}_3(-\mathcal{X}_{i-1}) \log(\mathcal{X}_{i+1}) + \frac{1}{2} \log(\mathcal{X}_{i-3}) \log(\mathcal{X}_i) \log^2(\mathcal{X}_{i-1}) \quad (3.21) \\
& \left. - \text{Li}_2(-\mathcal{X}_{i-1}) \log(\mathcal{X}_{i+1}) \left(3 \log(\mathcal{X}_{i-1}) - \log(\mathcal{X}_i) + \log(\mathcal{X}_{i+1}) \right) \right],
\end{aligned}$$

where the \mathcal{X}_i are defined in terms of x_1 and x_2 as in eq. (2.16), and the skew-dihedral sum indicates subtracting the dihedral flip $\mathcal{X}_i \rightarrow \mathcal{X}_{6-i}$ and taking a cyclic sum $i = 1$ to 5. This function differs in a few salient respects from the A_2 function defined in [10], but (necessarily) has the same cobracket

$$\delta(f_{A_2}) = - \sum_{\text{skew-dihedral}} \{ -\mathcal{X}_{i-1} \}_2 \wedge \{ -\mathcal{X}_{i+1} \}_2 + 3 \{ -\mathcal{X}_i \}_2 \wedge \{ -\mathcal{X}_{i+1} \}_2 + \frac{5}{2} \{ -\mathcal{X}_i \}_3 \otimes \mathcal{X}_{i+1}. \quad (3.22)$$

We highlight some of the properties enjoyed by this A_2 function below.

Remarkably, it was shown in [10] that all of the information contained in $\delta_{2,2}(R_n^{(2)})$ is encoded in the A_2 function, when this function is evaluated on some collection of the A_2

subalgebras of $\text{Gr}(4, n)$. That is,

$$\delta_{2,2}(R_n^{(2)}) = \sum_{(x_i \rightarrow x_j) \subset \text{Gr}(4, n)} c_{ij} \delta_{2,2}(f_{A_2}(x_i \rightarrow x_j)) \quad (3.23)$$

for some rational coefficients c_{ij} . Moreover, the terms in this sum can themselves be arranged into A_3 subalgebras, giving rise to a natural A_3 function of the form

$$f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3) \sim \sum_{(x_i \rightarrow x_j) \subset A_3} d_{ij} f_{A_2}(x_i \rightarrow x_j), \quad (3.24)$$

where the d_{ij} are some rational coefficients that we treat in full detail in the next section. For now, we merely highlight the fact that these coefficients can be chosen in such a way that $\delta_{2,2}(f_{A_3})$ contains only terms $\{\mathcal{X}_i\}_2 \otimes \{\mathcal{X}_j\}_2$ in which \mathcal{X}_i and \mathcal{X}_j appear together in at least one cluster of $\text{Gr}(4, n)$. Then, there exists a decomposition

$$\delta_{2,2}(R_n^{(2)}) = \sum_{(x_i \rightarrow x_j \rightarrow x_k) \subset \text{Gr}(4, n)} c_{ijk} \delta_{2,2}(f_{A_3}(x_i \rightarrow x_j \rightarrow x_k)). \quad (3.25)$$

Thus, the cobracket-level cluster adjacency enjoyed by these amplitudes is made manifest in the A_3 decomposition (3.25).

In the following sections we analyze these decompositions systematically, and find they can be extended to much larger subalgebras in seven-particle kinematics (although these techniques also extend to higher n [?]). As in the A_3 decomposition, the A_2 function will continue to play a privileged role. For this reason, we highlight some of the properties we have built into this function before turning to our general analysis.

Recall that the clusters of A_2 all take the form $1/\mathcal{X}_i \rightarrow \mathcal{X}_{i+1}$ in the \mathcal{X}_i variables (2.16). Computing the symbol

$$\begin{aligned} \mathcal{S}(f_{A_2}) = - \sum_{\text{skew-dihedral}} & \left[\mathcal{X}_2 \otimes \mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_3 + \mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_2 \otimes \mathcal{X}_1 + \mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_3 \otimes \mathcal{X}_2 \right. \\ & \left. - 2(\mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_2 \otimes \mathcal{X}_3 + \mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_4 \otimes \mathcal{X}_3 - \mathcal{X}_2 \otimes \mathcal{X}_3 \otimes \mathcal{X}_3 \otimes \mathcal{X}_4) \right], \end{aligned} \quad (3.26)$$

we see that f_{A_2} satisfies cluster \mathcal{X} -coordinate adjacency, and therefore also \mathcal{A} -coordinate adjacency (recall that the symbol is insensitive to the difference between \mathcal{X}_i and $1/\mathcal{X}_i$). Note that this means any sum of A_2 functions evaluated on the subalgebras of some larger cluster algebra also respects cluster \mathcal{A} -coordinate adjacency. At the level of a full function, f_{A_2} is also smooth and real-valued in the positive domain $x_1, x_2 > 0$. The A_2 cluster algebra plays a crucial role in endowing f_{A_2} with this analytic behavior, as $\text{Li}_{2,2}(x, y)$ and $\text{Li}_{1,3}(x, y)$ have branch cuts in three locations, namely $x = 1$, $y = 1$, and $xy = 1$. The first two branch cuts are trivially avoided as $-1/\mathcal{X}_i < 0$ for $x_1, x_2 > 0$. However, the last branch cut is also avoided because of the A_2 exchange relation:

$$0 < \left(-\frac{1}{\mathcal{X}_{i-1}} \right) \left(-\frac{1}{\mathcal{X}_{i+1}} \right) = \frac{1}{1 + \mathcal{X}_i} < 1. \quad (3.27)$$

Finally, we note that (just like $R_n^{(2)}$ and $\mathcal{E}_n^{(2)}$) f_{A_2} is expressible entirely in terms of polylogarithms taking the form $\text{Li}_{n_1, \dots, n_d}(-\mathcal{X}_i, \dots, -\mathcal{X}_j)$, where each \mathcal{X}_p is an \mathcal{X} -coordinate. This can be seen from the representation in (3.21), since the multiplicative inverses of \mathcal{X} -coordinates also always appear as \mathcal{X} -coordinates, and since the arguments of the logarithms can all be put in this form at the cost of factors of $i\pi$.

4 Cluster Subalgebra-Constructibility

We now turn to a more systematic exploration of the space of cluster polylogarithms that are subalgebra-constructible in the same way that $R_n^{(2)}$ is constructible in terms of its A_2 and A_3 subalgebras. Our focus will be on functions that can be defined on $\text{Gr}(4, 7) \simeq E_6$ and its subalgebras, which are complex enough to admit nontrivial functional embeddings, but which can be exhaustively explored since they are all finite. All of the functions we consider will also appear in the subalgebras of $\text{Gr}(4, n > 7)$, and hence many of the techniques we utilize can also be applied to these infinite cluster algebras [10?]. We will be particularly interested in searching for further ways in which the nonclassical component of $R_7^{(2)}$ can be decomposed into its subalgebras, probing the extent to which $\mathcal{N} = 4$ SYM knows about the rich subalgebra structure of E_6 .

The main takeaways from this section are:

1. f_{A_2} forms a complete basis for the $\delta_{2,2}$ component of cluster polylogarithms for any cluster algebra.
2. Cluster polylogarithms can be endowed with “subalgebra” structure, where polylogarithms associated with smaller algebras are used to construct polylogarithms associated with larger algebras. This is called *subalgebra-constructibility*.
3. Subalgebra-constructibility depends intricately on the automorphism structure of cluster algebras, and only a small collection of functions are subalgebra-constructible.
4. We show that $R_7^{(2)}$ is subalgebra-constructible, and describe two novel representations which express this constructibility in terms of D_5 and A_5 subalgebras.

4.1 A_2 Functions are a Basis for Cluster Polylogarithms

A remarkable (conjectured) property of f_{A_2} is that it forms a complete basis the $\delta_{2,2}$ component of any weight-4 cluster polylogarithms for all cluster algebras³. What this means is that given a weight-4 cluster polylogarithm F that is associated with cluster algebra \mathcal{A} , there exists at least one decomposition of F of the following form:

$$F = \sum_{(x_i \rightarrow x_j) \subset \mathcal{A}} c_{ij} f_{A_2}(x_i \rightarrow x_j) + C, \quad (4.1)$$

³This has been checked for E_6 and all subalgebras contained therein.

where the sum ranges over all A_2 subalgebras of \mathcal{A} and C is a purely classical polylogarithm which also satisfies cluster adjacency on \mathcal{A} . While this decomposition may not seem remarkable at first glance, it has several key features which we highlight here:

- The classical function C can be written purely in terms of $\text{Li}_k(-\mathcal{X})$ for $k \leq 4$, where \mathcal{X} is an \mathcal{X} -coordinate of A .
- This decomposition can be determined, up to terms proportional to constants, algorithmically given only the *symbol* for F .

Specifically, given the symbol for F , the algorithm for determining eq. (4.1) is:

- Solve for c_{ij} such that $\delta_{2,2}(F - \sum c_{ij} f_{A_2}(x_i \rightarrow x_j)) = 0$, where again the sum is over all A_2 subalgebras of \mathcal{A} .
- Once the c_{ij} are fixed, the resulting function $C = F - \sum c_{ij} f_{A_2}(x_i \rightarrow x_j)$ can be fit to an ansatz of weight-4 functions composed of (products of) $\text{Li}_{k \leq 4}(-\mathcal{X})$ over all \mathcal{X} -coordinates of \mathcal{A} .

Matching beyond-the-symbol terms must be dealt with through other considerations, such as analytic behavior and correct limits. See [7] for an in-depth application of this algorithm.

In practice, then, the existence (and uniqueness) of f_{A_2} solves the problem of writing down weight-4 cluster polylogarithms. While we do not have a rigorous proof for the fact that f_{A_2} forms a basis for $\delta_{2,2}$ (beyond the explicit check through E_6), we can understand why this might be the case. Specifically, recall that the cobracket integrability condition, eq. (3.12), requires some algebraic relationships between the arguments of the cobrackets. The A_2 exchange relation

$$1 + \mathcal{X}_i = \mathcal{X}_{i-1} \mathcal{X}_{i+1} \quad (4.2)$$

is a sufficient relationship to generate one solution to integrability, namely f_{A_2} . In contrast, the $A_1 \times A_1$ cluster algebra has two algebraically distinct variables and so provides no solutions to integrability.

For any finite cluster algebra⁴, *all* algebraic relationships between cluster coordinates can be interpreted as an exchange relation for some A_2 subalgebra⁵. This is equivalent to the statement that all finite cluster algebras are composed of A_2 and $A_1 \times A_1$ subalgebras (equivalently, the associahedron is composed of squares and pentagons). Therefore, any solution to the cobracket integrability condition can be viewed as a sum of solutions deriving from different A_2 exchange relations, i.e. f_{A_2} functions. While this does not constitute a proof that f_{A_2} forms a basis for cluster polylogarithms, it highlights the suggestive connection:

- all cluster algebras can be decomposed into A_2 and $A_1 \times A_1$ subalgebras and,

⁴The infinite case will be discussed in [1].

⁵Here we are assuming that the seed cluster is composed of algebraically independent coordinates.

- all weight-4 cluster polylogarithms can be decomposed in to f_{A_2} and classical polylogarithms.

The rest of this section will be focused on exploring extensions of this idea beyond A_2 . Specifically, we'll start by exploring the space of cluster polylogarithms one can associate with the A_3 cluster algebra. Once we have these functions we can look at rank > 3 algebras and ask: which weight-4 cluster polylogarithms can be decomposed in to “ A_3 functions”? More broadly, our goal is to understand to what extent one can construct cluster polylogarithms which mimic the subalgebra structure of their associate cluster algebra. From a purely mathematical standpoint we are motivated to study this question on the strength of the f_{A_2} forming a basis. From a physics perspective we are motivated to study this question because, as we will show, $R_n^{(2)}$ exhibits this subalgebraic structure.

4.2 A_2 -constructibility

We begin bootstrapping our way up to polylogarithms associated with larger algebras by determine the space of weight-4 cluster polylogarithms associated with the A_3 cluster algebra. This is equivalent to asking: how many cluster polylogarithms can be constructed out of the A_2 subalgebras of A_3 ?

Recall that there are six A_2 subalgebras in A_3 , and as discussed in sec. (2.3), there are multiple ways to think of these subalgebras. The simplest is that these are the $\binom{6}{5}$ pentagons embeddable in a hexagon. These are also present as the six pentagonal faces in the A_3 associahedron, fig. 3. For this application it is most practical to list the six distinct A_2 seeds that appear in the 14 cluster generated by mutating on the standard A_3 seed, $x_1 \rightarrow x_2 \rightarrow x_3$:

$$x_1 \rightarrow x_2, \quad x_2 \rightarrow x_3, \quad \frac{x_2}{1+x_{1,2}} \rightarrow \frac{(1+x_1)x_3}{1+x_{1,2,3}}, \quad (4.3)$$

$$\frac{x_1x_2}{1+x_1} \rightarrow x_3, \quad x_1(1+x_2) \rightarrow \frac{x_2x_3}{1+x_2}, \quad x_1 \rightarrow x_2(1+x_3).$$

We now construct an ansatz for f_{A_3} as a sum of f_{A_2} evaluated on these six subalgebras:

$$f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3) = c_1 f_{A_2}(x_1 \rightarrow x_2) + \dots + c_6 f_{A_2}(x_1 \rightarrow x_2(1+x_3)). \quad (4.4)$$

Due to the inherited properties of f_{A_2} , the above ansatz already has a cluster-y cobracket, satisfies cluster adjacency, and is smooth and real-valued in the positive domain. The remaining problem to solve is to find values for the c_i such that f_{A_3} is invariant (up to an overall sign) under the automorphisms of A_3 . These were introduced in sec. (2.6), we write them down explicitly here:

$$\sigma_{A_3} : \quad x_1 \rightarrow \frac{x_2}{1+x_{1,2}}, \quad x_2 \rightarrow \frac{x_3(1+x_1)}{1+x_{1,2,3}}, \quad x_3 \rightarrow \frac{1+x_{1,2}}{x_1x_2x_3}, \quad (4.5)$$

$$\tau_{A_3} : \quad x_1 \rightarrow \frac{1}{x_3}, \quad x_2 \rightarrow \frac{1}{x_2}, \quad x_3 \rightarrow \frac{1}{x_1}.$$

Recall that σ is essentially the cyclic automorphism on the hexagon, and τ is the dihedral flip.

The first thing to try is to make f_{A_3} completely invariant under the A_3 automorphisms, i.e.

$$\begin{aligned}\sigma_{A_3}(f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3)) &= f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3), \\ \tau_{A_3}(f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3)) &= f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3).\end{aligned}\tag{4.6}$$

In practice this means we take the ansatz for f_{A_3} in eq. (4.4) and attempt to find c_i such that

$$\begin{aligned}f_{A_3}\left(\frac{x_2}{1+x_{1,2}} \rightarrow \frac{x_3(1+x_1)}{1+x_{1,2,3}} \rightarrow \frac{1+x_{1,2}}{x_1x_2x_3}\right) &= f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3), \\ f_{A_3}\left(\frac{1}{x_3} \rightarrow \frac{1}{x_2} \rightarrow \frac{1}{x_1}\right) &= f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3).\end{aligned}\tag{4.7}$$

However it turns out there is no collection of c_i that solves such a constraint. If instead we allow for f_{A_3} to pick up an overall minus sign when acted upon by τ_{A_3} ,

$$\begin{aligned}\sigma_{A_3}(f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3)) &= f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3), \\ \tau_{A_3}(f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3)) &= -f_{A_3}(x_1 \rightarrow x_2 \rightarrow x_3),\end{aligned}\tag{4.8}$$

then we find the solution

$$c_i = \text{constant}.\tag{4.9}$$

We label this particular solution by

$$f_{A_3}^{+-}(x_1 \rightarrow x_2 \rightarrow x_3) = f_{A_2}(x_1 \rightarrow x_2) + \dots + f_{A_2}(x_1 \rightarrow x_2(1+x_3)) = \sum_{i=1}^6 \sigma_{A_3}^i(f_{A_2}(x_1 \rightarrow x_2)),\tag{4.10}$$

where the signs indicate behavior under σ and τ , respectively. Similarly, based on the lack of a solution for eq. (4.6) we say that $f_{A_3}^{++}(x_1 \rightarrow x_2 \rightarrow x_3) = 0$. There are two remaining sign choices to check, $f_{A_3}^{-+}$ and $f_{A_3}^{--}$, and we find $f_{A_3}^{-+} = 0$ and

$$f_{A_3}^{--}(x_1 \rightarrow x_2 \rightarrow x_3) = \sum_{i=1}^6 (-1)^i \sigma_{A_3}^i(f_{A_2}(x_1 \rightarrow x_2)).\tag{4.11}$$

Therefore we see that, unlike the case of A_2 , there are two functions one can associate with A_3 : $f_{A_3}^{+-}$ and $f_{A_3}^{--}$. These functions arise purely from the interplay between the overall symmetries of the A_3 cluster algebra and the structure of the A_2 subalgebras in A_3 , i.e. there has been no physics input so far. We can tabulate our results in the following table:

	$\sigma^+\tau^+$	$\sigma^+\tau^-$	$\sigma^-\tau^+$	$\sigma^-\tau^-$
A_3	0	1	0	1

(4.12)

This displays the total number of non-classical weight-4 cluster polylogarithm functions associated with the A_3 cluster algebra with the given sign changes under the σ and τ automorphisms.

4.3 A_2 -constructibility for Larger Algebras

It is a straightforward exercise to extend the analysis of the previous section to any (finite) cluster algebra, namely

- begin with an ansatz of f_{A_2} applied across all A_2 subalgebras of an algebra \mathcal{A} ,
- impose that the ansatz is invariant under the automorphisms of \mathcal{A} , up to overall sign choices,
- count the number of solutions to these constraints.

In the case of A_n algebras we only have to impose the dihedral automorphisms σ_n and τ_n , and so the resulting table is exactly analogous to eq. (4.12):

	$\sigma^+\tau^+$	$\sigma^+\tau^-$	$\sigma^-\tau^+$	$\sigma^-\tau^-$
A_3	0	1	0	1
A_4	0	3	0	0
A_5	2	5	2	5

(4.13)

To reiterate, the above table counts the dimension of the space of cluster polylogarithms with corresponding behavior under automorphisms. For example, one can read from this table that there are only two linearly independent cluster polylogarithms on A_5 that are invariant under both σ_{A_5} and τ_{A_5} . It is interesting to note that the $+-$ sign choice admits at least one solution for all A_n studied so far, and that is the only sign choice for A_4 .

As discussed in sec. (2.6), D_4 has the automorphism group $D_4 \times S_3$, with two cyclic generators $\sigma_{D_4}^{(4)}$ and $\sigma_{D_4}^{(3)}$ (corresponding to the D_4 and S_3 , respectively) and then the D_4 flip τ_{D_4} as well as the S_3 flip denoted \mathbb{Z}_{2,D_4} . In the following table these are abbreviated $\sigma_4, \sigma_3, \tau_4, \mathbb{Z}_2$, respectively. Because of the four automorphism generators, there are 16 possible sign choices to impose on the collection of 36 distinct f_{A_2} 's in D_4 .

	$\sigma_4^+\tau_4^+$	$\sigma_4^+\tau_4^-$	$\sigma_4^-\tau_4^+$	$\sigma_4^-\tau_4^-$
	σ_3^+	σ_3^-	σ_3^+	σ_3^-
\mathbb{Z}_2^+	0	0	2	0
\mathbb{Z}_2^-	1	0	0	0

(4.14)

Here we again see that the space of functions satisfying automorphisms is remarkably constrained, with no functions exhibiting a sign flip under σ_3 and only one sign choice – $\sigma_4^+\tau_4^-\sigma_3^+\mathbb{Z}_2^+$ – that offers more than one possible solution.

D_5 is less symmetric than D_4 , with only $D_5 \times \mathbb{Z}_2$ automorphism group (in the table the generators are labeled by σ, τ , and \mathbb{Z}_2). Therefore there are 8 possible sign choices to impose on the collection of 125 distinct f_{A_2} 's in D_5 , and the resulting number of solutions is

$\sigma^+\tau^+$	$\sigma^+\tau^-$	$\sigma^-\tau^+$	$\sigma^-\tau^-$
\mathbb{Z}_2^+	\mathbb{Z}_2^-	\mathbb{Z}_2^+	\mathbb{Z}_2^-
5	0	10	0
0	3	0	7

(4.15)

Finally we turn to E_6 , which has automorphism group D_{14} with generators σ, τ , and \mathbb{Z}_2 . E_6 is much larger than the algebras studied so far, with 504 distinct A_2 subalgebras, however the space of automorphic functions is still quite constrained:

$$\begin{array}{cccc}
\begin{array}{c} \sigma^+ \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{14} \quad \boxed{16} \end{array} &
\begin{array}{c} \sigma^+ \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{23} \quad \boxed{19} \end{array} &
\begin{array}{c} \sigma^- \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{0} \end{array} &
\begin{array}{c} \sigma^- \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{0} \end{array}
\end{array} \tag{4.16}$$

Here it is very surprising that there exist no functions on E_6 which pick up a minus sign under the dihedral cycle. Recall at this point that $R_7^{(2)}$ is a cluster polylogarithm on $E_6 \simeq \text{Gr}(4, 7)$ which is invariant under all of σ, τ , and \mathbb{Z}_2 . Therefore it exists as some linear combination of the 14 degrees of freedom in $f_{E_6}^{++++}$.

4.4 A_3 -constructibility

In larger algebras there will be many instances of A_2 subalgebras joining together in to A_3 subalgebras, so it is interesting to explore the space of functions which can also exhibit this property, i.e. f_{A_2} 's grouping together in to f_{A_3} 's. In other words, having exhausted our explorations of A_2 -constructible functions, we can now turn to looking at ways in which “ A_3 functions” can be used to construct cluster polylogarithms. And since A_3 functions are themselves built out of A_2 functions, the space of A_3 -constructible functions will be a subset of our results from the previous section.

Since there are two functions associated with A_3 , $f_{A_3}^{+-}$ and $f_{A_3}^{--}$, the analysis is not quite as straightforward as the case of A_2 . To simplify matters we will start by choosing only one of the A_3 functions and using it as our basis. Future analysis could include looking at the space of functions that can be written down as a sum of both $f_{A_3}^{+-}$ and $f_{A_3}^{--}$, however currently we are aware of no function of relevance to physics with such a property. Instead, it is known that $R_7^{(2)}$ can be decomposed in to $f_{A_3}^{--}$ [7, 10] functions, and so we will focus on that function for this analysis.

The procedure is identical to that of the previous section:

- begin with an ansatz of $f_{A_3}^{--}$ applied across all A_3 subalgebras of an algebra \mathcal{A} ,
- impose that the ansatz is invariant under the automorphisms of \mathcal{A} , up to overall sign choices,
- count the number of solutions to these constraints.

Interestingly, we find that there are no $A_3^{--} \subset A_4$ or $A_3^{--} \subset D_4$ functions. For A_5 we have

$$\begin{array}{c|c|c|c|c}
& \sigma^+ \tau^+ & \sigma^+ \tau^- & \sigma^- \tau^+ & \sigma^- \tau^- \\
\hline
A_3^{--} \subset A_5 & 1 & 1 & 0 & 3
\end{array} \tag{4.17}$$

For D_5 we have

$$\begin{array}{cccc}
\begin{array}{c} \sigma^+ \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{2} \quad \boxed{0} \end{array} &
\begin{array}{c} \sigma^+ \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{3} \quad \boxed{0} \end{array} &
\begin{array}{c} \sigma^- \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{1} \end{array} &
\begin{array}{c} \sigma^- \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{5} \end{array}
\end{array} \tag{4.18}$$

And finally E_6 gives

$$\begin{array}{cccc}
\begin{array}{c} \sigma^+ \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{8} \quad \boxed{8} \end{array} &
\begin{array}{c} \sigma^+ \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{8} \quad \boxed{11} \end{array} &
\begin{array}{c} \sigma^- \tau^+ \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{0} \end{array} &
\begin{array}{c} \sigma^- \tau^- \\ \mathbb{Z}_2^+ \quad \mathbb{Z}_2^- \\ \boxed{0} \quad \boxed{0} \end{array}
\end{array} \tag{4.19}$$

5 Subalgebra-Constructibility of $R_7^{(2)}$

So far we have seen how we can bootstrap smaller cluster polylogarithms into bigger ones via automorphisms and subalgebra structures. One could continue in this vein, for example look for D_4 -constructible functions, etc. It would be interesting to understand why certain subalgebra chains produce valid cluster polylogarithms while others do not.

Rather than continuing to bootstrap our way up and explore all possibilities, we'll shift our focus to the target of $R_7^{(2)}$ and finding ways in which it can be decomposed in to functions on subalgebras of $\text{Gr}(4, 7)$ (we will use as a seed the quiver in eq. (2.44) with the nodes labeled by eq. (2.45)). This is a top-down approach for finding new and interesting cluster polylogarithms, rather than the bottom-up approach we have been describing.

The question we will answer in the upcoming sections is: for each class of subalgebra $\mathcal{A} \in E_6$, does $R_7^{(2)}$ admit a decomposition in to cluster polylogarithms of type \mathcal{A} ? In other words, is $R_7^{(2)}$ \mathcal{A} -constructible? The results of sec. (4.3) give us a menu of possible building blocks with which to construct $R_7^{(2)}$, some of which are entirely fixed while others contain free parameters. For each candidate subalgebra, we scan over all non-zero cluster polylogarithms and see if an ansatz consisting of that function evaluated on all of the relevant subalgebras of E_6 can be matched to the non-classical component of $R_7^{(2)}$. If there does exist a solution, we can then look for further decompositions of that solution in to even smaller subalgebras.

Since D_5 is the largest subalgebra of E_6 (in terms of number of clusters), we will begin there.

5.1 D_5 Decompositions of $R_7^{(2)}$

Let us now work through finding all possible D_5 decompositions of $R_7^{(2)}$. Specifically, we will

- begin with the 5-parameter solution for $f_{D_5}^{+++}$ (see eq. (4.15)),
- evaluate $f_{D_5}^{+++}$ over the 14 D_5 subalgebras of $\text{Gr}(4, 7)$,
- thus generating an ansatz with 19 free parameters: 5 inside $f_{D_5}^{+++}$ and 14 from the sum over all the subalgebras,

- then we check if there is a choice of these parameters which matches the non-classical component of $R_7^{(2)}$.

This procedure is then repeated for all non-zero sign choices for f_{D_5} , namely $f_{D_5}^{+-+}$ (10 parameters), $f_{D_5}^{-+-}$ (3 parameters), and $f_{D_5}^{---}$ (7 parameters).

There exists a solution only in the case of $f_{D_5}^{---}$. Surprisingly, two free parameters remain. These represent internal degrees of freedom in $f_{D_5}^{---}$ which cancel when summed over all D_5 subalgebras of E_6 . The solution can be characterized as

$$\begin{aligned}
f_{D_5}^{---} = \sum_{D_5^{---}} & \frac{1}{2} c_1 f_{A_2} (x_1 \rightarrow x_2 (1 + x_{34})) + (1 - c_1) f_{A_2} (x_2 \rightarrow x_3 (1 + x_4)) + \\
& \left(-\frac{1}{2} + \frac{c_1}{2} \right) f_{A_2} \left(\frac{x_1 x_2}{1 + x_1} \rightarrow x_3 (1 + x_4) \right) + \frac{1}{2} c_2 f_{A_2} (x_3 \rightarrow x_4) + \\
& (c_1 - c_2) f_{A_2} \left(\frac{x_2 x_3}{1 + x_2} \rightarrow x_4 \right) + \frac{1}{4} c_2 f_{A_2} \left(\frac{x_1 x_2 x_3}{1 + x_{12}} \rightarrow x_4 \right) + \\
& \left(\frac{1}{2} - c_1 + \frac{3c_2}{4} \right) f_{A_2} \left(x_2 (1 + x_3) \rightarrow \frac{x_3 x_4}{1 + x_3} \right)
\end{aligned} \tag{5.1}$$

where we have adopted a shorthand notation for the fully antisymmetric sum over D_5 :

$$\sum_{D_5^{---}} = \sum_{i=0}^4 \sum_{j=0}^1 \sum_{k=0}^1 (-1)^{i+j+k} \sigma_{D_5}^i \tau_{D_5}^j \mathbb{Z}_{2,D_5}^k. \tag{5.2}$$

To reiterate, the form of $f_{D_5}^{---}$ as presented in eq. (5.1) can be used as a basis to express $R_7^{(2)}$:

$$R_7^{(2)} = \sum_{D_5 \subset \text{Gr}(4,7)} f_{D_5}^{---}(x_i \rightarrow \dots) + C, \tag{5.3}$$

where the arguments of $f_{D_5}^{---}$ range over the 14 D_5 subalgebras of $\text{Gr}(4,7)$, and C represents a sum of classical polylogarithms with arguments drawn from the set of (negative) \mathcal{X} -coordinates of $\text{Gr}(4,7)$.

We can tune the remaining internal degrees of freedom, c_1 and c_2 , to embed further subalgebra structure inside $f_{D_5}^{---}$. This follows exactly the same procedure for finding D_5 -decompositions of $R_7^{(2)}$, instead now we are trying to find A_4 -, D_4 -, or A_3 -decompositions of $f_{D_5}^{---}$. It turns out that there are two different possibilities: one of which involves A_4 subalgebras and the other involving A_3 . We will review these decompositions now.

Setting

$$c_2 = -\frac{6}{5} + \frac{8}{5} c_1 \tag{5.4}$$

and leaving c_1 free makes $f_{D_5}^{---}$ an A_3^{--} -constructible function:

$$\begin{aligned}
f_{A_3 \subset D_5}^{---} = \frac{1}{10} \sum_{D_5^{---}} & (-3 + 4c_1) f_{A_3}^{--} (x_1 \rightarrow x_2 \rightarrow x_3 (1 + x_4)) + (6 - 3c_1) f_{A_3}^{--} \left(x_1 \rightarrow x_2 (1 + x_3) \rightarrow \frac{x_3 x_4}{1 + x_3} \right) \\
& + (2 - c_1) f_{A_3}^{--} (x_2 \rightarrow x_3 (1 + x_4) \rightarrow x_5) + \left(\frac{1}{2} - \frac{3c_1}{2} \right) f_{A_3}^{--} \left(\frac{1}{x_4} \rightarrow x_3 (1 + x_4) \rightarrow x_5 \right) \\
& + \left(-1 + \frac{c_1}{2} \right) f_{A_3}^{--} \left(\frac{1}{x_4} \rightarrow \frac{x_2 x_3 (1 + x_4)}{1 + x_2} \rightarrow x_5 \right). \tag{5.5}
\end{aligned}$$

There is no A_3^{+-} representation. Setting

$$c_1 = \frac{3}{5}, \quad c_2 = 0, \tag{5.6}$$

makes $f_{D_5}^{---}$ an A_4^{+-} -constructible function:

$$f_{A_4 \subset D_5}^{---} = \sum_{D_5^{---}} f_{A_4}^{+-} (x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4) \tag{5.7}$$

where we have used a new function

$$f_{A_4}^{+-} = \frac{1}{40} \sum_{A_4^{+-}} 3f_{A_2} (x_1 \rightarrow x_2 (1 + x_3)) - f_{A_2} (x_2 \rightarrow x_3 (1 + x_4)). \tag{5.8}$$

Here we have used the notation $\sum_{A_4^{+-}}$ to denote the skew-dihedral sum over A_4 ,

$$\sum_{A_4^{+-}} = \sum_{i=0}^6 \sum_{j=0}^1 (-1)^j \sigma_{A_4}^i \tau_{A_4}^j. \tag{5.9}$$

$f_{A_4}^{+-}$ does not have an A_3 -decomposition.

The $A_4 \subset D_5$ representation of $R_7^{(2)}$ in particular is quite remarkable, and can be described as:

- evaluate $f_{A_4}^{+-}$ on all A_4 subalgebras of $\text{Gr}(4, 7)$,
- add them together so that they are antisymmetric under the automorphisms of the D_5 subalgebras of $\text{Gr}(4, 7)$,
- and the resulting function matches the non-classical portion of $R_7^{(2)}$.

5.2 A_5 Decompositions of $R_7^{(2)}$

We now turn to the other corank-1 subalgebra of E_6 , A_5 . The procedure for finding decompositions of $R_7^{(2)}$ in terms of A_5 subalgebras is identical to the previous section.

As was the case with D_5 , only the fully antisymmetric case $f_{A_5}^{--}$ generates a representation of $R_7^{(2)}$. This time there is one internal degree of freedom which cancels in the sum over the 7 A_5 subalgebras of $\text{Gr}(4, 7)$:

$$f_{A_5}^{--} = \sum_{A_5^{--}} \frac{1}{2} c_1 f_{A_2}(x_2 \rightarrow x_3(1+x_4)) + (-1-c_1) f_{A_2}(x_2 \rightarrow x_3(1+x_{45})) \\ + \left(-\frac{1}{2} - c_1\right) f_{A_2}\left(\frac{x_1 x_2}{1+x_1} \rightarrow x_3(1+x_4)\right) \quad (5.10)$$

Following in complete analogy with the D_5 case, we can tune c_1 to decompose $f_{A_5}^{--}$ into either A_3 or A_4 functions. Unlike D_5 , the A_3 representation employs the skew-symmetric A_3 function, $f_{A_3}^{+-}$. Setting $c_1 = -\frac{1}{2}$ gives

$$f_{A_3 \subset A_5}^{--} = -\frac{1}{160} \sum_{A_5^{--}} f_{A_3}^{+-} \left(x_2 \rightarrow x_3(1+x_4) \rightarrow \frac{x_4 x_5}{1+x_4} \right). \quad (5.11)$$

The overall coefficient of $-\frac{1}{160}$ is simply a result of overcounting in the symmetric sum, as that A_3 has only 4 automorphic partners in A_5 .

Setting $c_1 = -1$ makes $f_{A_5}^{--}$ an A_4 -constructible function:

$$f_{A_4 \subset A_5}^{--} = -\frac{1}{20} \sum_{A_5^{--}} f_{A_4}^{+-}(x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4). \quad (5.12)$$

Remarkably, this is the same $f_{A_4}^{+-}$ function defined for the D_5 function in eq. (5.8). In other words, we have **[temporarily ignoring the overall symmetry factors, which we can mess with later]**

$$R_7^{(2)} = \sum_{D_5 \subset \text{Gr}(4,7)} \sum_{A_4 \subset D_5^{--}} f_{A_4}^{+-}(x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4) + C \quad (5.13)$$

$$= \sum_{A_5 \subset \text{Gr}(4,7)} \sum_{A_4 \subset A_5^{--}} f_{A_4}^{+-}(x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4) + C \quad (5.14)$$

where C is again a function of classical polylogarithms and is the same for both the D_5 and A_5 representations.

6 Conclusion

1. since there are so few D_5 and A_5 subalgebras of $\text{Gr}(4,7)$, these decompositions distill more intrinsic (better word?) geometric structure than the A_3 function
2. would be interesting to find the D_5 decomposition of higher-point amplitudes, to see if this seter our determined constants

3. not possible to get the full (function-level) amplitude out of a single (function-level) subalgebra function; it merits checking whether a linear combination of different subalgebra functions could be found
4. bring up mutations as active vs passive transformations?

Acknowledgements

We thank Jacob Bourjaily, James Drummond, Marcus Spradlin, Henriette Elvang, ... for illuminating discussions. The authors are grateful to the Kavli Institute for Theoretical Physics (National Science Foundation grant NSF PHY11-25915) for hospitality.

A Integrability and Adjacency for A_2

B Counting Subalgebras of Finite Cluster Algebras

In this appendix we catalog the subalgebra structure for the finite connected cluster algebras $\subseteq E_6$. These algebras are: $A_2, A_3, A_4, D_4, A_5, D_5, E_6$.

When counting distinct subalgebras, we lump together all subalgebras which are labeled by the same mutable nodes. However the same subalgebra may appear multiple times but dressed by different frozen nodes – these appear as distinct subpolytopes in the full associahedra. We have included the counts for both subpolytopes and distinct subalgebras. Also note that in our counting for \mathcal{X} -coordinates we have included both x and $1/x$.

A_2 : clusters: 5 a -coordinates: 5 x -coordinates: 10

A_3 : clusters: 14 a -coordinates: 9 x -coordinates: 30

Type	Subpolytopes	Subalgebras
A_2	6	6
$A_1 \times A_1$	3	3

A_4 : clusters: 42 a -coordinates: 14 x -coordinates: 70

Type	Subpolytopes	Subalgebras
A_2	28	21
$A_1 \times A_1$	28	28
A_3	7	7
$A_2 \times A_1$	7	7
$A_1 \times A_1 \times A_1$	0	0

D_4 : clusters: 50 a -coordinates: 16 x -coordinates: 104

Type	Subpolytopes	Subalgebras
A_2	36	36
$A_1 \times A_1$	30	18
A_3	12	12
$A_2 \times A_1$	0	0
$A_1 \times A_1 \times A_1$	4	4

A_5 : clusters: 132 a -coordinates: 20 x -coordinates: 140

Type	Subpolytopes	Subalgebras
A_2	120	56
$A_1 \times A_1$	180	144
A_3	36	28
$A_2 \times A_1$	72	72
$A_1 \times A_1 \times A_1$	12	12
D_4	0	0
A_4	8	8
$A_3 \times A_1$	8	8
$A_2 \times A_2$	4	4
$A_2 \times A_1 \times A_1$	0	0
$A_1 \times A_1 \times A_1 \times A_1$	0	0

D_5 : clusters: 182 a -coordinates: 25 x -coordinates: 260

Type	Subpolytopes	Subalgebras
A_2	180	125
$A_1 \times A_1$	230	145
A_3	70	65
$A_2 \times A_1$	60	50
$A_1 \times A_1 \times A_1$	30	30
D_4	5	5
A_4	10	10
$A_3 \times A_1$	5	5
$A_2 \times A_2$	0	0
$A_2 \times A_1 \times A_1$	5	5
$A_1 \times A_1 \times A_1 \times A_1$	0	0

E_6 : clusters: 833 a -coordinates: 42 x -coordinates: 770

Type	Subpolytopes	Subalgebras
A_2	1071	504
$A_1 \times A_1$	1785	833
A_3	476	364
$A_2 \times A_1$	714	490
$A_1 \times A_1 \times A_1$	357	357
D_4	35	35
A_4	112	98
$A_3 \times A_1$	112	112
$A_2 \times A_2$	21	14
$A_2 \times A_1 \times A_1$	119	119
$A_1 \times A_1 \times A_1 \times A_1$	0	0
D_5	14	14
A_5	7	7
$D_4 \times A_1$	0	0
$A_4 \times A_1$	14	14
$A_3 \times A_2$	0	0
$A_3 \times A_1 \times A_1$	0	0
$A_2 \times A_2 \times A_1$	7	7
$A_2 \times A_1 \times A_1 \times A_1$	0	0
$A_1 \times A_1 \times A_1 \times A_1 \times A_1$	0	0

C Cobracket Spaces in Finite Cluster Algebras

References

- [1] L. J. Dixon, J. M. Drummond, M. von Hippel, and J. Pennington, “Hexagon functions and the three-loop remainder function,” *JHEP*, vol. 1312, p. 049, 2013.
- [2] L. J. Dixon, J. M. Drummond, C. Duhr, and J. Pennington, “The four-loop remainder function and multi-Regge behavior at NNLLA in planar $\mathcal{N} = 4$ super-Yang-Mills theory,” *JHEP*, vol. 1406, p. 116, 2014.
- [3] J. M. Drummond, G. Papathanasiou, and M. Spradlin, “A Symbol of Uniqueness: The Cluster Bootstrap for the 3-Loop MHV Heptagon,” *JHEP*, vol. 03, p. 072, 2015.
- [4] S. Caron-Huot, L. J. Dixon, A. McLeod, and M. von Hippel, “Bootstrapping a Five-Loop Amplitude Using Steinmann Relations,” *Phys. Rev. Lett.*, vol. 117, no. 24, p. 241601, 2016.
- [5] L. J. Dixon, J. Drummond, T. Harrington, A. J. McLeod, G. Papathanasiou, and M. Spradlin, “Heptagons from the Steinmann Cluster Bootstrap,” *JHEP*, vol. 02, p. 137, 2017.
- [6] S. Caron-Huot, “Superconformal symmetry and two-loop amplitudes in planar $\mathcal{N} = 4$ super Yang-Mills,” *JHEP*, vol. 1112, p. 066, 2011.

- [7] J. Golden and M. Spradlin, “An analytic result for the two-loop seven-point MHV amplitude in $\mathcal{N} = 4$ SYM,” *JHEP*, vol. 1408, p. 154, 2014.
- [8] J. Golden, A. B. Goncharov, M. Spradlin, C. Vergu, and A. Volovich, “Motivic Amplitudes and Cluster Coordinates,” *JHEP*, vol. 1401, p. 091, 2014.
- [9] J. Golden and M. Spradlin, “The differential of all two-loop MHV amplitudes in $\mathcal{N} = 4$ Yang-Mills theory,” *JHEP*, vol. 1309, p. 111, 2013.
- [10] J. Golden, M. F. Paulos, M. Spradlin, and A. Volovich, “Cluster Polylogarithms for Scattering Amplitudes,” *J. Phys.*, vol. A47, no. 47, p. 474005, 2014.
- [11] J. Golden and M. Spradlin, “A Cluster Bootstrap for Two-Loop MHV Amplitudes,” *JHEP*, vol. 02, p. 002, 2015.
- [12] J. Drummond, J. Foster, and O. Gurdogan, “Cluster adjacency properties of scattering amplitudes,” 2017.
- [13] J. Golden, A. J. McLeod, M. Spradlin, A. Volovich, in progress.
- [14] S. Caron-Huot and S. He, “Jumpstarting the All-Loop S-Matrix of Planar $\mathcal{N} = 4$ Super Yang-Mills,” *JHEP*, vol. 1207, p. 174, 2012.
- [15] L. J. Dixon and M. von Hippel, “Bootstrapping an NMHV amplitude through three loops,” *JHEP*, vol. 1410, p. 65, 2014.
- [16] L. J. Dixon, M. von Hippel, and A. J. McLeod, “The four-loop six-gluon NMHV ratio function,” *JHEP*, vol. 01, p. 053, 2016.
- [17] J. M. Drummond, J. M. Henn, and J. Trnka, “New differential equations for on-shell loop integrals,” *JHEP*, vol. 04, p. 083, 2011.
- [18] J. L. Bourjaily, A. J. McLeod, M. von Hippel, and M. Wilhelm, “Rationalizing Loop Integration,” 2018.
- [19] J. Henn, E. Herrmann, and J. Parra-Martinez, “Bootstrapping two-loop Feynman integrals for planar $\mathcal{N}=4$ sYM,” 2018.
- [20] S. Caron-Huot, L. J. Dixon, M. von Hippel, A. J. McLeod, and G. Papathanasiou, “The Double Pentagonal Integral to All Orders,” 2018.
- [21] N. Arkani-Hamed, J. L. Bourjaily, F. Cachazo, A. B. Goncharov, A. Postnikov, *et al.*, “Scattering Amplitudes and the Positive Grassmannian,” 2012.
- [22] I. Prlina, M. Spradlin, J. Stankowicz, S. Stanojevic, and A. Volovich, “All-Helicity Symbol Alphabets from Unwound Amplituhedra,” *JHEP*, vol. 05, p. 159, 2018.
- [23] S. Fomin and A. Zelevinsky, “Cluster algebras I: Foundations,” *J. Am. Math. Soc.*, vol. 15, no. 2, pp. 497–529, 2002.
- [24] V. V. Fock and A. B. Goncharov, “Cluster ensembles, quantization and the dilogarithm,” *Ann. Sci. Éc. Norm. Supér. (4)*, vol. 42, no. 6, pp. 865–930, 2009.
- [25] J. S. Scott, “Grassmannians and cluster algebras,” *Proc. Lond. Math. Soc., III. Ser.*, vol. 92, no. 2, pp. 345–380, 2006.
- [26] Z. Bern, L. J. Dixon, and V. A. Smirnov, “Iteration of planar amplitudes in maximally supersymmetric Yang-Mills theory at three loops and beyond,” *Phys. Rev.*, vol. D72, p. 085001,

2005.

- [27] J. Drummond, J. Henn, G. Korchemsky, and E. Sokatchev, “Conformal Ward identities for Wilson loops and a test of the duality with gluon amplitudes,” *Nucl.Phys.*, vol. B826, pp. 337–364, 2010.
- [28] Z. Bern, L. Dixon, D. Kosower, R. Roiban, M. Spradlin, *et al.*, “The Two-Loop Six-Gluon MHV Amplitude in Maximally Supersymmetric Yang-Mills Theory,” *Phys.Rev.*, vol. D78, p. 045007, 2008.
- [29] J. Drummond, J. Henn, G. Korchemsky, and E. Sokatchev, “Hexagon Wilson loop = six-gluon MHV amplitude,” *Nucl.Phys.*, vol. B815, pp. 142–173, 2009.
- [30] J. M. Drummond, J. Henn, V. A. Smirnov, and E. Sokatchev, “Magic identities for conformal four-point integrals,” *JHEP*, vol. 01, p. 064, 2007.
- [31] Z. Bern, M. Czakon, L. J. Dixon, D. A. Kosower, and V. A. Smirnov, “The Four-Loop Planar Amplitude and Cusp Anomalous Dimension in Maximally Supersymmetric Yang-Mills Theory,” *Phys. Rev.*, vol. D75, p. 085010, 2007.
- [32] Z. Bern, J. Carrasco, H. Johansson, and D. Kosower, “Maximally supersymmetric planar Yang-Mills amplitudes at five loops,” *Phys.Rev.*, vol. D76, p. 125020, 2007.
- [33] L. F. Alday and J. M. Maldacena, “Gluon scattering amplitudes at strong coupling,” *JHEP*, vol. 0706, p. 064, 2007.
- [34] J. M. Drummond, J. Henn, G. P. Korchemsky, and E. Sokatchev, “Dual superconformal symmetry of scattering amplitudes in N=4 super-Yang-Mills theory,” *Nucl. Phys.*, vol. B828, pp. 317–374, 2010.
- [35] S. Fomin and A. Zelevinsky, “Cluster algebras II: Finite type classification,” *Invent. Math.*, vol. 154, no. 1, pp. 63–121, 2003.
- [36] D. Parker, A. Scherlis, M. Spradlin, and A. Volovich, “Hedgehog Bases for A_n Cluster Polylogarithms and An Application to Six-Point Amplitudes,” *JHEP*, vol. 11, p. 136, 2015.
- [37] W. Chang and B. Zhu, “Cluster automorphism groups of cluster algebras of finite type,” 2015.
- [38] K.-T. Chen, “Iterated path integrals,” *Bull. Amer. Math. Soc.*, vol. 83, no. 5, pp. 831–879, 1977.
- [39] F. C. Brown, “Multiple zeta values and periods of moduli spaces $\overline{\mathcal{M}}_{0,n}(\mathbb{R})$,” *Annales Sci.Ecole Norm.Sup.*, vol. 42, p. 371, 2009.
- [40] A. B. Goncharov, “A simple construction of Grassmannian polylogarithms,” *Adv. Math.*, vol. 241, pp. 79–102, 2013.
- [41] S. Caron-Huot, L. J. Dixon, M. von Hippel, A. J. McLeod, G. Papathanasiou, in progress.
- [42] C. Duhr, H. Gangl, and J. R. Rhodes, “From polygons and symbols to polylogarithmic functions,” *JHEP*, vol. 1210, p. 075, 2012.
- [43] C. Duhr, “Mathematical aspects of scattering amplitudes,” 2014.
- [44] F. Brown, “On the decomposition of motivic multiple zeta values,” 2011.
- [45] L. F. Alday, D. Gaiotto, and J. Maldacena, “Thermodynamic Bubble Ansatz,” *JHEP*, vol. 09, p. 032, 2011.

- [46] G. Yang, “A simple collinear limit of scattering amplitudes at strong coupling,” *JHEP*, vol. 03, p. 087, 2011.
- [47] N. Beisert, B. Eden, and M. Staudacher, “Transcendentality and Crossing,” *J. Stat. Mech.*, vol. 0701, p. P01021, 2007.
- [48] O. Steinmann, “Über den Zusammenhang zwischen den Wightmanfunktionen und der retardierten Kommutatoren,” *Helv. Physica Acta*, vol. 33, p. 257, 1960.
- [49] O. Steinmann, “Wightman-Funktionen und retardierten Kommutatoren. II,” *Helv. Physica Acta*, vol. 33, p. 347, 1960.
- [50] K. E. Cahill and H. P. Stapp, “Optical Theorems and Steinmann Relations,” *Annals Phys.*, vol. 90, p. 438, 1975.
- [51] G. Yang, “Scattering amplitudes at strong coupling for 4K gluons,” *JHEP*, vol. 12, p. 082, 2010.