Three Problems on Grothendieck Polynomials and Pipedreams

Chen-An Jack Chou

Advisor: Professor Brendon Rhoades

Department of Mathematics

University of California San Diego

In partial fulfillment of the requirements

for the Mathematics Honors Program

June, 2024

Acknowledgment

I would like to thank Professor Brendon Rhoades for advising me in this thesis and teaching me various topics in algebraic combinatorics. I would like to thank my collaborator Tianyi Yu for giving me the opportunity to do research with him, and this thesis would not be possible without his guidance. I would like to thank all my family and friends for supporting me along the way. Lastly, I would like to thank my mother, whom I will miss forever, for everything.

Abstract

Pipedreams are combinatorial objects that compute Grothendieck polynomials. In this thesis, we study the support of Grothendieck polynomials using pipedreams and answer three questions. First, we provide an explicit algorithm that constructs the unique maximal pipedream of the leading monomial for any double Grothendieck polynomial, resolving a frustration of Pechenik, Speyer, and Weigandt. Then, we introduce the first direct combinatorial formula for the top degree components of Grothendieck polynomials. Finally, we prove the inverse fireworks case of a conjecture of Mészáros, Setiabrata, and St. Dizier on the support of Grothendieck polynomials as an application of our new combinatorial object.

Contents

1. Introduction

Some of the most natural things we can ask in enumerative geometry are questions like "how many lines intersect this set of lines in 3D space?". Making enumerations like these rigorous was the goal of Hilbert's 15^{th} Problem, and the associated program of Schubert calculus.

In this thesis, we introduce three types of polynomials in Schubert calculus that are all labeled by permutations. For $w \in S_n$, Lascoux and Schützenberger [LS82] introduced the Schubert polynomial $\mathfrak{S}_w(\mathbf{x})$. They are generalizations of the classical Schur polynomials and they represent the cohomology classes of Schubert varieties in the flag variety. They also introduced Grothendieck polynomials $\mathfrak{G}_w(\mathbf{x})$, which are explicit polynomial representatives of the K-classes of structure sheaves of Schubert varieties in flag varieties. In general, Grothendieck polynomials are not homogeneous. Their lowest degree homogeneous components recover the Schubert polynomials. Lastly, double Grothendieck polynomials are generalizations of Grothendieck polynomials in variables $x_1, \ldots, x_n, y_1, \ldots, y_n$; they represent Schubert classes in the torus-equivariant K-theory of flag varieties. All three polynomials can be computed using combinatorial objects called pipedreams [BB93, BJS93, FK94], which are certain diagrams under a staircase tiled with crossings and bumps. Each $w \in S_n$ is associated with a set of pipedreams, denoted as $\mathsf{PD}(w)$. We may compute the monomial supports of our polynomials by assigning certain weights to the tiles in each pipedream in $\mathsf{PD}(w)$.

The matrix Schubert variety X_w is a determinantal variety that has been studied extensively (see for instance [FUL92, KM05, KMY09, WY18]). Castelnuovo–Mumford regularity measures the algebraic complexity of varieties. Since matrix Schubert varieties are Cohen–Macaulay [FUL92, KM05, Ram85], the Castelnuovo–Mumford regularity of X_w is the difference between the top and bottom degree of its K-polynomial. By the work of Knutson and Miller [KM05], the K-polynomial of X_w is the Grothendieck polynomial $\mathfrak{G}_w(\mathbf{x})$. Consequently, determining the Castelnuovo–Mumford regularity of X_w reduces to computing the degree of $\mathfrak{G}_w(\mathbf{x})$. With this motivation, there has been a recent surge in the study of top degree components of $\mathfrak{G}_w(\mathbf{x})$ [DMSD22, Haf22, PSW21, PY23, RRR+21, RRW23], which we denote as $\widehat{\mathfrak{G}}_w(\mathbf{x})$.

Pechenik, Speyer, and Weigandt [PSW21] defined a statistic $\mathsf{rajcode}(\cdot)$ on S_n using increasing subsequences of permutations. They showed that $x^{\mathsf{rajcode}(w)}y^{\mathsf{rajcode}(w^{-1})}$ is the leading monomial in the top degree components of the double Grothendieck polynomials $\mathfrak{G}_w(\mathbf{x}, \mathbf{y})$ with respect to the lexicographical order where $x_n > \cdots > x_1$ and $y_n > \cdots > y_1$. However, in Remark 7.2, they said:

> "We find it frustrating that we do not have a direct recipe for the maximal pipe dream in terms of w."

We relieve their frustration in Section 4 by providing an explicit algorithm that construct the maximal pipedream with row weight $\mathsf{rajcode}(w)$ and column weight $\mathsf{rajcode}(w^{-1})$.

Furthermore, it is conjectured by Mészáros, Setiabrata, and St. Dizier [MSSD22] that $\widehat{\mathfrak{G}}_w(\mathbf{x})$ governs the support of $\mathfrak{G}_w(\mathbf{x})$. Pechenik, Speyer, and Weigandt [PSW21] showed that any $\widehat{\mathfrak{G}}_w(\mathbf{x})$ is equal to $\widehat{\mathfrak{G}}_u(\mathbf{x})$ for some inverse fireworks u. In Section 5, we provide the first direct combinatorial formula of $\widehat{\mathfrak{G}}_w(\mathbf{x})$ when w is inverse fireworks. We remove certain pipes from a pipedream to obtain a novel combinatorial object called a marked verticalless pipedream (see Definition 5.4). In other words, we introduce a set $\mathsf{MVPD}(w)$ and a weight preserving bijection $\Phi : \mathsf{PD}(w) \to \mathsf{MVPD}(w)$. Consequently, each $M \in \mathsf{MVPD}(w)$ is associated with a monomial $\mathsf{wt}_M(\mathbf{x})$ which agrees with its corresponding pipedream. We then classify the highest weight MVPDs, which we denote as $\widehat{\mathsf{MVPD}(w)}$, when w is inverse fireworks. We then biject them with a similar combinatorial object which we call bumpless vertical-less pipedreams (BVPDs), obtaining our formula for $\widehat{\mathfrak{G}}_w(\mathbf{x})$.

Finally, we provide an application of MVPDs in Section 6 by proving the inverse fireworks case of the following conjecture by Mészáros, Setiabrata, and St. Dizier on the support Grothendick polynomials.

Conjecture 1.1. [MSSD22, Conjecture 1.2] *If $\alpha \in \mathsf{Supp}(\mathfrak{G}_w)$ and $|\alpha| < \deg(\mathfrak{G}_w)$, then there exists $i \in [n]$ such that $x_i\alpha \in \mathsf{Supp}(\mathfrak{G}_w)$.*

Our proof is constructive, for any $P \in \mathsf{PD}(w)$ with weight α, we construction $P' \in \mathsf{PD}(w)$ with the desired weight.

The content of this thesis is mainly from [CY23] and [CY24].

2. Polynomials in Schubert calculus

2.1. **Permutation.** Polynomials in Schubert calculus are often labeled by permutations. We start off with some definitions related to permutations. Let $[n]$ denote the set $\{1, 2, \ldots, n\}$ and S_n be the symmetric group of n objects.

Definition 2.1. A *permutation* $w \in S_n$ is a bijection from $[n]$ to itself. We write w in *one-line notation* as the string $w(1)\, w(2)\, \ldots\, w(n)$.

Example 2.2. The permutation $w = 521463$ is the bijection in S_6 that sends $1 \to 5$, $2 \to 2$, $3 \to 1$, $4 \to 4$, $5 \to 6$, and $6 \to 3$.

Definition 2.3. The *inversion set* of $w \in S_n$ is

$$\mathsf{Inv}(w) := \{(i, j) : i < j, w(i) > w(j)\}$$

The *length* of w is $\ell(w) := |\mathsf{Inv}(w)|$. Elements in $\mathsf{Inv}(w)$ are called *inversions*. The weak decomposition that counts the number of inversions for each i is the *invcode* or *Lehmer code*.

Example 2.4. For $w = 521463$, we have

$$\mathsf{Inv}(w) = \{(1, 2), (1, 3), (1, 4), (1, 6), (2, 3), (4, 6), (5, 6)\},$$

$\ell(w) = 7$, and $\mathsf{invcode}(w) = (4, 1, 0, 1, 1, 0)$.

Definition 2.5. The *long element*, denoted w_0, is the permutation in S_n with the longest length, or the most inversions. It has one-line notation $n\, n-1\, \cdots\, 2\, 1$.

We then introduce inverse fireworks permutations, a special family of permutations which we will study in Section 5 and 6.

Definition 2.6. A permutation is *fireworks* if the start of each decreasing run is increasing.

Example 2.7. The permutation $w = 154263$ is fireworks since its decreasing runs 1, 542, and 63 have their starting number in increasing order $1 < 5 < 6$. The permutation $w = 164253$ is not fireworks since its decreasing runs 1, 642, and 53 do not have their starting number in increasing order.

Fireworks permutations are also known as $3 - 12$ avoiding permutations.

Definition 2.8. A permutation is *inverse fireworks* if its inverse is fireworks.

Example 2.9. The permutation $w = 146325$ is inverse fireworks since its inverse $w^{-1} = 154263$ is fireworks.

Proposition 2.10. [PSW21] *Inverse fireworks permutations in S_n are enumerated by the n^{th} Bell number, which is the number of set partitions of $[n]$.*

Proof. For each set partition of $[n]$, we order the blocks in increasing order of their largest number, and order the numbers in each block in decreasing order. This gives a natural bijection between set partitions and fireworks permutations, which bijects with inverse fireworks permutations. $\square$

2.2. **Schubert polynomial.** We now define Schubert polynomials, which were first introduced by Lascoux and Schützenberger [LS82]. Schubert polynomials represent the cohomology classes of Schubert varieties in the flag variety. They can be defined recursively using the divided difference operator.

Definition 2.11. Consider the ring $\mathbb{Z}[x_1, \ldots, x_n]$ and the group action of S_n that permutes the variables by their subscripts. For $i \in [n]$, the *divided difference operator* ∂_i is defined as follow:

$$\partial_i(f) := \frac{f - s_i(f)}{x_i - x_{i+1}}, \qquad \forall f \in \mathbb{Z}[x_1, \ldots, x_n]$$

where s_i is the adjacent transposition in S_n that swaps i and $i+1$.

They are called the divided difference operators because $\partial_i(f)$ is symmetric in x_i and x_{i+1}. Furthermore, $\partial_i(f) = 0$ if and only if f is already symmetric in x_i and x_{i+1}.

Definition 2.12. The *Schubert polynomials* $\mathfrak{S}_w$ are defined as follow:

$$\mathfrak{S}_w := \begin{cases} x_1^{n-1} x_2^{n-2} \cdots x_{n-2}^2 x_{n-1}^1 & w = w_0 \\ \partial_i(\mathfrak{S}_{ws_i}) & w(i) < w(i+1) \end{cases}$$

Example 2.13. We calculate $\mathfrak{S}_{312}$ by definition.

$$\begin{aligned} \mathfrak{S}_{312} &= \partial_2(\mathfrak{S}_{312 \cdot s_2}) = \partial_2(\mathfrak{S}_{321}) = \partial_2(x_1^2 x_2) \\ &= \frac{x_1^2 x_2 - x_1^2 x_3}{x_2 - x_3} = \frac{x_1^2(x_2 - x_3)}{x_2 - x_3} = x_1^2 \end{aligned}$$

Similarly, we can calculate the Schubert polynomials for all the permutations in S_3 and form the following diagram:

$$\begin{tikzcd} & \mathfrak{S}_{321} = x_1^2 x_2 \arrow[dl, "\partial_1"'] \arrow[dr, "\partial_2"] & \\ \mathfrak{S}_{231} = x_1 x_2 \arrow[d, "\partial_2"'] & & \mathfrak{S}_{312} = x_1^2 \arrow[d, "\partial_1"] \\ \mathfrak{S}_{213} = x_1 \arrow[dr, "\partial_1"'] & & \mathfrak{S}_{132} = x_1 + x_2 \arrow[dl, "\partial_2"] \\ & \mathfrak{S}_{123} = 1 & \end{tikzcd}$$

2.3. **Grothendieck polynomial.** Grothendieck polynomials, also introduced by Lascoux and Schützenberger [LS82], are generalizations of Schubert polynomials. They are the K-theoritic analogs of Schubert polynomials and they represent the classes of the structure sheaves of Schubert varieties in K-theory. They can also be defined recursively using a similar operator.

Definition 2.14. For $i \in [n]$, we define a variation of the divided difference operator, denoted as $\overline{\partial_i}$, as follow:

$$\overline{\partial_i}(f) := \partial_i(f - x_{i+1} f), \qquad \forall f \in \mathbb{Z}[x_1, \ldots, x_n]$$

Definition 2.15. The *Grothendieck polynomials* $\mathfrak{G}_w(\boldsymbol{x})$ are defined as follow:

$$\mathfrak{G}_w(\mathbf{x}) := \begin{cases} x_1^{n-1}x_2^{n-2}\cdots x_{n-2}^2 x_{n-1}^1 & w = w_0 \\ \overline{\partial_i}(\mathfrak{G}_{ws_i}(\mathbf{x})) & w(i) < w(i+1) \end{cases}$$

Example 2.16. We calculate $\mathfrak{G}_{312}$ and $\mathfrak{G}_{132}$ by definition.

$$\begin{aligned} \mathfrak{G}_{312} &= \overline{\partial_2}(\mathfrak{G}_{312\cdot s_2}) = \overline{\partial_2}(\mathfrak{G}_{321}) = \overline{\partial_2}(x_1^2x_2) = \partial_2(x_1^2x_2 - x_1^2x_2x_3) \\ &= \frac{x_1^2x_2 - x_1^2x_2x_3 - x_1^2x_3 + x_1^2x_2x_3}{x_2 - x_3} = \frac{x_1^2(x_2 - x_3)}{x_2 - x_3} = x_1^2 \\ \mathfrak{G}_{132} &= \overline{\partial_1}(\mathfrak{G}_{132\cdot s_1}) = \overline{\partial_1}(\mathfrak{G}_{312}) = \overline{\partial_1}(x_1^2) = \partial_1(x_1^2 - x_1^2x_2) \\ &= \frac{x_1^2 - x_1^2x_2 - x_2^2 - x_1x_2^2}{x_1 - x_2} = \frac{(x_1 - x_2)(x_1 + x_2) - x_1x_2(x_1 - x_2)}{x_1 - x_2} \\ &= x_1 + x_2 - x_1x_2 \end{aligned}$$

Similarly, we can calculate the Grothendieck polynomials for all the permutations in S_3 and form the following diagram:

$$\begin{tikzcd} & \mathfrak{G}_{321} = x_1^2x_2 \arrow[dl, "\overline{\partial_1}"'] \arrow[dr, "\overline{\partial_2}"] & \\ \mathfrak{G}_{231} = x_1x_2 \arrow[d, "\overline{\partial_2}"'] & & \mathfrak{G}_{312} = x_1^2 \arrow[d, "\overline{\partial_1}"] \\ \mathfrak{G}_{213} = x_1 \arrow[dr, "\overline{\partial_1}"'] & & \mathfrak{G}_{132} = x_1 + x_2 - x_1x_2 \arrow[dl, "\overline{\partial_2}"] \\ & \mathfrak{G}_{123} = 1 & \end{tikzcd}$$

In general, Grothendieck polynomials are not homogeneous. The lowest degree homogeneous part of a Grothendieck polynomial $\mathfrak{G}_w$ is the respective Schubert polynomial $\mathfrak{S}_w$.

Double Grothendieck polynomials are further generalizations of Grothendieck polynomials in $2n$ variables labeled $x_1, \ldots, x_n, y_1, \ldots, y_n$. They represent Schubert classes in the torus-equivariant K-theory of flag varieties.

Definition 2.17. The *double Grothendieck polynomials* $\mathfrak{G}(\boldsymbol{x}, \boldsymbol{y})$ are defined as follow:

$$\mathfrak{G}_w(\mathbf{x}, \mathbf{y}) := \begin{cases} \prod_{i+j\leqslant n}(x_i + y_j - x_iy_j) & w = w_0 \\ \overline{\partial_i}(\mathfrak{S}_{ws_i}(\mathbf{x}, \mathbf{y})) & w(i) < w(j+1) \end{cases}$$

Double Grothendieck polynomials specializes to Grothendieck polynomials when we set $y_1 = y_2 = \cdots = y_n = 0$. Equivalently, $\mathfrak{G}_w(\mathbf{x}, 0) = \mathfrak{G}_w(\mathbf{x})$.

In the next section, we introduce combinatorial objects that allow us to calculate $\mathfrak{G}_w(\mathbf{x})$ without using the divided difference operators.

3. Combinatorial Objects

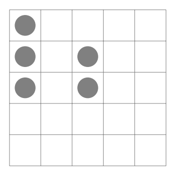
We study the polynomials in Section 2 through combinatorial objects called pipedreams [BB93, BJS93, FK94, KM05] and bumpless pipedreams [LLS21, LLS23].

3.1. Rothe diagram. We start off by representing permutations with diagrams. For $w \in S_n$, we can find a pipedream $P \in \mathsf{PD}(w)$ through its Rothe diagram $\mathsf{Rothe}(w)$, which encodes the inversions of w.

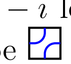
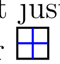
Definition 3.1. For $w \in S_n$, we define its *Rothe diagram* $\mathsf{Rothe}(w)$ as the following subset of cells in an $n \times n$ grid

$$\mathsf{Rothe}(w) = \{(i, w(j)) : (i, j) \in \mathsf{Inv}(w)\}$$

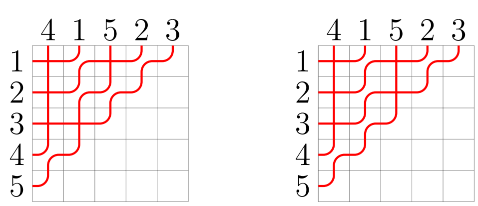
Example 3.2. The following is $\mathsf{Rothe}(w)$ for the permutation $w = 24513$. Its inversion set is $\mathsf{Inv}(w) = \{(1,4),(2,4),(2,5),(3,4),(3,5)\}$, so its Rothe diagram consists of the cells $\{(1,1),(2,1),(2,3),(3,1),(3,3)\}$.



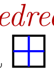
3.2. Pipedream. We now introduce our first combinatorial object that computes Schubert and Grothendieck polynomials called pipedreams, defined by Bergeron and Billey [BB93].

Definition 3.3. A *reduced pipedream* of size n is a tiling with $n+1-i$ left justified tiles in row i. The tile $(i, n+1-i)$ is  for $i \in [n]$. All other tiles can be  or  but multiple crossings of two pipes are not allowed. We first label the left of the diagram $1, \ldots, n$ from top to bottom, then we trace the pipes from left to top and read off the labels of the pipes on the top edge of the reduced pipedream as a permutation $w \in S_n$. We say this reduced pipedream is associated with w^{-1}. Let $\mathsf{PD}'(w)$ denote the set of all reduced pipedreams associated with w.

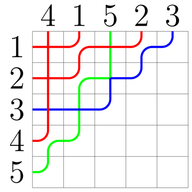
Example 3.4. The following are two reduced pipedreams of the permutation w with one-line notation 24513. Its inverse has one-line notation 41523.



Notice that no pair of pipes crossed more than once.

Definition 3.5. A *non-reduced pipedream* is similar to a reduced pipedream but we allow double crossings. Suppose we see a  where the pipe on the left (resp. bottom) has label p (resp. q). If pipe p and q have not crossed before, we say they cross in this tile and let pipe p (resp. q) exit from the right (resp. top). Otherwise, we let pipe p (resp. q) exit from the top (resp. right). Notice that this rule is the same as saying pipe $\max(p, q)$ exits from the top and the other exits from the right.

Example 3.6. The following is a non-reduced pipedream of the same permutation 24513 with inverse 41523.



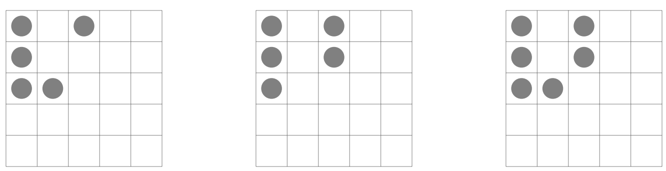
We make pipe 3 blue and pipe 5 green. Notice that pipe 3 and pipe 5 cross at $(3,2)$. However, pipe 3 and pipe 5 do not cross at $(2,3)$ since they already crossed.

Definition 3.7. We simply call the union of reduced and non-reduced pipedreams of $w \in S_n$ as the *pipedreams* of w, denote $\mathsf{PD}(w)$. The *row (resp. column) weight* of a pipedream is a weak composition where the i^{th} entry is the number of $\boxplus$ in row (resp. column) i of the pipedream.

The pipedream in Example 3.6 therefore has row weight $(2,2,2,0,0)$ and column weight $(3,1,2,0,0)$.

Since the positions of all the $\boxplus$ uniquely determine a pipedream, we may denote a pipedream as an $n \times n$ diagram where colored cells are the locations of $\boxplus$ and blank cells are the locations of .

Example 3.8. Below are the pipedreams in Example 3.4 and 3.6 under our new convention.



We now determine row (resp. column) weight by counting the number of cells in each row (resp. column).

We define the weight of a pipedream as

$$\mathsf{wt}_P(\mathbf{x}) = \prod_{(i,j)\in P} x_i, \quad \mathsf{wt}_P(\mathbf{x},\mathbf{y}) = \prod_{(i,j)\in P} (x_i + y_j - x_i y_j)$$

Theorem 3.9. *Following* [FK94] *and* [KM05]*, Schubert polynomials $\mathfrak{S}_w(\boldsymbol{x})$, Grothendieck polynomial $\mathfrak{G}_w(\boldsymbol{x})$, and double Grothendieck polynomial $\mathfrak{G}_w(\boldsymbol{x},\boldsymbol{y})$ can be defined as*

$$\mathfrak{S}_w(\boldsymbol{x}) := \sum_{P \in \mathsf{PD}'(w)} \mathsf{wt}_P(\boldsymbol{x})$$

$$\mathfrak{G}_w(\boldsymbol{x}) := \sum_{P \in \mathsf{PD}(w)} (-1)^{|\mathsf{wty}(P)| - \ell(w)} \mathsf{wt}_P(\boldsymbol{x})$$

$$\mathfrak{G}_w(\boldsymbol{x},\boldsymbol{y}) := \sum_{P \in \mathsf{PD}(w)} (-1)^{|\mathsf{wty}(P)| - \ell(w)} \mathsf{wt}_P(\boldsymbol{x},\boldsymbol{y})$$

Therefore, each pipedream represents a monomial in $\mathfrak{G}_w(\mathbf{x})$ with degree equal to its row weight. We may then study these polynomials and their support using pipedreams.

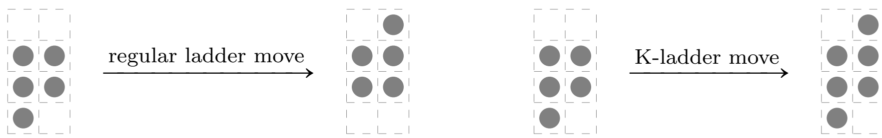
We now introduce actions that we can perform on a pipedream $P \in \mathsf{PD}(w)$ that give us other pipedreams in $\mathsf{PD}(w)$.

Definition 3.10. When row r column c of a pipedream P is $\boxplus$, we write $(r, c) \in P$. We may apply a ladder move on a $\boxplus$ in row r column c of a pipedream P if all the following are satisfied:

- $(r, c+1) \notin P$.
- There exists $r' < r$ such that $(r', c) \notin P$ and $(r', c+1) \notin P$. In addition, $(i, c), (i, c+1) \in P$ for any $r' < i < r$.

Now we perform the ladder move at the $\boxplus$ in row r column c of P. First turn the $\boxslash$ at row r' column $c+1$ into a $\boxplus$. Then we may or may not turn the $\boxplus$ at row r column c into $\boxslash$. If we do that, the move is called a *(regular) ladder move*. Otherwise, the move is called a *K-ladder move*.

Locally, the moves look like the following:

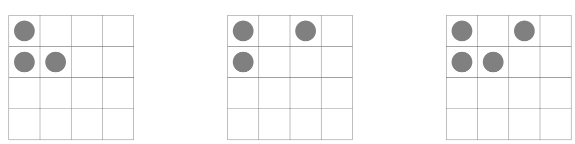


Let $\overleftarrow{\mathsf{Rothe}(w)}$ be the diagram obtained by left-justifying all cells in $\mathsf{Rothe}(w)$. The following proposition includes three statements about (K-)ladder moves and pipedreams that are well known to experts. We will prove them below for completeness.

Proposition 3.11. *For all $w \in S_n$, we have the following:*

(1) $\overleftarrow{\mathsf{Rothe}(w)}$ is a (reduced) pipedream of $\mathsf{PD}(w)$.
(2) $\mathsf{PD}(w)$ is closed under (K-)ladder moves.
(3) Every pipedream $P \in \mathsf{PD}(w)$ can be obtained by applying a series of (K-)ladder moves starting from $\overleftarrow{\mathsf{Rothe}(w)}$.

Example 3.12. The following are all the pipedreams of the permutation $w = 2413$. The left pipedream is $\overleftarrow{\mathsf{Rothe}(w)}$. We apply a regular (resp. K-) ladder move on $(2, 2)$ to obtain the middle (resp. right) pipedream.



Since these are all the pipedreams in $\mathsf{PD}(w)$, by Theorem 3.9,

$$\begin{aligned}
\mathfrak{S}_w(\mathbf{x}) &= x_1x_2^2 + x_1^2x_2 \\
\mathfrak{G}_w(\mathbf{x}) &= x_1x_2^2 + x_1^2x_2 - x_1^2x_2^2 \\
\mathfrak{G}_w(\mathbf{x}, \mathbf{y}) &= (x_1 + y_1 - x_1y_1)(x_2 + y_1 - x_2y_1)(x_2 + y_2 - x_2y_2) \\
&\quad + (x_1 + y_1 - x_1y_1)(x_2 + y_1 - x_2y_1)(x_1 + y_3 - x_1y_3) \\
&\quad - (x_1 + y_1 - x_1y_1)(x_2 + y_1 - x_2y_1)(x_2 + y_2 - x_2y_2)(x_1 + y_3 - x_1y_3)
\end{aligned}$$

Proof. Statement (1) is trivial, and (2) can also be seen by tracing the inputs and outputs along the edge of the $n \times 2$ rectangle.
We prove (3) by showing that all pipedreams in $\mathsf{PD}(w)$ are $\overleftarrow{\mathsf{Rothe}(w)}$ after a series of reverse (K-)ladder moves. We use the fact that any left-justified pipedream in $\mathsf{PD}(w)$ must be

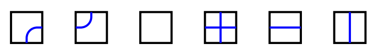
$\overleftarrow{\mathsf{Rothe}}(w)$. Suppose $P \in \mathsf{PD}(w)$ is not left-justified. Let (r, c) be the crossing in P such that $(r, c-1)$ is not a crossing and r is chosen to be maximal. Since r is chosen to be maximal, there does not exist $r' > r$ such that $\{(r', c), (r', c-1)\} \cap P = \{(r', c-1)\}$. Therefore, there exists $\hat{r} > r$ that satisfies the following two statements:

$$\{(r', c), (r', c-1)\} \cap P = \{(r', c), (r', c-1)\} \quad \forall r < r' < \hat{r} \tag{1}$$

$$\{(\hat{r}, c-1), (\hat{r}, c)\} \cap P = \{(\hat{r}, c)\} \quad \text{or} \quad \{(\hat{r}, c-1), (\hat{r}, c)\} \cap P = \varnothing \tag{2}$$

If $\{(\hat{r}, c-1), (\hat{r}, c)\} \cap P = \{(\hat{r}, c)\}$, then we can perform a reverse K-ladder move on (r, c) that simply removes (r, c) from P. And if $\{(\hat{r}, c-1), (\hat{r}, c)\} \cap P = \varnothing$, then we can perform a reverse ladder move on (r, c) that removes (r, c) and adds $(\hat{r}, c-1)$. We repeat this action if our pipedream is still not left-justified. Since we are strictly decreasing the column index of cells during each reverse (K-)ladder move, our sequence of action must terminate, resulting in a left-justified pipedream. □

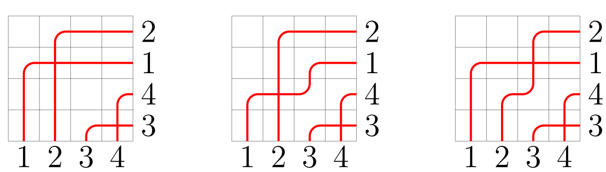
3.3. **Bumpless pipedream.** Introduced by Lam, Lee, and Shimozono [LLS21], bumpless pipedream is another combinatorial object in Schubert calculus. Similar to pipedreams, we will define reduced bumpless pipedreams that compute Schubert polynomials and (non-reduced) bumpless pipedreams that compute Grothendieck polynomials. Both of which are tilings consisting of the following six types of tiles:



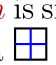
Compared to pipedreams, we are not using any , hence the name "bumpless pipedream".

Definition 3.13. A *reduced bumpless pipedream* is a tiling of an $n \times n$ grid with the 6 tiles above forming a system of n pipes entering from the bottom of the diagram and exiting from the right. Multiple crossings are not allowed. We label the pipes by the column they start in and read off the labels on the right as a permutation w in one-line notation. We say this bumpless pipedream is associated w and denote the set of all reduced bumpless pipedream as $\mathsf{BPD}'(w)$.

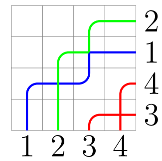
Example 3.14. The following are all the reduced bumpless pipedreams of the permutation $w = 2143$.



Notice that no pair of pipes crossed more than once.

Definition 3.15. A *non-reduced bumpless pipedream* is similar to a reduced bumpless pipedream but we allow multiple crossings. Suppose we see a  where the pipe on the left (resp. bottom) has label p (resp. q). If pipe p and q have not crossed before, we say they cross in this tile and let pipe p (resp. q) exit from the right (resp. top). Otherwise, we let pipe p (resp. q) exit from the top (resp. right). Notice that this rule is the same as saying pipe $\max(p, q)$ exits from the top and the other exits from the right.

Example 3.16. The following is a non-reduced bumpless pipedream of the permutation $w = 2143$.



We make pipe 1 blue and pipe 2 green. Notice that pipe 1 and pipe 2 cross at $(3,2)$. However, pipe 1 and pipe 2 do not cross at $(2,3)$ since they already crossed.

We simply call the union of all reduced bumpless pipedreams and non-reduced bumpless pipedreams associated with the permutation w as the bumpless pipedreams of w, denoted $\mathsf{BPD}(w)$. For any bumpless pipedream P, let $B(P)$ be the set of in P and $J(P)$ be the set of in P.

Theorem 3.17 ([LLS21, Theorem 5.2])**.**

$$\mathfrak{S}_w(\boldsymbol{x}) = \sum_{P \in \mathsf{BPD}'(w)} \prod_{(i,j)\in B(P)} x_i$$

Equivalently, the Schubert polynomial $\mathfrak{S}_w(\mathbf{x})$ is the sum over all weighted reduced bumpless pipedreams of w, where the weight of P is the number of in each row.

Theorem 3.18.

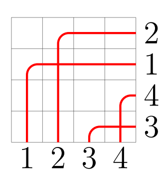
$$\mathfrak{G}_w(\boldsymbol{x}) = \sum_{P \in \mathsf{BPD}'(w)} \prod_{(i,j)\in B(P)} x_i \prod_{(i,j)\in J(P)} (1 - x_i)$$

Equivalently, the Grothendieck polynomial $\mathfrak{G}_w(\mathbf{x})$ is the sum over all weighted bumpless pipedreams of w, where each contributes x_i and contributes $1 - x_i$.

To generate all bumpless pipedreams of w, we define the Rothe bumpless pipedream $\mathsf{RBPD}(w)$ and (K-) droop moves. We first notice that each bumpless pipedream is uniquely determined by the locations of its and .

Definition 3.19. Let the *Rothe bumpless pipedream* of w, denote $\mathsf{RBPD}(w)$, be the reduced bumpless pipedream of w obtained by placing at $(w^{-1}(i), i)$ with no . It is called the Rothe bumpless pipedream because the of $\mathsf{RBPD}(w)$ is exactly $\mathsf{Rothe}(w)$.

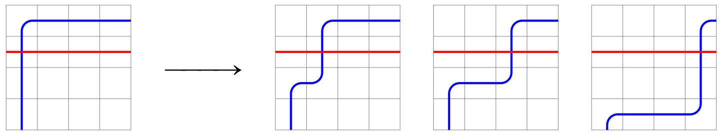
Example 3.20. Let $w = 2143$. The following is $\mathsf{RBPD}(w)$.



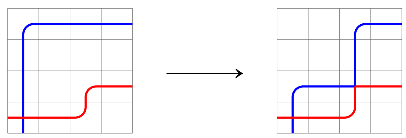
We now define droop moves and K-droop moves that can be performed on a bumpless pipedream of w to obtain other bumpless pipedreams of w.

Definition 3.21. A *droop move* is a local move that replaces a with and replaces a to its southeast with a . A *K-droop move* replaces a with and replaces a to its southeast with a , where this is not the only crossing between the two pipes. For either move, we reconnect the pipes accordingly afterwards.

Example 3.22. The following are some of the possible droop moves applied to the at the top left corner.



Example 3.23. The following is an example of a K-droop moves applied to the at the top left corner.



Notice that the number of increases by 1 after a K-droop move.

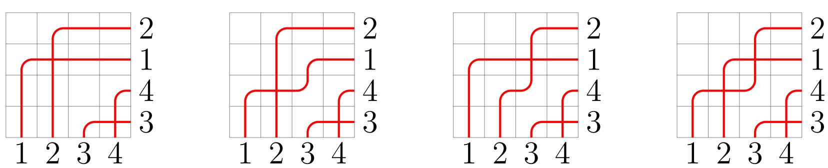
Proposition 3.24 ([LLS21, Proposition 5.3])**.** *Every reduced bumpless pipedream of w can be obtained from* $\mathsf{RBPD}(w)$ *by a sequence of droop moves.*

We may obtain the bumpless pipedreams in Example 3.14 from $\mathsf{RBPD}(w)$ by applying a droop move to the at $(1, 2)$ and $(2, 1)$ respectively.

Proposition 3.25. *Every bumpless pipedream of w can be obtained from* $\mathsf{RBPD}(w)$ *by a sequence of droop moves and K-droop moves.*

We may obtain the non-reduced bumpless pipedream in Example 3.16 by applying a K-droop move to $(1, 2)$ in the second bumpless pipedream or to $(2, 1)$ in the third bumpless pipedream in Example 3.14.

Example 3.26. The following are all the bumpless pipedreams for $w = 2143$. The first three are reduced bumpless pipedreams and the last one is non-reduced.



Therefore

$$\mathfrak{S}_w(\mathbf{x}) = x_1x_3 + x_1x_2 + x_1^2$$
$$\mathfrak{G}_w(\mathbf{x}) = x_1x_3 + x_1x_2(1 - x_3) + x_1^2(1 - x_3) + x_1^2x_2(1 - x_3)$$

Since $\mathsf{PD}'(w)$ and $\mathsf{BPD}'(w)$ both compute $\mathfrak{S}_w(\mathbf{x})$, one might expect a bijection between these two combinatorial objects. Gao and Huang gave a direct bijection by interpreting reduced compatible sequences on bumpless pipedreams.

Theorem 3.27 ([GH23, Theorem 3.6])**.** *There exits a canonical bijection between* $\mathsf{PD}'(w)$ *and* $\mathsf{BPD}'(w)$ *that preserves Monk's rule.*

Tianyi Yu communicated with us that Theorem 3.27 can be extended to a bijection between $\mathsf{PD}(w)$ and $\mathsf{BPD}(w)$.

4. Maximal pipedreams of double Grothendieck polynomials

In this section, we provide an explicit algorithm that constructs the unique maximal pipedream that represents the leading monomial of $\mathfrak{G}_w(\mathbf{x}, \mathbf{y})$.

4.1. **Background.** Pechenik, Speyer, and Weigandt [PSW21] defined a statistic $\mathsf{rajcode}(\cdot)$ on permutations $w \in S_n$ using their increasing subsequences. They showed $x^{\mathsf{rajcode}(w)}$ is the leading monomial in the top degree components of its Grothendieck polynomial $\mathfrak{G}_w(\mathbf{x})$ with respect to the lexicographical order where $x_n > \cdots > x_1$. This permutation statistic also generalizes to double Grothendieck polynomials by the following theorem.

Theorem 4.1 ([PSW21, Theorem 1.4])**.** *The leading monomial of top degree components of $\mathfrak{G}_w(\boldsymbol{x}, \boldsymbol{y})$ is $x^{\mathsf{rajcode}(w)} y^{\mathsf{rajcode}(w^{-1})}$ with coefficient 1 for any term order with $x_n > \cdots > x_1$ and $y_n > \cdots > y_1$.*

Pechenik, Speyer, and Weigandt established Theorem 4.1 by showing there exists a unique pipedream in $\mathsf{PD}(w)$ with row weight $\mathsf{rajcode}(w)$ and column weight $\mathsf{rajcode}(w^{-1})$, which they call the maximal pipedream of w. However, in Remark 7.2, they said:

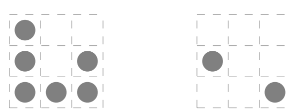
> "We find it frustrating that we do not have a direct recipe for the maximal pipe dream in terms of w."

The main goal of this section is to relieve their frustration: We give an explicit algorithm to construct the maximal pipedream $\widehat{P}(w) \in \mathsf{PD}(w)$.

We start by a diagrammatic definition of $\mathsf{rajcode}(w)$ given by Pan and Yu [PY23].

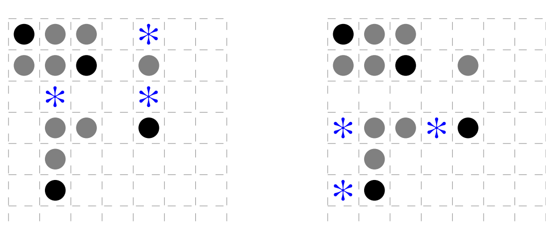
Definition 4.2. For any diagrams D, we defined $\mathsf{dark}(D) \subseteq D$ which can be computed as follows: Scan through D from bottom to top. For each row r, if there exists $(r, c) \in D$ such that currently there is no cells in column c of $\mathsf{dark}(D)$, we find the largest such c and put (r, c) in $\mathsf{dark}(D)$. Cells in $\mathsf{dark}(D)$ of D are called *dark clouds* of D.

Example 4.3. The following is a diagram D and $\mathsf{dark}(D)$



Definition 4.4 ([PY23]). Take $w \in S_n$ and find $\mathsf{dark}(\mathsf{Rothe}(w))$. For each cell in $\mathsf{dark}(\mathsf{Rothe}(w))$, we fill all the empty cells above it in $\mathsf{Rothe}(w)$. The resulting diagram is the *snow diagram* $\mathsf{Snow}(w)$ of w. We define $\mathsf{rajcode}(w)$ as the row weight of $\mathsf{Snow}(w)$. Similarly, if for each cell in $\mathsf{dark}(\mathsf{Rothe}(w))$, we fill all the empty cells to its right in $\mathsf{Rothe}(w)$, we obtain the *left snow diagram* of w. We define $\mathsf{rajcode}(w^{-1})$ as the column weight of the left snow diagram.

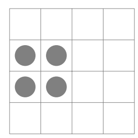
Example 4.5. Take $w \in S_7$ with one-line notation 4617352. The following is its snow diagram and left snow diagram. For clarity, we represent dark clouds by a black circle and use $\ast$ to denote the added cells.



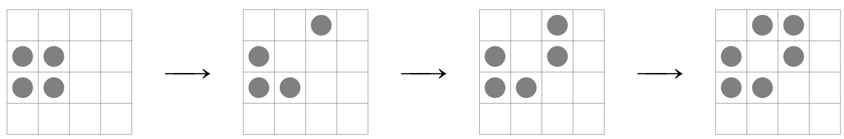
Thus, $\mathsf{rajcode}(w) = (4, 4, 2, 3, 1, 1)$ and $\mathsf{rajcode}(w^{-1}) = (4, 5, 3, 1, 2)$

4.2. **Algorithm.** We now give the algorithm that constructs $\widehat{P}(w)$ starting from $\overleftarrow{\mathsf{Rothe}}(w)$. We perform an iterative algorithm. Each iteration places a bar right above row i for $i = n-2, n-3, \cdots, 1$. During each iteration, we only look under the bar and imagine row i is the topmost row. Scan through the columns from right to left. Within each column, scan through the $\boxplus$ from top to bottom. Whenever we see a $\boxplus$ at which we can perform a ladder move, we perform a regular ladder move. After going through a column, if we have performed ladder moves on this column, we turn the last ladder move into a K-ladder move. We repeat until no moves can be made under the bar.

Example 4.6. Take $w \in S_5$ with one-line notation 14523. We start from the following pipedream:



When $i = 3$ and 2, we do not make any moves. When $i = 1$, we perform:



Theorem 4.7 ([CY23, Theorem 1.2])**.** *For $w \in S_n$, the pipedream $\widehat{P}(w)$ we construct has row weight* $\mathsf{rajcode}(w)$ *and column weight* $\mathsf{rajcode}(w^{-1})$*.*

Furthermore, our algorithm gives the pipedreams of all leading monomials in each homogeneous component of a Grothendieck polynomial $\mathfrak{G}_w(\mathbf{x})$. Dreyer, Mészáros, and St. Dizier [DMSD22] found the leading monomial in each homogeneous component of $\mathfrak{G}_w(\mathbf{x})$. Let $\mathsf{reg}(w)$ be the difference between the sum of entries in $\mathsf{rajcode}(w)$ and the sum of entries in $\mathsf{invcode}(w)$. Define the map $\mathsf{IR}(\cdot)$ that sends w to a sequence of monomials $m_0, m_1, \cdots, m_{\mathsf{reg}(w)}$. First, $m_0 := x^{\mathsf{invcode}(w)}$. For $i > 0$, $m_i := m_{i-1}x_p$ where p is the largest such that $m_{i-1}x_p$ divides $x^{\mathsf{rajcode}(w)}$. For each m_i, Dreyer, Mészáros, and St. Dizier [DMSD22] explicitly constructed a climbing chain, another combinatorial model of $\mathfrak{G}_w(\mathbf{x})$ introduced in [LRS06], showing m_i is the leading monomial in its degree of $\mathfrak{G}_w(\mathbf{x})$. In our algorithm, we start from a pipedream with row weight $\mathsf{invcode}(w)$. During the algorithm, we obtain the pipedreams corresponding to $m_1, \cdots, m_{\mathsf{reg}(w)}$.

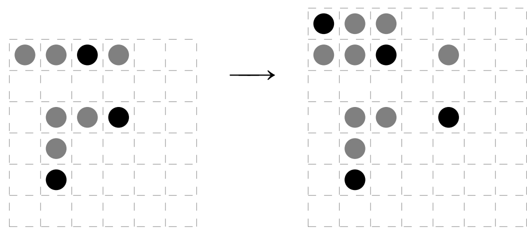
Theorem 4.8 ([CY23, Theorem 1.4])**.** *Let $w \in S_n$. Perform our algorithm to compute $\widehat{P}(w)$. The algorithm makes* $\mathsf{reg}(w)$ *K-ladder moves. Right after the i^{th} K-ladder move, we record the row weight of the pipedream as $a_i(w)$. Then $x^{a_i(w)} = m_i$ where* $\mathsf{IR}(w) = (m_0, m_1, \cdots, m_{\mathsf{reg}(w)})$*.*

4.3. **Various recursions.** In this subsection, we provide recursive ways of constructing $\mathsf{Rothe}(w)$, $\overleftarrow{\mathsf{Rothe}}(w)$, $\mathsf{dark}(\mathsf{Rothe}(w))$, and $\mathsf{Snow}(w)$. Then we obtain recursive formulas for $\mathsf{rajcode}(w)$ and $\mathsf{rajcode}(w^{-1})$. Notice that $\mathsf{invcode}(\cdot)$ is a bijection from S_n to weak compositions $(\alpha_1, \alpha_2, \cdots)$ where $\alpha_i \leqslant n - i$ for $i \in [n-1]$ and $\alpha_n = \alpha_{n+1} = \cdots = 0$. We identify $w \in S_n$ with $(a, u) \in \{0, 1, \cdots, n-1\} \times S_{n-1}$ where $a = \mathsf{invcode}(w)_1$ and u is the unique permutation in S_{n-1} with $\mathsf{invcode}(u) = (\mathsf{invcode}(w)_2, \mathsf{invcode}(w)_3, \cdots)$. We simply write $w = (a, u)$.

We then may recursively construct $\mathsf{Rothe}(w)$ as follows. Start from $\mathsf{Rothe}(u)$. Shift all cells downward by 1. Then shift all cells in columns $a+1, a+2, \cdots$ to the right by 1. Finally, put cells at $(1,1), \cdots, (1,a)$. The resulting diagram is $\mathsf{Rothe}(w)$.

Similarly, to construct $\mathsf{dark}(\mathsf{Rothe}(w))$, we can start from $\mathsf{dark}(\mathsf{Rothe}(u))$. Shift all cells downward by 1. Then shift all cells in columns $a+1, a+2, \cdots$ to the right by 1. Finally, find the largest $c \in [a]$ such that $\mathsf{dark}(\mathsf{Rothe}(u))$ has no cells in column c. Put $(1,c)$ into $\mathsf{dark}(\mathsf{Rothe}(u))$.

Example 4.9. Keep $w \in S_7$ with one-line notation 4617352. We have $w = (a, u)$ where $a = 3$ and $u \in S_6$ has one-line notation 516342. We depict how $\mathsf{Rothe}(u)$ and $\mathsf{Rothe}(w)$ as follows. The dark cells form $\mathsf{dark}(\mathsf{Rothe}(u))$ and $\mathsf{dark}(\mathsf{Rothe}(w))$ respectively.



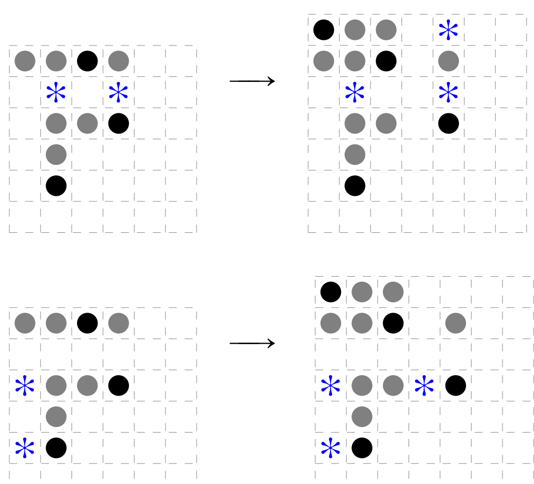
Consequently, we may compute $\mathsf{rajcode}(w)$ and $\mathsf{rajcode}(w^{-1})$ recursively. Let $d_c(u)$ be the number of cells in $\mathsf{dark}(\mathsf{Rothe}(u))$ that are strictly to the right of column c.

Proposition 4.10. *Take $w = (a, u) \in S_n$.*
- *We can get $\mathsf{rajcode}(w)$ by prepending $a + d_a(u)$ to $\mathsf{rajcode}(u)$.*
- *To obtain $\mathsf{rajcode}(w^{-1})$, we just insert $d_a(u)$ between the a^{th} and $(a+1)^{th}$ entries of $\mathsf{rajcode}(u^{-1})$. Then increase the first a entries by 1.*

Consequently, $\mathsf{reg}(w) - \mathsf{reg}(u) = d_a(u)$.

Proof. Follows directly from the recursive constructions of $\mathsf{Rothe}(w)$ and $\mathsf{dark}(\mathsf{Rothe}(w))$. □

Example 4.11. Keep $w = (a, u)$ in Example 4.9. We show how the snow diagram and left snow diagram of w differ from those of u:



We have $d_a(u) = 1$. We obtain $\mathsf{rajcode}(w) = (4,4,2,3,1,1)$ by prepending $a + d_a(u) = 4$ to $\mathsf{rajcode}(u) = (4,2,3,1,1)$. We obtain $\mathsf{rajcode}(w^{-1}) = (4,5,3,1,2)$ by inserting $d_a(u)$ after the a^{th} entry of $\mathsf{rajcode}(u^{-1}) = (3,4,2,2)$ and then increase the first a entries by 1.

Notice that when $w = (a, u)$, $\mathsf{invcode}(w)$ can be obtained by prepending the number a to $\mathsf{invcode}(u)$. Thus, we also have a recursive formula for $\mathsf{IR}(w)$. For a monomial m, let $\overrightarrow{m}$ be the monomial obtained by turning each x_i in m into x_{i+1}.

Proposition 4.12. *Take $w = (a, u) \in S_n$. Let*

$$(M_0, \cdots, M_{\mathsf{reg}(w)}) = \mathsf{IR}(w), (m_0, \cdots, m_{\mathsf{reg}(u)}) = \mathsf{IR}(u).$$

Then $\mathsf{reg}(w) = \mathsf{reg}(u) + d_a(u)$ and

$$M_j = \begin{cases} x_1^a \overrightarrow{m_j} & \text{if } j = 0, 1, \cdots, \mathsf{reg}(u), \\ x_1^{a+j-\mathsf{reg}(u)} \times \overrightarrow{m_{\mathsf{reg}(u)}} & \text{if } j = \mathsf{reg}(u) + 1, \cdots, \mathsf{reg}(w). \end{cases}$$

Proof. Follows directly from the recursive formula of $\mathsf{rajcode}(w)$ and the definition of $\mathsf{IR}(\cdot)$. □

Example 4.13. Keep $w = (a, u)$ in Example 4.9. We have $\mathsf{reg}(u) = 2$ and $\mathsf{reg}(w) = \mathsf{reg}(u) + d_a(u) = 3$. Since

$$\mathsf{IR}(u) = (x^{(4,0,3,1,1)}, x^{(4,1,3,1,1)}, x^{(4,2,3,1,1)}),$$

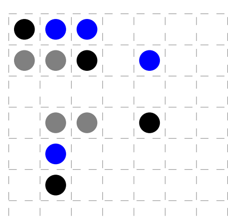
we have

$$\mathsf{IR}(w) = (x^{(3,4,0,3,1,1)}, x^{(3,4,1,3,1,1)}, x^{(3,4,2,3,1,1)}, x^{(4,4,2,3,1,1)})$$

4.4. **Proof of Theorem 4.7 and 4.8.** To prove our main theorems, we need to introduce a new permutation statistic.

Definition 4.14. For $w \in S_n$, its *movecode*, denoted as $\mathsf{movecode}(w)$, is a weak composition where $\mathsf{movecode}(w)_i$ is the number of cells in column i of $\mathsf{Rothe}(w)$ with no dark clouds strictly to its right.

Example 4.15. Take $w \in S_7$ with one-line notation 4617352. The following is $\mathsf{Rothe}(w)$, where the black cells are dark clouds and blue cells are non-dark cloud cells without dark clouds to their right.



Then $\mathsf{movecode}(w)$ is the number of black and blue cells in each column, which is $(1, 3, 2, 0, 2)$.

We have the following observation regarding this permutation statistic.

Proposition 4.16. *Take $w \in S_n$ and $c \in [n]$. Then*

$$\mathsf{rajcode}(w^{-1})_{c+1} - \max(\mathsf{movecode}(w)_{c+1} - 1, 0) = d_c(u) = \mathsf{rajcode}(w^{-1})_c - \mathsf{movecode}(w)_c.$$

Proof. We refer to cells in $\mathsf{dark}(\mathsf{Rothe}(w))$ as dark clouds. Consider the left snow diagram of w. In the diagram, there are four types of cells.

- Type 1: Dark clouds
- Type 2: Cells that do not belong to $\mathsf{Rothe}(w)$.
- Type 3: Cells in $\mathsf{Rothe}(w)$ with a dark cloud in its row on its right.

- Type 4: Cells in $\mathsf{Rothe}(w)$ that is not a dark cloud and has no dark cloud in its row on its right.

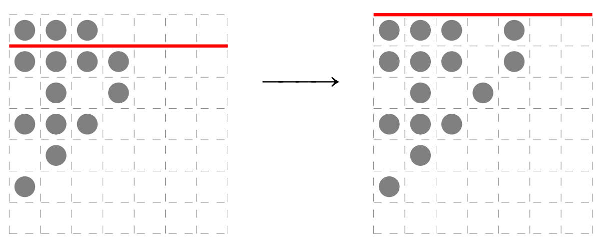
The number of type 1, 2 and 4 cells in column $c + 1$ is $d_c(w)$. The number of all cells in column $c + 1$ is $\mathsf{rajcode}(w^{-1})_{c+1}$. The number of type 3 cells in column $c + 1$ is $\max(\mathsf{movecode}(w)_c - 1, 0)$, so we have the first equation.

The number of type 2 and 3 cells in column c is $d_c(w)$. The number of all cells in column c is $\mathsf{rajcode}(w^{-1})_c$. The number of type 1 and 4 cells in column c is $\mathsf{movecode}(w)_c$, so we have the second equation. $\square$

The main application of $\mathsf{movecode}(w)$ is to characterize the number of cells moved when our algorithm processes each column.

Proposition 4.17. *Take $v = (a, w) \in S_n$. During the last iteration of the algorithm that computes $\widehat{P}(v)$, the number of cells moved in column c is $\mathsf{movecode}(w)_c$ if $c > a$ and 0 otherwise.*

Example 4.18. Keep $v \in S_7$ with one-line notation 4617352. We have $v = (a, w)$ where $a = 3$ and $w \in S_6$ has one-line notation 516342. We have $\mathsf{movecode}(w) = (0, 2, 1, 2)$. During the last iteration of the algorithm, the bar is right above row 1. The algorithm moves 0 cells in column $c > 4$, since $\mathsf{movecode}(w)_c = 0$. The algorithm moves 2 cells in column 4 since $4 > a$ and $\mathsf{movecode}(w)_4 = 2$. It moves 0 cells in column $3, 2$, and 1 since $1, 2, 3 \leqslant a$.



We prove this proposition in subsection 4.5. Our proof requires a few technical lemmas which also lead to the following result:

Corollary 4.19. *Consider the iteration when the bar is right above row i in our algorithm. Let D_1 (resp. D_2) be the diagram before (resp. after) processing one column. If the algorithm makes a move in this column, then $\mathsf{wt}(D_2)$ is obtained from increasing i^{th} entry of $\mathsf{wt}(D_1)$ by 1.*

Using Proposition 4.17 and Corollary 4.19, we can prove our main results. We start with Theorem 4.8.

Proof of Theorem 4.8. We induct on n. The base case ($n = 1$) is trivial. Let $w = (a, u) \in S_n$ with $n > 1$. By our inductive hypothesis, the algorithm made $\mathsf{reg}(u)$ K-ladder moves before the last iteration. By Proposition 4.17, in the last iteration of the algorithm, it makes a K-ladder move in column c if and only if $c > a$ and $\mathsf{movecode}(u)_c > 0$. This is exactly the number $d_a(u)$, which equals $\mathsf{reg}(w) - \mathsf{reg}(u)$ by Proposition 4.10. Thus, the algorithm to compute $\widehat{P}(w)$ makes $\mathsf{reg}(w)$ K-ladder moves in total.

Let

$$\mathsf{IR}(w) = (M_0, \cdots, M_{\mathsf{reg}(w)}), \mathsf{IR}(u) = (m_0, \cdots, m_{\mathsf{reg}(u)}).$$

By Proposition 4.12, for $i = 0, \cdots, \mathsf{reg}(u)$, we have $M_i = x_1^a \overrightarrow{m_i}$. When the algorithm makes the i^{th} K-ladder move, the bar has not reached row 1. Before the bar reaches row 1, the algorithm ignores the first row of the diagram, which has a cells, and behaves as if computing $\widehat{P}(u)$. Thus, the statement holds for $i = 0, 1, \cdots, \mathsf{reg}(u)$ by our inductive hypothesis.

For $i = \mathsf{reg}(u) + 1, \cdots, \mathsf{reg}(w)$, the i^{th} K-ladder move happens when the bar is above row 1. Let D be the diagram right after the $(i-1)^{\text{th}}$ K-ladder move and D' be the diagram right after the i^{th} K-ladder move. By Corollary 4.19, $x^{\mathsf{wt}(D')} = x_1 \cdot x^{\mathsf{wt}(D)}$, which concludes the proof. $\square$

Proof of Theorem 4.7. By Theorem 4.8, the row weight of $\widehat{P}(w)$ is $\mathsf{rajcode}(w)$. For the column weight, we prove by induction on n. The base case $n = 1$ is trivial. Now assume $n > 1$ and $w = (a, u) \in S_n$. Let D be the diagram we have right before the last iteration of the algorithm computing $\widehat{P}(w)$. It can be obtained by shifting $\widehat{P}(u)$ downward by 1 and append a left-justified cells in the first row. By our inductive hypothesis, $\widehat{P}(u)$ has column weight $\mathsf{rajcode}(u^{-1})$. Now take $c \in [n-1]$ and consider three cases:

- Suppose $c > a + 1$. Consider the last iteration of the algorithm. By Proposition 4.17, the algorithm makes $\mathsf{movecode}(u)_c$ (resp. $\mathsf{movecode}(u)_{c-1}$) moves in column c (resp. $c-1$). Thus, column c loses $\max(\mathsf{movecode}(u)_c - 1, 0)$ cells and then gain $\mathsf{movecode}(u)_{c-1}$ cells. By Proposition 4.16, $\widehat{P}(w)$ has
$$\mathsf{rajcode}_c(u^{-1}) - \max(\mathsf{movecode}(u)_c - 1, 0) + \mathsf{movecode}(u)_{c-1} = \mathsf{rajcode}_{c-1}(u^{-1})$$
cells in column c. Finally, by Proposition 4.10, $\mathsf{rajcode}_{c-1}(u^{-1})$ is just $\mathsf{rajcode}_c(w^{-1})$.
- Suppose $c = a + 1$. By Proposition 4.17, the algorithm makes $\mathsf{movecode}(u)_c$ moves in column c, and makes 0 moves in column $c-1$ if it exists. Thus, column c loses $\max(\mathsf{movecode}(u)_c - 1, 0)$ cells. By Proposition 4.16, $\widehat{P}(w)$ has
$$\mathsf{rajcode}_c(u^{-1}) - \max(\mathsf{movecode}(u)_c - 1, 0) = d_a(u)$$
cells in column c. Finally, by Proposition 4.10, $d_a(u)$ is just $\mathsf{rajcode}_c(w^{-1})$.
- Suppose $c \in [a]$. By Proposition 4.17, the algorithm makes 0 moves in column c, and makes 0 moves in column $c-1$ if it exists. Thus, $\widehat{P}(w)$ has $\mathsf{rajcode}(u^{-1})_c + 1$ cells in column c. Finally, by Proposition 4.10, $\mathsf{rajcode}(u^{-1})_c + 1$ is just $\mathsf{rajcode}_c(w^{-1})$. $\square$

4.5. **Proof of Proposition 4.17 and Corollary 4.19.** Following subsection 4.3, we derive a recursive way to compute $\mathsf{movecode}(w)$.

Lemma 4.20. *For $w \in S_n$, we write $w = (a, u)$. Then $\mathsf{movecode}(w)$ can be determined starting from $\mathsf{movecode}(u)$. First, insert a 0 between $\mathsf{movecode}(u)_a$ and $\mathsf{movecode}(u)_{a+1}$. Then start from the a^{th} entry and increase each entry by 1 from right to left. Whenever we change a 0 into a 1, we stop immediately. The resulting weak composition is $\mathsf{movecode}(w)$.*

Proof. Follows directly from the recursive constructions of $\mathsf{Rothe}(w)$ and $\mathsf{dark}(\mathsf{Rothe}(w))$. $\square$

Example 4.21. Take $w \in S_7$ with one-line notation 4617352. We have $w = (3, u)$ where $u \in S_6$ has one-line notation 516342. We have $\mathsf{movecode}(u) = (0, 2, 1, 2)$. Then we insert a 0 between $\mathsf{movecode}(u)_3$ and $\mathsf{movecode}(u)_4$, obtaining $(0, 2, 1, 0, 2)$. We then increases entries by 1 from right to left, starting from the thrid entry. When we turn the 0 in the first entry into 1, we stop, obtaining $(1, 3, 2, 0, 2)$.

Our proofs rely on a simple operator on diagrams. We may break the algorithm into a sequence of this operator.

Definition 4.22. We define the operator $L_{i,c}$ on diagrams. Take diagram D and put a bar above row i in D. We ignore everything above the bar, imagining row i is the top-most row. Then we scan through cells in column c from top to bottom. Whenever we see a cell at which we can perform a ladder move, we perform a regular ladder move. After going through this column, if we made a move, turn the last move into a K-ladder move.

With this notion, applying the algorithm on $w \in S_n$ can be rewritten as

$$\widehat{P}(w) = (L_{1,1} \cdots L_{1,n-2}) \cdots (L_{n-3,1}L_{n-3,2})(L_{n-2,1})(\overleftarrow{\mathsf{Rothe}(w)}) \tag{3}$$

In words, we iterate through $i = n-2, \cdots, 2, 1$. For each i, we iterate through $c = n-1-i, \cdots, 2, 1$ and apply $L_{i,c}$.

We start by observing a straightforward recursive property of this operator.

Remark 4.23. Fix $i, c \in \mathbb{Z}_{>0}$ and let D be a diagram. Suppose $(i, c) \notin D$ and $(i, c+1) \notin D$.

- Suppose $(i+1, c) \in D$ and $(i+1, c+1) \notin D$. Let D' be the diagram obtained by moving $(i+1, c)$ to $(i, c+1)$ in D. If $L_{i+1,c}(D') \neq D'$, we know $L_{i,c}(D) = L_{i+1,c}(D')$. Otherwise, $L_{i,c}(D) = D' \sqcup \{(i+1, c)\}$. Informally, in this case, $L_{i,c}$ behaves as if $L_{i+1,c}$ after the regular ladder move on $(i+1, c)$.
- Suppose $(i+1, c) \in D$ and $(i+1, c+1) \in D$. Then intuitively, $L_{i,c}$ behaves as if row $i+1$ is ignored: Let D' be obtained from D by removing $(i+1, c)$ and $(i+1, c+1)$. If $(i+1, c+1) \notin L_{i+1,c}(D')$, $L_{i,c}(D) = L_{i+1,c}(D') \sqcup \{(i+1, c), (i+1, c+1)\}$. Otherwise, $L_{i,c}(D) = L_{i+1,c}(D') \sqcup \{(i+1, c), (i, c+1)\}$.

We are primarily interested in applying $L_{i,c}$ to a diagram in the following case.

Definition 4.24. We say the operator $L_{i,c}$ *acts initially on* D if D is fixed by $L_{i+1,c}$.

Eventually, we will show all $L_{i,c}$ in our algorithm acts initially. We first derive a few properties when $L_{i,c}$ acts initially on D.

Lemma 4.25. *Suppose $L_{i,c}$ acts initially on D and $L_{i,c}$ moves at least one cell. We let $(r_1, c), \cdots, (r_k, c)$ be the cells moved where $r_1 < \cdots < r_k$. Let $r_0 = i$. Then we know the cell (r_j, c) is moved to $(r_{j-1}, c+1)$ for $j \in [k]$. Thus, $\mathsf{wt}(L_{i,c}(D))$ is obtained from $\mathsf{wt}(D)$ by adding 1 to the i^{th} entry.*

Proof. If $L_{i,c}$ moves (r_1, c) to $(r', c+1)$ for some $r' > i$, then $L_{i+1,c}$ will also move (r_1, c) to $(r', c+1)$. This contradicts our assumption that $L_{i,c}$ acts initially on D. Thus, $L_{i,c}$ moves (r_1, c) to $(i, c+1)$.

For $j > 1$, when (r_j, c) moves, (r_{j-1}, c) and $(r_{j-1}, c+1)$ must both be empty since the cell in (r_{j-1}, c) just performed a ladder move. Therefore (r_j, c) must be moved to $(r', c+1)$ for some $r' \geqslant r_{j-1}$. However, $r' > r_{j-1}$ contradicts the assumption that $L_{i,c}$ acts initially on D, so $r' = r_{j-1}$. $\square$

To better describe the effect of $L_{i,c}$ when it acts initially, we introduce the following notion.

Definition 4.26. The *(i, c)-initial segment* of a diagram D is the set of (r, c) such that $(r', c) \in D$ for all $i \leqslant r' \leqslant r$.

This notion characterizes the destination of cells moved by $L_{i,c}$ when it acts initially.

Lemma 4.27. *Suppose $L_{i,c}$ acts initially on D. Then it moves cells to the $(i, c+1)$-initial segment of $L_{i,c}(D)$.*

Proof. Let $(r_1, c), (r_2, c), \ldots, (r_k, c)$ where $r_1 < r_2 < \cdots < r_k$ be the cells of D moved by $L_{i,c}$. Let $r_0 = i$. By Lemma 4.25, for $j \in [k]$, (r_j, c) is moved to $(r_{j-1}, c+1)$. We show (r_{j-1}, c) is in the $(j, c+1)$-initial segment of $L_{i,c}(D)$ by induction on j. For the base case, $(r_0, c+1) = (i, c+1)$ is clearly in the $(j, c+1)$-initial segment of $L_{i,c}(D)$.

For $j > 1$. assume $(r_{j-2}, c+1)$ is in the $(i, c+1)$-initial segment of $L_{i,c}(D)$. Since (r_{j-1}, c) is moved to $(r_{j-2}, c+1)$, we know $(r', c+1) \in L_{i,c}(D)$ for any $r_{j-2} < r' < r_{j-1}$. Thus, $(r_{j-1}, c+1)$ is in the $(i, c+1)$-initial segment of $L_{i,c}(D)$. □

We can also use "initial segment" to characterize what cells can be moved by $L_{i,c}$ when it acts initially.

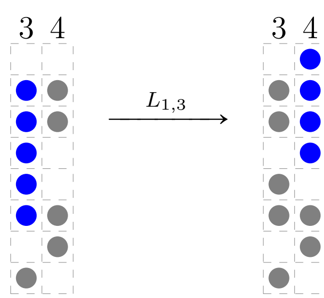
Lemma 4.28. *Suppose $L_{i,c}$ acts initially on D. If $(i, c) \in D$, then D is fixed by $L_{i,c}$. Otherwise, a cell $(r, c) \in D$ is moved by $L_{i,c}$ if and only if it is in the $(i+1, c)$-initial segment of D and $(r, c+1) \notin D$.*

Proof. The lemma is immediate when $(i, c) \in D$. Otherwise, let $(r_1, c), \cdots, (r_k, c) \in D$ be the cells moved by $L_{i,c}$ where $r_1 < \cdots < r_k$. Let $r_0 = i$. Clearly, $(r_j, c+1) \notin D$ for each $j \in [k]$. We prove (r_j, c) is in the $(i+1, c)$-initial segment of D by induction. First, by Lemma 4.25, (r_1, c) is moved to $(r_0, c+1)$, so $(r', c) \in D$ for $r_0 = i < r' < r_1$. In other words, (r_1, c) is in the $(i+1, c)$-initial segment of D. For $j > 1$, by Lemma 4.25, (r_j, c) is moved to $(r_{j-1}, c+1)$, so $(r', c) \in D$ for $r_{j-1} < r' < r_j$. The inductive step is finished since (r_{j-1}, c) is in the $(i+1, c)$-initial segment of D.

Now assume (r, c) is a cell in the $(i+1, c)$-initial segment of D and $(r, c+1) \notin D$. Assume toward contradiction that (r, c) is not moved by $L_{i,c}$. Take the smallest such r. Since $L_{i,c}$ moves (r_j, c) to (r_{j-1}, c), we know $(r', c+1) \in D$ for any $r_{j-1} < r' < r_j$. Thus, we cannot have $r_{j-1} < r < r_j$ for $j \in [k]$. Since (r, c) is not moved, we know r is not $r_1, \cdots, r_k$. Thus, $r > r_k$. By the minimality of r, $(r', c), (r', c+1) \in D$ for $r_k < r' < r$. Thus, $L_{i,c}$ moves (r_k, c), it can perform a ladder move at (r, c). Contradiction. □

The following example is a demonstration of the previous two lemmas related to initial segments.

Example 4.29. Let D be a diagram whose column 3 and 4 look like the picture on the left. Notice that D will be fixed by $L_{2,3}$. After applying $L_{1,3}$, these two columns look like the picture on the right:



We color the $(2, 3)$-initial segment of D and $(1, 4)$-initial segment of $L_{1,3}(D)$. Notice that $L_{1,3}$ move cells to the $(1, 4)$-initial segment of $L_{1,3}(D)$. Also notice that cells in column 3 is moved if and only if it is in the $(2, 3)$-initial segment of D and has no cell on its right.

We also have the "converse statement" of Lemma 4.28.

Lemma 4.30. *Suppose $(i, c) \notin D$. If $L_{i,c}$ only moves cells in the $(i+1, c)$-initial segment of D, then it acts initially on D.*

Proof. Suppose to the contrary that D is not fixed by $L_{i+1,c}$. Let (r,c) be the first cell moved by $L_{i+1,c}$. Clearly, (r,c) is not in the $(i+1,c)$-initial segment of D and it will also be moved by $L_{i,c}$. □

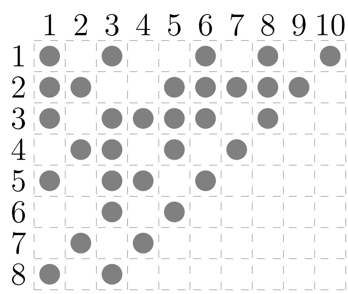
We introduce more definitions that captures the structure of columns for intermediate diagrams during our algorithm.

Definition 4.31. We say a diagram D is *(i,c)-paired* if the following are satisfied:

- Take any cell $(R,c) \in D$ with $i \leqslant R$ and $(R,c+1) \notin D$. There exists $(r,c+1) \in D$ with $i \leqslant r < R$ and $(r,c) \notin D$. Moreover, $(r',c),(r',c+1) \in D$ for any $r < r' < R$.
- Take any cell $(r,c+1) \in D$ with $i \leqslant r$ and $(r,c) \notin D$. There exists $(R,c) \in D$ with $r < R$ and $(R,c+1) \notin D$. Moreover, $(r',c),(r',c+1) \in D$ for any $r < r' < R$.

Remark 4.32. Notice that if D is (i,c)-paired, then $L_{i,c}$ fixes D.

Example 4.33. Consider the following diagram D.

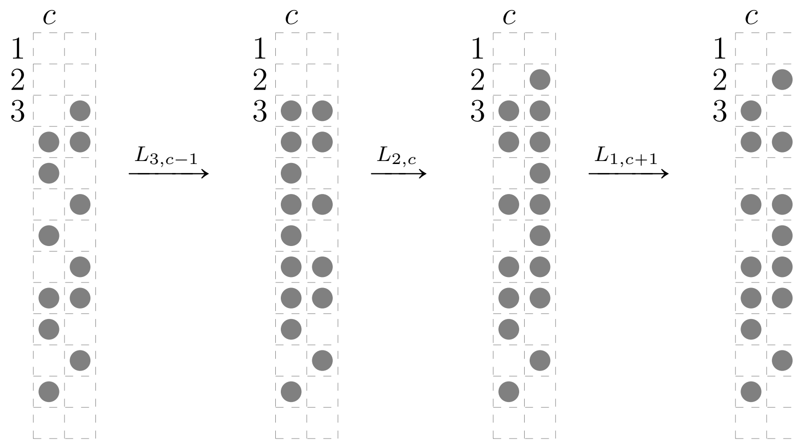


Then D has the following properties: $(1,5)$-paired, $(1,9)$-paired, $(4,1)$-paired, $(6,1)$-paired.

We have the following lemma regarding this new notion.

Lemma 4.34. *Let diagram D be $(3,c)$-paired and $(2,c+1) \notin D$. We consider the action of $L_{1,c+1}L_{2,c}L_{3,c-1}$ on D. Assume $L_{3,c-1}$ and $L_{2,c}$ act initially. Let $(r_1,c),\cdots,(r_m,c)$ be the cells moved by $L_{2,c}$ with $r_1 < \cdots < r_m$ and let $r_0 = 2$. We further assume $L_{1,c+1}$ moves $(r'_1,c+1),\cdots,(r'_m,c+1)$ with $r_{i-1} \leqslant r'_i < r_i$. Then $D' = L_{1,c+1}L_{2,c}L_{3,c-1}(D)$ is $(2,c)$-paired.*

Example 4.35. Consider the action of $L_{1,c+1}L_{2,c}L_{3,c-1}$ on D whose column c and $c+1$ are depicted in the left-most figure. We see D is $(3,c)$-paired. The action of $L_{2,c}$ and $L_{1,c+1}$ satisfy the condition in Lemma 4.34: For instance, $L_{2,c}$ moves $(5,c)$ to $(2,c+1)$ and there is a unique cell $(r,c+1)$ moved by $L_{1,c+1}$ with $2 \leqslant r < 5$, namely $(3,c+1)$. Then by the Lemma, we know $L_{1,c+1}L_{2,c}L_{3,c-1}(D)$, whose column c and $c+1$ are depicted in the right-most figure, is $(2,c)$-paired.



Proof. Say (t,c) is the bottom-most cell in the $(2,c)$-initial segment of $L_{3,c-1}(D)$. Since $L_{3,c-1}$ acts initially on D, it will only move cells to the $(2,c)$-initial segment by Lemma 4.27. Since $L_{2,c}$ acts initially on D, it will only move cells in the $(2,c)$-initial segment by Lemma 4.28. Then by our assumption in the lemma, $L_{1,c+1}$ also moves cells above row t. Thus, D and D' agreed under row t in column c and $c+1$. Now we check D' is $(2,c)$-paired.

Take (R,c) in D' such that $R \geqslant 2$ and $(R,c+1) \notin D'$. We find the r satisfying the condition in the definition of $(2,c)$-paired by considering two cases.

- If $R > t$, then $(R,c) \in D$ and $(R,c+1) \notin D$. Since D is $(3,c)$-paired, we can find $(r,c+1) \in D$ such that $2 \leqslant r < R$, $(r,c) \notin D$ and $(r',c),(r',c+1) \in D$ for $r < r' < R$. It remains to show $r > t$. If not, (r,c) is in the $(2,c)$-initial segment of $L_{3,c-1}(D)$, then so is (R,c), contradicting to $R > t$.
- If $R \leqslant t$, then $(R,c) \in L_{3,c-1}(D)$. If $(R,c+1) \notin L_{3,c-1}(D)$, by Lemma 4.28, $L_{2,c}$ moves (R,c). Since (R,c) is in D', we know it is the last cell moved by $L_{2,c}$, so $R = r_m$. By Lemma 4.25, $L_{2,c}$ moves (r_m,c) to $(r_{m-1},c+1)$. We have $(r_{m-1},c) \notin D'$. By our assumption on $L_{1,c-1}$, it does not make a regular ladder move on cells between row r_{m-1} and row r_m. Thus, we may pick $r = r_{m-1}$.

  Now assume $(R,c+1) \in L_{3,c-1}(D)$. Then, $L_{1,c+1}$ moves $(R,c+1)$, so $R = r'_i$ for some $i \in [m-1]$. We know $L_{2,c}$ moves (r_i,c) to $(r_{i-1},c+1)$. By our assumption on $L_{1,c+1}$, $r_{i-1} < r'_i$ and $L_{1,c+1}$ does not make a move between row r_{i-1} and r'_i. Thus, we may pick $r = r_{i-1}$.

Take $(r,c+1)$ in D' such that $r \geqslant 2$ and $(r,c) \notin D'$. We find the R satisfying the condition in the definition of $(2,c)$-paired by considering two cases.

- If $r > t$, then $(r,c+1) \in D$ and $(r,c) \notin D$. Moreover, since $(2,c+1) \notin D$, we know $r \geqslant 3$. By D is $(3,c)$-paired, we can find $R > r > t$ such that $(R,c) \in D$, $(R,c+1) \notin D$ and $(r',c),(r',c+1) \in D$ for $r < r' < R$.
- If $r \leqslant t$, then $(r,c) \in L_{3,c-1}(D)$. We know $L_{2,c}$ performs a regular ladder move on (r,c), so $r = r_i$ for some $i \in [m-1]$. We know $r_i < r'_{i+1} < r_{i+1}$ and $(r',c),(r',c+1) \in L_{2,c}L_{3,c-1}(D)$ for $r_i < r' < r_{i+1}$. If $i+1 < m$, then $L_{1,c+1}$ makes a regular ladder move on $(r'_{i+1},c+1)$. We have $(r'_{i+1},c) \in D'$ and $(r_{i+1},c+1) \notin D'$. We may pick $R = r'$. If $i+1 = m$, then $L_{1,c+1}$ makes a K-ladder move on $(r'_{i+1},c+1)$. We may pick $R = r_{i+1}$. $\square$

The last piece of our preparation work is the following observation.

Remark 4.36. Notice that $L_{i,c}$ and $L_{i',c'}$ commute if $|c-c'|>1$. Therefore, we know applying

$$L_{1,1}L_{1,2}\cdots L_{1,n-2} \quad L_{2,1}L_{2,2}\cdots L_{2,n-3}$$

is the same as applying

$$L_{1,1}\ \ L_{1,2}L_{2,1}\ \ L_{1,3}L_{2,2}\ \ \cdots\ \ L_{1,n-4}L_{2,n-3}\ \ L_{1,n-2}L_{2,n-3}.$$

Moreover, each $L_{i,c}$ behaves the same in both expressions.

Now we embark on proving Proposition 4.17 and Corollary 4.19. We start by introducing two claims which will imply Proposition 4.17 and Corollary 4.19 respectively. For a diagram D, let $D^{\downarrow k}$ be the diagram obtained by shifting all cells of D downward by k. We claim:

- Claim 1: Take $N\in\mathbb{Z}_{>0}$ and $w\in S_N$. Consider
$$(L_{1,2}L_{2,1})\cdots(L_{1,N-2}L_{2,N-3})(L_{1,N}L_{2,N-1})(\widehat{P}(w)^{\downarrow 2}). \tag{4}$$
Take any $c\in[N-1]$. Then $L_{2,c}$ and $L_{1,c+1}$ moves the same number of cells. More specifically, suppose $L_{2,c}$ moves a cell (r,c) to $(\hat{r},c+1)$. Then there exists a unique r' such that $\hat{r}\leqslant r'<r$ and $(r',c+1)$ is moved by $L_{1,c+1}$. In addition, after the action of $L_{1,c+1}$, the diagram is $(2,c)$-paired.
- Claim 2: Take $N\in\mathbb{Z}_{>0}$ and $w\in S_N$. Consider
$$L_{1,1}\cdots L_{1,N-1}(\widehat{P}(w)^{\downarrow 1}).$$
Each $L_{1,c}$ acts initially.

We will inductively show both claims hold for all N. The induction is based on Lemma 4.37 and Lemma 4.38.

Lemma 4.37. *Suppose Claim 1 and Claim 2 hold for $N\leqslant n$, then Claim 2 holds for $N=n+1$.*

Proof. Suppose $w=(b,u)\in S_{n+1}$. Let D be the diagram obtained by putting b left-justified cells in the second row of $\widehat{P}(u)^{\downarrow 2}$. Then $\widehat{P}(w)^{\downarrow 1}=L_{2,1}L_{2,2}\cdots L_{2,n-1}(D)$ and each $L_{2,c}$ acts initially by Claim 2 for u. By Remark 4.36, we may write $L_{1,1}\cdots L_{1,N-1}(\widehat{P}(w)^{\downarrow 1})$ as

$$L_{1,1}\cdots L_{1,N-1}\,L_{2,1}\cdots L_{2,n-1}(D)=(L_{1,2}L_{2,1})\cdots(L_{1,N-2}L_{2,N-3})(L_{1,N}L_{2,N-1})(D). \tag{5}$$

Clearly, for $c\leqslant b$, $L_{1,c}$ acts initially on $\widehat{P}(w)^{\downarrow 1}$. Now take $c>b$. We know the $L_{1,c}$ behaves the same in both sides of (5). By Lemma 4.30, it is enough to show each $L_{1,c}$ on the right hand side moves cells in the $(2,c)$-initial segment. Since $L_{2,c-1}$ acts initially, by Lemma 4.27, $L_{2,c-1}$ move cells into the $(2,c)$-initial segment. Then by claim 1 of u, $L_{1,c}$ moves cells in the $(2,c)$-initial segment. □

Lemma 4.38. *Suppose Claim 1 holds for $N\leqslant n$ and Claim 2 holds for $N\leqslant n+1$, then Claim 1 holds for $N=n+1$.*

Proof. Since Claim 2 holds for $N\leqslant n+1$, each $L_{1,c}$ and $L_{2,c}$ in (4) acts initially by Remark 4.36. We prove Claim 1 by induction on $c=n,\cdots,2,1$. The base case with $c=n$ is trivial.

Suppose $c\in[n-1]$. Let D' be the diagram right before applying $L_{2,c}$ in (4). By our inductive hypothesis for $c+1$, D' is $(2,c+1)$-paired. Now apply $L_{2,c}$ to D'. Let $(r_1,c),\cdots,(r_k,c)$ be the cells moved by $L_{2,c}$. Let $r_0=2$. For $j\in[k]$, by Lemma 4.25, (r_j,c) is moved to $(r_{j-1},c+1)$. By Lemma 4.27, $(r_{j-1},c+1)$ is in the $(2,c+1)$-initial segment of $L_{2,c}(D)$. We consider two cases.

- If $(r_{j-1}, c+2) \notin D'$, then $(r_{j-1}, c+1)$ will be moved by $L_{1,c+1}$ by Lemma 4.28. For $r_{j-1} < r' < r$, by D' is $(2, c+1)$-paired, we know $(r', c+1), (r', c+2) \in D'$. By Lemma 4.28, $L_{1,c+1}$ will not move $(r', c+1)$.
- Now assume $(r_{j-1}, c+2) \in D'$. Since D' is $(2, c+1)$-paired and $(r_{j-1}, c+) \notin D'$, we can find $R > r_{j-1}$ such that $(R, c+1) \in D'$, $(R, c+2) \notin D$ and $(r', c+1), (r', c+2) \in D'$ for any $r_{j-1} < r' < R$. We know $(r_j, c+1) \notin D'$, so $R < r_j$. For $R < r' < r_j$, since $(r', c+1) \in D'$ and D' is $(2, c+1)$-paired, we must have $(r', c+2) \in D'$. By 4.28, $(R, c+1)$ is the unique cell moved during $L_{1,c+1}$ between row r_{j-1} and row r_j.

Now we show $L_{1,c+1}$ and $L_{2,c}$ move the same number of cells, we already know $L_{1,c+1}$ makes exactly one move between row r_{j-1} and row r_j inclusively for $j \in [k]$. We just need to show $L_{1,c+1}$ does not move any $(r, c+1)$ for any $r > r_k$. Notice that $(r_k, c+1) \notin L_{2,c}(D')$, so $(r, c+1)$ is not in the $(2, c+1)$-initial segment of $L_{2,c}(D')$. By Lemma 4.28, $(r, c+1)$ will not be moved.

It remains to check $L_{1,c+1}L_{2,c}(D')$ is $(2, c)$-paired. Write w as (b, u). Let D be the diagram obtained by putting b left-justified cells in row 3 of $\widehat{P}(u)^{\downarrow 3}$. Then

$$\widehat{P}(w)^{\downarrow 2} = L_{3,1}L_{3,2}\cdots L_{3,n-1}(D)$$

By Remark 4.36,

$$\begin{aligned} &(L_{1,2}L_{2,1})\cdots(L_{1,n+1}L_{2,n})(\widehat{P}(w)^{\downarrow 2}) \\ =&(L_{1,2}L_{2,1})\cdots(L_{1,n+1}L_{2,n})(L_{3,1}L_{3,2}\cdots L_{3,n-1})(D) \\ =&(L_{1,2}L_{2,1})(L_{1,3}L_{2,2}L_{3,1})\cdots(L_{1,n+1}L_{2,n}L_{3,n-1})(D) \end{aligned}$$

If $c > b$, then $(3, c) \notin D$. By claim 1 of u, after $L_{2,c+1}$ the diagram is $(3, c)$-paired. Therefore, by Lemma 4.34, after $L_{1,c+1}$ the diagram is $(2, c)$-paired.

Now consider $c \leqslant b$, so $(3, c) \in D$. We consider three cases:

- Case 1: $(3, c)$ is moved by $L_{2,c}$ and not the last cell moved by $L_{2,c}$. Then $L_{2,c}$ performs a regular ladder move on $(3, c)$ moving it to $(2, c+1)$. Later, $L_{1,c+1}$ will move $(2, c+1)$. Since $L_{1,c+1}$ and $L_{2,c}$ moves the same number of cells, we know $L_{1,c+1}$ makes a regular ladder move on $(2, c+1)$. By Remark 4.23, the action of $L_{1,c+1}L_{2,c}$ is the same as first moving $(3, c)$ to $(1, c+2)$, and then perform $L_{2,c+1}L_{1,c+2}$. By Claim 1 of u, the diagram after applying $L_{1,c+1}$ is $(3, c)$-paired. Since $(2, c), (2, c+1)$ are not in the diagram, it is $(2, c)$-paired.
- Case 2: $(3, c)$ is the last cell moved by $L_{2,c}$. Then $L_{2,c}$ performs a K-ladder move on $(3, c)$ moving it to $(2, c+1)$. Later, $L_{1,c+1}$ will move $(2, c+1)$. Since $L_{1,c+1}$ and $L_{2,c}$ moves the same number of cells, we know $L_{1,c+1}$ makes K-ladder move on $(2, c+1)$. By Remark 4.23, the action of $L_{1,c+1}L_{2,c}$ can be described as follows: Remove $(3, c)$, perform $L_{2,c+2}L_{3,c}$, and then add cells $(3, c)$, $(2, c+1)$ and $(1, c+2)$. By Claim 1 of u, before adding those three cells, the diagram is $(3, c)$-paired. Thus, after adding these three cells, the diagram is $(2, c)$-paired.
- If $(3, c)$ is not moved by $L_{2,c}$, then $(3, c+1) \in D$. By Remark 4.23, applying $L_{1,c+1}L_{2,c}$ is the same as applying $L_{2,c+2}L_{3,c}$ while ignoring row 3. By Claim 1 of u, after the action of $L_{1,c+1}$, the diagram is $(2, c)$-paired. □

Lemma 4.39. *Claim 1 and 2 hold for all $N \in \mathbb{Z}_{>0}$.*

Proof. The claims are obvious when $N = 1$. Then we prove by induction on N. The inductive step is given by Lemma 4.37 and Lemma 4.38. □

Corollary 4.40. *In (3), each $L_{i,c}$ acts initially.*

Proof. Suppose $w \in S_n$ and we prove the corollary by induction on n. Suppose $w = (b, u)$. Since the corollary holds for u, we know $L_{i,c}$ in (3) acts initially when $i > 2$. Finally, each $L_{1,c}$ acts initially by Claim 2. □

Now we may prove the main results of this subsection using the two claims.

Proof of Proposition 4.17. We induct on n. The base cases $n = 2$ is trivial. Now suppose $n > 2$ and take $v = (a, w) \in S_n$. Let D be the diagram obtained by putting a left-justified cells in row 1 of $\widehat{P}(w)^{\downarrow 1}$. The last iteration to compute $\widehat{P}(v)$ is to apply $L_{1,1} \cdots L_{1,n-2}L_{1,n-2}$ on D. For $c \in [a]$, since $L_{1,c}$ acts initially and $(1, c) \in D$, $L_{1,c}$ does not move any cells.

Now assume $c > a$. We want to show $L_{1,c}$ moves exactly $\mathsf{movecode}(w)_c$ cells. Let $w = (b, u)$ and let D' be the diagram obtained by putting b left-justified cells in the row 2 of $\widehat{P}(u)^{\downarrow 2}$. Then,

$$\begin{aligned}
&L_{1,1} \cdots L_{1,n-2}L_{1,n-1}(D) \\
=&(L_{1,1} \cdots L_{1,n-2}L_{1,n-1})(L_{2,1} \cdots L_{2,n-3}L_{2,n-2})(D') \\
=&(L_{1,1})(L_{1,2}L_{2,1}) \cdots (L_{1,n-1}L_{2,n-2})(D')
\end{aligned}$$

For $c > b$, by our induction hypothesis, applying $L_{2,c}$ moves exactly $\mathsf{movecode}(u)_c$ cells. Then by Claim 1, applying $L_{1,c+1}$ to D also moves exactly $\mathsf{movecode}(u)_c$ cells. Therefore the number of cells moved by $L_{1,c+1}$ is $\mathsf{movecode}(u)_c = \mathsf{movecode}(w)_{c+1}$. Now clearly each $L_{2,c}$ does not move any cells for $c \in [b]$. We know $L_{1,b+1}$ also moves no cells since the $(2, b+1)$-initial segment is empty. Therefore $L_{1,b+1}$ moves $0 = \mathsf{movecode}(w)_{b+1}$ cells.

Let c_0 be the largest in $[b]$ such that $\mathsf{movecode}(u)_{c_0} = 0$. Say $c_0 = 0$ if no such c_0 exists. For $c \in [b]$, by Lemma 4.20, we have

$$\mathsf{movecode}(w)_c = \begin{cases} \mathsf{movecode}(u)_c + 1 & \text{if } c \geqslant c_0. \\ \mathsf{movecode}(u)_c & \text{otherwise} \end{cases}$$

We first inductively show that for $c = b, \cdots, c_0 + 1$, there is no cell at $(2, c+1)$ right before the action of $L_{1,c}$, so $L_{1,c}$ moves $(2, c)$. Moreover, $L_{1,c}$ moves $\mathsf{movecode}(w)_c > 2$ cells, so the move on $(2, c)$ is a regular ladder move. For $c = b$, we know $(2, b+1)$ is always empty. For $c_0 < c < b$, we know $L_{1,c+1}$ makes a regular ladder move on $(2, c+1)$, so $(2, c+1)$ is empty right before the action of $L_{1,c}$. Now for $c = b, \cdots, c_0 + 1$, after $L_{1,c}$ moves $(2, c)$, it behaves as if $L_{2,c}$ by Remark 4.23. Thus, the total number of cells moved is $\mathsf{movecode}(u)_c + 1 = \mathsf{movecode}(w)_c$.

Now consider L_{1,c_0} when $c_0 > 0$. Right before its action, $(2, c_0+1)$ is empty. Thus, L_{1,c_0} will first move $(2, c_0)$ to $(1, c_0+1)$. After that, the number of cells it moves is $\mathsf{movecode}(u)_{c_0}$, which is zero. Thus, the move on $(2, c_0)$ is a K-ladder move. Also, L_{1,c_0} moves $1 = \mathsf{movecode}(w)_{c_0}$ cell.

Finally, we prove by induction that for $c = c_0 - 1, \cdots, 1$, right before the action of $L_{1,c}$, the diagram contains $(2, c)$ and $(2, c+1)$. For the base case, right before the action of L_{1,c_0-1}, we know $(2, c_0)$ is in the diagram. Now assume right before the action of $L_{1,c}$, the diagram contains $(2, c)$ and $(2, c+1)$ for some $c < c_0$. Then $L_{1,c}$ will not move $(2, c)$. After the action of $L_{1,c}$, we know $(2, c)$ is still in the diagram. The inductive step is finished. Now by Remark 4.23, the action of $L_{1,c}$ moves the same number of cells as $L_{2,c}$ on the diagram without $(2, c)$ and $(2, c+1)$. Thus, $L_{1,c}$ makes $\mathsf{movecode}(u)_c = \mathsf{movecode}(w)_c$ moves. □

Proof of Corollary 4.19. Implied by Corollary 4.40 and Lemma 4.25. □

5. Bumpless vertical-less pipedreams

In this section, we introduce two new combinatorial objects. Marked bumpless pipedreams that naturally recast pipedreams and bumpless vertical-less pipedreams that gives the first direct combinatorial formula for the top degree homogeneous component of Grothendieck polynomials $\mathfrak{G}_w(\mathbf{x})$.

5.1. Background.

We define $\widehat{\mathfrak{G}}_w(\mathbf{x})$ to be the top degree homogeneous component of $\mathfrak{G}_w(\mathbf{x})$. Therefore, $\widehat{\mathfrak{G}}_w(\mathbf{x}) = \sum_P \mathsf{wt}_P(\mathbf{x})$ where the sum is over all $P \in \mathsf{PD}(w)$ with $\mathsf{raj}(w)$ weighty tiles. We start off with the following two properties of the raj statistic and an additional lemma regarding pipedreams.

Proposition 5.1 ([PSW21, Proposition 3.8])**.** *For $w \in S_n$, $\mathsf{raj}(w) = \mathsf{maj}(w)$ if and only if w is fireworks. Here, the major index of w is defined as*

$$\mathsf{maj}(w) = \sum_{\{i: w(i) > w(i+1)\}} i$$

Corollary 5.2 ([PSW21, Corollary 4.5])**.** *For $w \in S_n$, $\mathsf{raj}(w) = \mathsf{raj}(w^{-1})$.*

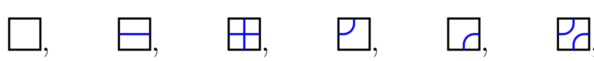
Lemma 5.3. *Say three pipes enter a row of a PD from the bottom: Pipe a enters on the left of pipe b and pipe b enters on the left of pipe c. Suppose pipe a and pipe b have not crossed, but pipe a and pipe c have crossed. Then pipe b and pipe c must have crossed.*

Proof. Since pipe a enters the row on the left of pipe b and they did not cross, we have $a < b$. Since pipe a enters the row on the left of pipe c and they have crossed, we have $c < a$. Thus, $c < a < b$. Since pipe b enters the row on the left of pipe c, they have crossed. □

5.2. Marked vertical-less pipedreams.

We introduce combinatorial objects which we call marked vertical-less pipedreams (MVPD). An MVPD can be obtained by removing certain pipes from a PD. We rephrase the formulas in Theorem 3.9 and obtain MVPD formulas for $\mathfrak{G}_w(\mathbf{x})$ and $\mathfrak{G}_w(\mathbf{x}, \mathbf{y})$ in Corollary 5.10.

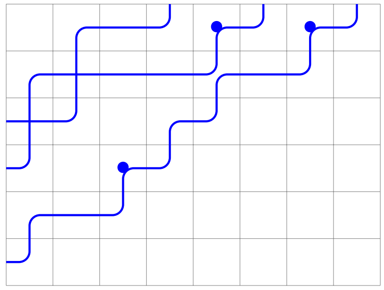
Definition 5.4. A *vertical-less pipedream* (VPD) consists of the following six tiles:



on an $n \times n$ grid. Notice that we are not using the vertical tile . The pipes of a VPD enter from the left edge of the $n \times n$ grid and exit from the top edge. We trace pipes from left to top in the same way as PDs. The pipe entering from row p is called pipe p. A *marked vertical-less pipedreams* (MVPD) is a VPD where some are marked as . The pipe in a marked tile must have a on the left of this tile.

The *column-to-row code* of a MVPD M is a sequence of n numbers. If there is no pipe exiting at column c of M, then the c^{th} entry is 0. Otherwise, say pipe r exits in column c, then the c^{th} entry is r. When drawing a MVPD, we omit blank rows on the bottom and blank columns to the right.

Example 5.5. Suppose $n = 8$. The following is a MVPD which has column-to-row code $(0, 0, 0, 4, 0, 3, 0, 6)$. Notice that $(1, 2)$, $(2, 1)$, and $(5, 1)$ cannot be marked while $(3, 4)$ may or may not be marked.



For a MVPD M, we let $\mathsf{wty}(M)$ be the set of (i,j) that is , or in M. We define

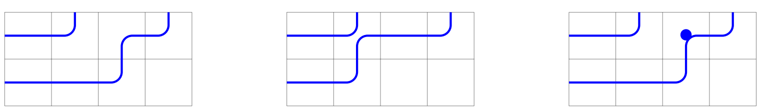
$$\mathsf{wt}_M(\mathbf{x}) = \prod_{(i,j)\in\mathsf{wty}(M)} x_i, \quad \mathsf{wt}_M(\mathbf{x},\mathbf{y}) = \prod_{(i,j)\in\mathsf{wty}(M)} (x_i + y_j - x_i y_j)$$

We associate certain MVPDs to each permutation $w \in S_n$. The *left-to-right maximums* of $w \in S_n$ are the numbers $w(i)$ such that $w(j) < w(i)$ for all $j < i$. For instance, the left-to-right maximums of the permutation with one-line notation 2143 are 2 and 4.

Remark 5.6. Notice that in the one-line notation of a permutation, its left-to-right maximums must increase from left to right. Consequently, in $P \in \mathsf{PD}(w)$, pipes labeled by left-to-right maximums of w^{-1} cannot cross.

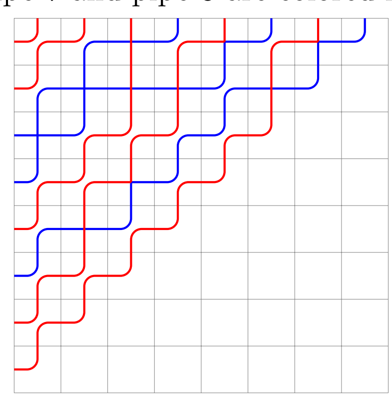
Take $w \in S_n$. We start from $(w^{-1}(1), \cdots, w^{-1}(n))$ and turn the left-to-right maximums of w^{-1} into 0. Let $\alpha'(w)$ be the resulting sequence. Finally, define $\mathsf{MVPD}(w)$ as the set of MVPDs with column-to-row code $\alpha'(w)$.

Example 5.7. Take $w \in S_n$ such that w^{-1} has one-line notation 3142. We have $\alpha'(w) = (0,1,0,2)$. The set $\mathsf{MVPD}(w)$ has the following three elements:



We describe a bijection from $\mathsf{PD}(w)$ to $\mathsf{MVPD}(w)$. Take $P \in \mathsf{PD}(w)$ for some $w \in S_n$. Let $p_1, \cdots, p_k$ be the left-to-right of w^{-1}. We may remove the pipes $p_1, \cdots, p_k$. If a becomes a after the removal, we mark it as . Let $\Phi(P)$ be the resulting tiling.

Example 5.8. Suppose $n = 8$. Consider $w \in S_8$ where w^{-1} has one-line notation 12547386. The left-to-right maximums of w^{-1} are $1, 2, 5, 7$ and 8. We consider the following $P \in \mathsf{PD}(w)$ where pipe 1, pipe 2, pipe 5, pipe 7 and pipe 8 are colored red.



Readers may check $\Phi(P)$ would be the MVPD in Example 5.5.

Proposition 5.9. *The map Φ is a bijection from* $\mathsf{PD}(w)$ *to* $\mathsf{MVPD}(w)$ *that preserves* $\mathsf{wty}(\cdot)$

Proof. Take any $P \in \mathsf{PD}(w)$ and consider $\Phi(P)$. First, if both P and $\Phi(P)$ has pipe p, then it travels the same in P and $\Phi(P)$. We now check $\Phi(P) \in \mathsf{MVPD}(w)$.

- We make sure $\Phi(P)$ has no . Suppose to the contrary that $\Phi(P)$ at (i, j) is a , then P must have a at (i, j). Let pipe p (resp. q) be the pipe going horizontally (resp. vertically) in this tile. Then we know pipe p is removed by Φ, so p is a left-to-right maximum of w^{-1}. However, since pipe p and pipe q crossed in this tile, we know $p < q$ and q appears on the left of p in the one-line notation of w^{-1}, contradicting to p being a left-to-right maximum of w^{-1}.
- Assume pipe p has at (i, j) of $\Phi(P)$, we check pipe p has a before. We know P has a at (i, j). Let pipe q be the other pipe in (i, j) of P, so this pipe is removed by Φ. We know pipe p and pipe q already crossed before (i, j) in P, say at (i', j'). Then after removing pipe q, the (i', j') becomes in $\Phi(P)$.

We have checked $\Phi(P)$ is a valid MVPD. Clearly, $\Phi(P)$ has column-to-row code $\alpha'(w)$, so it is in $\mathsf{MVPD}(w)$. We check Φ preserves $\mathsf{wty}(\cdot)$. Take a in P. We check it becomes a weighty tile in $\Phi(P)$. Say pipe p exits from the top and pipe q exits from the right of this . By 5.6, it is impossible that both pipe p and pipe q are removed by Φ, so this will not become a . It also cannot become a : If so, we know q is a left-to-right maximum in w^{-1}, $q < p$, and q ends up on the right of p in w^{-1}. This is a contradiction. It is also obvious that this cannot be mapped to or by the rules of Φ. Therefore Φ maps to weighty tiles. On the other hand, for any in P, they cannot be mapped to , , or by the rules of Φ.

Thus, Φ is a $\mathsf{wty}(\cdot)$ preserving map from $\mathsf{PD}(w)$ to $\mathsf{MVPD}(w)$. It remains to construct its inverse. Take $M \in \mathsf{MVPD}(w)$. We change the cell (i, j) based on the following:

- If it is or , it becomes .
- If it is , we know $i + j \leqslant n + 1$. If $i + j < n$, we change it into .
- If it is , we turn it into .
- Finally, suppose it is . If $i + j < n$, we turn it into . If $i + j = n$, we turn it into .

Clearly, we obtain a PD P by adding pipes to each tile. We claim for each pipe in M, it goes in the same way in both P and M. In addition, the added pipes in P belong to pipes which do not exist in M. We prove by induction on the tiles from bottom to top, and left to right in each row. Consider the tile (i, j)

- If (i, j) is a containing pipe p in M, it becomes a in P. We need to verify that pipe p goes horizontally in (i, j) of P. Let pipe q be the pipe entering from the bottom of (i, j) in P, so pipe q does not exist in P. Assume toward contradiction that p does not go horizontally in (i, j). Then pipe p and pipe q have already crossed, where the pipe p travels vertically. Then the corresponding cell in M would be a , which is impossible.
- If (i, j) is a containing pipe p in M, it becomes a in P. We need to verify that pipe p does not go vertically in (i, j) of P. Let pipe q be the pipe entering from the

left of (i,j) in P, so pipe q does not exist in P. We need to show pipe p and q have crossed before. Since (i,j) is in M, we may find a containing pipe p under row i. In P, it becomes a where pipe p crosses with some added pipe, say pipe t. If $t = q$, we are done. Otherwise, we know the added pipes cannot cross. Thus, the three pipes enter row i with the order t, q, p from left to right. By Lemma 5.3, we know pipe q and p have crossed.

- The other cases of (i,j) is straightforward to check.

Say $P \in PD(u)$. The claim above says the k^{th} entry of $\alpha'(w)$, if non-zero, agrees with $u^{-1}(k)$. Since the added pipes are not crossing, we know $u^{-1}(k)$ is obtained from $\alpha'(w)$ by turning 0s into the missing numbers in increasing order, which yields w^{-1}. Thus, $u = w$ and the map defined above sends $\mathsf{MVPD}(w)$ to $\mathsf{PD}(w)$. It is clearly the inverse of Φ. $\square$

Corollary 5.10. *For $w \in S_n$, we have*

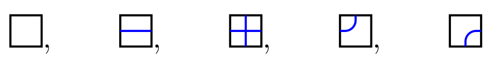
$$\mathfrak{G}_w(\boldsymbol{x}) = \sum_{M \in \mathsf{MVPD}(w)} (-1)^{|\mathsf{wty}(M)| - \ell(w)} \mathsf{wt}_M(\boldsymbol{x}),$$

$$\mathfrak{G}_w(\boldsymbol{x}, \boldsymbol{y}) = \sum_{M \in \mathsf{MVPD}(w)} (-1)^{|\mathsf{wty}(M)| - \ell(w)} \mathsf{wt}_M(\boldsymbol{x}, \boldsymbol{y}).$$

Proof. Follows from Theorem 3.9 and Proposition 5.9. $\square$

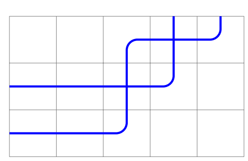
5.3. **Describing the BVPD formula.** We introduce *bumpless vertical-less pipedreams* (BVPD) as the first direct combinatorial formula for $\widehat{\mathfrak{G}}_w(\mathbf{x})$ when w is inverse fireworks. Specifically, we are interested in the special case when w is inverse fireworks because Pechenik, Speyer, and Weigandt [PSW21] showed that each $\widehat{\mathfrak{G}}_w(\mathbf{x})$ is an integer multiple of $\widehat{\mathfrak{G}}_u(\mathbf{x})$ for some inverse fireworks permutation u. Thus, for any $w \in S_n$, we may find its associated inverse fireworks permutation u and determine $\widehat{\mathfrak{G}}_w(\mathbf{x})$ using $\mathsf{BVPD}(u)$.

BVPDs consist of tilings where the following five types of tiles are placed



on an $n \times (n-1)$ grid. The pipes of a BVPD enter from the left edge and exit from the top edge. We trace pipes from left to top. For each , we trace the pipes in the same way as PDs and MVPDs. We name the pipe entering from row p as pipe p. When drawing a BVPD, we omit blank rows on the bottom and blank columns to the right.

Example 5.11. Let $n = 6$. The following is a BVPD



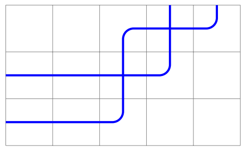
with pipe 2 going to column 5 and pipe 3 going to column 4.

The *column-to-row code* of a BVPD is a sequence of $n - 1$ numbers, defined similarly as that of a MVPD. The column-to-row code of the BVPD in Example 5.11 is $(0, 0, 0, 3, 2)$.

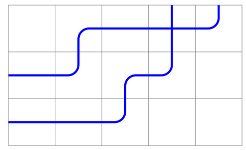
Take $w \in S_n$ be inverse fireworks. We obtain a sequence $\alpha(w)$ as follows. We start from the sequence $(w^{-1}(1), \cdots, w^{-1}(n))$ and set the first number in each decreasing run to be 0. Then $\alpha(w)$ is obtained by removing the first entry. Notice that $\alpha(w)$ can be obtained from

$\alpha'(w)$ by removing the first entry. Let $\mathsf{BVPD}(w)$ be the set of all BVPDs with column-to-row code $\alpha(w)$.

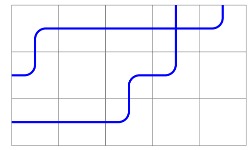
Example 5.12. Say $n = 6$ and w has one-line notation 165234. Thus, w^{-1} has one-line notation 145632 where $1, 4, 5$ and 6 are the first numbers in the decreasing runs. We have $\alpha(w) = (0, 0, 0, 3, 2)$, so $\mathsf{BVPD}(w)$ consists of all BVPDs with pipe 2 going to column 5, pipe 3 going to column 4, and there are no other pipes. There are six such BVPDs:



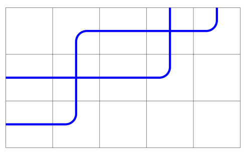
$x_1^2x_2^4x_3^3$



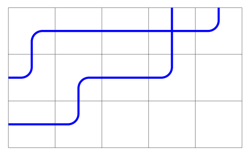
$x_1^3x_2^3x_3^3$



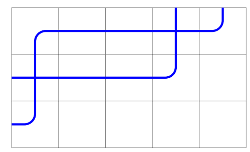
$x_1^4x_2^2x_3^3$



$x_1^3x_2^4x_3^2$



$x_1^4x_2^3x_3^2$



$x_1^4x_2^4x_3^1$

Finally, define the *weighty tiles* of a BVPD B, denoted as $\mathsf{wty}(B)$, as the set of (i, j) that is , or in B. Let the *weight* of B, denoted as $\mathsf{wt}_B(\mathbf{x})$, be the monomial $\Pi_{(i,j)\in\mathsf{wty}(B)}\, x_i$. We write the weight of each BVPD under itself in Example 5.12.

Theorem 5.13 ([CY24, Theorem 4.3])**.** *For inverse fireworks w, we have*

$$\widehat{\mathfrak{G}}_w(\boldsymbol{x}) = \sum_{B\in\mathsf{BVPD}(w)} \mathsf{wt}(B)$$

Continuing on Example 5.12. If w has one-line notation 165234, then

$$\widehat{\mathfrak{G}}_w(\mathbf{x}) = x_1^2x_2^4x_3^3 + x_1^3x_2^3x_3^3 + x_1^4x_2^2x_3^3 + x_1^3x_2^4x_3^2 + x_1^4x_2^3x_3^2 + x_1^4x_2^4x_3^1.$$

Theorem 5.14 ([CY24, Theorem 4.4])**.** *For w inverse fireworks, there exists a bijection Ψ from $\mathsf{BVPD}(w)$ to $\widehat{\mathsf{PD}(w)}$ that preserves the positions of weighty tiles.*

Roughly speaking, for $B \in \mathsf{BVPD}(w)$, $\Psi(B)$ is the pipedream with a at row i column j for each $(i, j) \in \mathsf{wty}(B)$ and no elsewhere. This result also characterizes the pipedreams of w with the maximal number of when w is inverse fireworks.

5.4. **Proof of Theorem 5.13 and Theorem 5.14.** We start with one simple property on the number of weighty tiles in a MVPD. For $w \in S_n$, define $r(w) := \sum_i i - 1$ where i ranges over all number such that the i^{th} number in $\alpha'(w)$ is non-zero.

Lemma 5.15. *Take $M \in \mathsf{MVPD}(w)$. Let k be the number of and in M. We have $|\mathsf{wty}(M)| = r(w) - k$.*

Proof. We first associate each tile in M that is not or to each pipe. These tiles must be , , , or . We associate each such tile to the pipe that exits from the right.

Take an arbitrary pipe p and suppose it goes to column c_p. In other words, the c^{th} number in $\alpha'(w)$ is p. For each column i, we count the number of cells associated with pipe p in this column:

- If the pipe p exits column i and goes to column $i+1$ (i.e. $1 \leqslant i < c_p$), there is exactly one tile associated with pipe p in column i.
- Otherwise (i.e. $i \geqslant c_p$), there is no tile associated with pipe p.

Now there are $c_p - 1$ tiles associated with the pipe p. Let k_p be the number of and associated with pipe p. The number of weighty tiles associated with p is $(c_p - 1) - k_p$. We have

$$\begin{aligned}|\mathsf{wty}(M)| &= \sum_{\text{pipes } p \text{ in } M} \mathsf{wt}(p) = \sum_{\text{pipes } p \text{ in } M} (c_p - 1) - k_p \\ &= \sum_{\text{pipes } p \text{ in } M} (c_p - 1) - \sum_{\text{pipes } p \text{ in } M} k_p = r(w) - k.\end{aligned}$$ $\square$

Let $\widehat{\mathsf{MVPD}(w)}$ be the subset of $\mathsf{MVPD}(w)$ with maximal number of weighty tiles. Recall that $\mathsf{raj}(w)$ is the degree of $\mathfrak{G}_w(\mathbf{x})$, so an element of $\widehat{\mathsf{MVPD}(w)}$ has $\mathsf{raj}(w)$ weighty tiles. We can describe $\widehat{\mathsf{MVPD}(w)}$ of inverse fireworks w as follows.

Lemma 5.16. *Let w be an inverse fireworks permutation. Then $\widehat{\mathsf{MVPD}(w)}$ consists of elements in $\mathsf{MVPD}(w)$ without and .*

Proof. By Lemma 5.15, it remains to show $r(w)$ is the maximal number of weighty tiles of an element in $\mathsf{MVPD}(w)$, which is $\mathsf{raj}(w)$. By Corollary 5.2, $\mathsf{raj}(w) = \mathsf{raj}(w^{-1})$. Since w^{-1} is fireworks, by Proposition 5.1, $\mathsf{raj}(w^{-1}) = \mathsf{maj}(w^{-1})$. It remains to check $r(w) = \mathsf{maj}(w^{-1})$.

Recall that $r(w) = \sum_{i \in I}(i-1)$, where $I = \{i : i^{\text{th}} \text{ entry of } \alpha'(w) \text{ is not } 0\}$. In other words, I consists of all i such that $w^{-1}(i)$ is not a left-to-right maximum of w^{-1}. Since w^{-1} is fireworks, I consists of i such that $w^{-1}(i)$ is not the first number in its decreasing run. Then we have

$$\begin{aligned}\{i-1 : i \in I\} &= \{j : w^{-1}(i) \text{ is not the last number in its decreasing run}\} \\ &= \{j : w^{-1}(j) > w^{-1}(j+1)\}.\end{aligned}$$

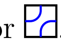
Thus, $r(w) = \sum_{i \in I}(i-1) = \sum_{j : w^{-1}(j) > w^{-1}(j+1)} j = \mathsf{maj}(w^{-1})$. $\square$

Corollary 5.17. *The first column of any $M \in \widehat{\mathsf{MVPD}(w)}$ only consists of and .*

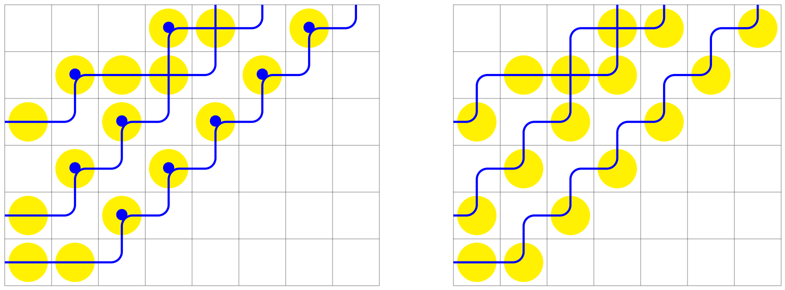
Proof. Suppose not. Say pipe p has a in column 1 of M. We know $w^{-1}(1)$ is a left-to-right maximum in w^{-1}, so the first entry in $\alpha'(w)$ is 0. In other words, pipe p must exit column 1. Find the cell in column 1 where pipe p exits enters from the bottom and exits from the right. By $M \in \widehat{\mathsf{MVPD}(w)}$ and Lemma 5.16, this cell can only be where the two pipe or . It cannot be a since pipe p has not crossed with the pipe entering from the left. It cannot be a since pipe p does not have a before. Contradiction. $\square$

Now it remains to establish a bijection from $\widehat{\mathsf{MVPD}(w)}$ to $\mathsf{BVPD}(w)$. We describe the map $\Phi^{M \to B}$ as follows. Take $M \in \widehat{\mathsf{MVPD}(w)}$, we remove its first column and change all into , obtaining a tiling B. The inverse of this map, denoted as $\Phi^{B \to M}$ is also straightforward: Add a column on the left of B consisting of and and change all in B into .

Proposition 5.18. *The maps $\Phi^{M \to B}$ and $\Phi^{B \to M}$ are bijections between $\widehat{\mathsf{MVPD}(w)}$ and $\mathsf{BVPD}(w)$ that preserve $\mathsf{wty}(\cdot)$*

Proof. Say $\Phi^{M\to B}$ sends $M \in \widehat{\mathsf{MVPD}(w)}$ to B. Since M has neither  nor , B is a BVPD. Recall that $\alpha(w)$ is obtained from $\alpha'(w)$ by removing the first 0. Since M has column-to-row code $\alpha'(w)$, we know B has column-to-row code $\alpha(w)$, so $B \in \mathsf{BVPD}(w)$. The two maps are clearly inverses of each other. To show the bijections preserve $\mathsf{wty}(\cdot)$, we present the following example. $\square$

Example 5.19. The left diagram is $M \in \widehat{\mathsf{MVPD}(w)}$ and the left diagram is $\Phi^{M\to B}(M) = B \in \mathsf{BVPD}(w)$. Their weighty tiles (highlighted yellow) agree.



Now we prove the main results of this section.

Proof of Theorem 5.13. By Corollary 5.10, $\widehat{\mathfrak{G}}_w(\mathbf{x}) = \sum_{M\in\widehat{\mathsf{MVPD}(w)}} \mathsf{wt}_M(\mathbf{x})$. Then by Proposition 5.18, $\sum_{M\in\widehat{\mathsf{MVPD}(w)}} \mathsf{wt}_M(\mathbf{x}) = \sum_{B\in\mathsf{BVPD}(w)} \mathsf{wt}_B(\mathbf{x})$. $\square$

Proof of Theorem 5.14. Take $B \in \mathsf{BVPD}(w)$. To obtain $P \in \widehat{\mathsf{PD}(w)}$, we simply apply $\Phi^{B\to M}$ to B, followed by the bijection from $\mathsf{MVPD}(w)$ to $\mathsf{PD}(w)$. Both maps preserve $\mathsf{wty}(\cdot)$, so $\mathsf{wty}(B) = \mathsf{wty}(P)$. $\square$

6. Conjecture 6.3 for inverse fireworks permutations

In this section, we prove Conjecture 6.3 for inverse fireworks permutations using MVPDs defined in Section 5. Our approach is constructive: For $M \in \mathsf{MVPD}(w)\backslash\widehat{\mathsf{MVPD}(w)}$, we construct M' such that $\mathsf{wt}_M(\mathbf{x})x_i = \mathsf{wt}_{M'}(\mathbf{x})$ for some i using "droop moves".

6.1. **Support of Grothendieck.** We start by introducing several conjectures by Mészáros, Setiabrata, and St. Dizier [MSSD22] on the support of Grothendieck polynomials.

Definition 6.1. For $f \in \mathbb{Z}[x_1, \dots, x_n]$ where $f = \sum_{\alpha \in \mathbb{Z}^n_{\geqslant 0}} c_\alpha x^\alpha$, the *support* of f is

$$\mathsf{Supp}(f) = \{\alpha \in \mathbb{Z}^n_{\geqslant 0} : c_\alpha \neq 0\}$$

It is the set of monomials of f with non-zero coefficients.

For $\alpha, \beta \in \mathbb{Z}^n_{\geqslant 0}$, we say $\alpha \leqslant \beta$ if $\alpha_i \leqslant \beta_i$ for all i. We say $\alpha < \beta$ if $\alpha \leqslant \beta$ and $\alpha_j < \beta_j$ for some j. Let $|\alpha|$ be the sum of all α_i. For a fix $w \in S_n$, we have the following three conjectures on $\mathsf{Supp}(\mathfrak{G}_w(\mathbf{x}))$.

Conjecture 6.2 ([MSSD22, Conjecture 1.1])**.** *If $\alpha \in \mathsf{Supp}(\mathfrak{G}_w)$ and $|\alpha| < \deg(\mathfrak{G}_w)$, then there exists $\beta \in \mathsf{Supp}(\mathfrak{G}_w)$ such that $\alpha < \beta$.*

Conjecture 6.3 ([MSSD22, Conjecture 1.2])**.** *If $\alpha \in \mathsf{Supp}(\mathfrak{G}_w)$ and $|\alpha| < \deg(\mathfrak{G}_w)$, then there exists $\beta \in \mathsf{Supp}(\mathfrak{G}_w)$ such that $\alpha < \beta$ and $|\alpha| + 1 = |\beta|$.*

Conjecture 6.4 ([MSSD22, Conjecture 1.3])**.** *If $\alpha, \gamma \in \mathsf{Supp}(\mathfrak{G}_w)$, then*

$$\{\beta : \alpha \leqslant \beta \leqslant \gamma\} \subseteq \mathsf{Supp}(\mathfrak{G}_w)$$

Each conjecture is strictly stronger than the previous one. Mészáros, Setiabrata, and St. Dizier proved these conjectures for Grassmannian permutations [MSSD22], which was later generalized to vexillary permutations by Hafner [Haf22]. Mészáros, Setiabrata, and St. Dizier also proved Conjecture 6.2 for fireworks permutations. We prove Conjecture 6.3 for inverse fireworks permutations. The same result was proved by Anna Weigandt separately using a different method.

6.2. **Properties of MVPD.** Let $w \in S_n$ be an arbitrary permutation in this section. We start with two observations on $\mathsf{MVPD}(w)$.

Lemma 6.5. *Take $M \in \mathsf{MVPD}(w)$. For every pipe, we can find a $\boxminus$ in M containing that pipe.*

Proof. Consider the pipe from row r of M. We know r appears in $\alpha'(w)$, so r is not a left-to-right maximum in w^{-1}. Say $m > r$ is a number on the left of r in w^{-1}. In the pipedream corresponding to M, there must be a $\boxplus$ where the pipe from row r goes from left to right and the pipe from row m goes from bottom to top. To obtain M from this pipedream, we remove the pipe from row m, so this tile becomes a $\boxminus$. □

We say a $\boxplus$ of M is a *real crossing* if its two pipes really cross in it (i.e. the pipe entering from the bottom exits from top). Otherwise, we say the $\boxplus$ is a *fake crossing*.

Lemma 6.6. *Take $M \in \mathsf{MVPD}(w)$. Say pipe p and pipe q have a real crossing in (i, j) and a fake crossing in (i', j'). We consider the region enclosed by the two pipes from (i, j) to (i', j'). For any pipe that appears in this region, it must cross with both pipe p and pipe q.*

Proof. For a pipe t to enter or exit this region, it must cross with pipe p or pipe q. Since two pipes cannot cross more than once, pipe t must cross both pipe p and pipe q. □

Next, we define the *droop moves* on MVPDs, which look similar to the droop moves on bumpless pipedreams introduced in [LLS21].

Definition 6.7. Take $M \in \mathsf{MVPD}(w)$. We define $\mathsf{droop}_{(i,j)}(M)$ if the following are all satisfied

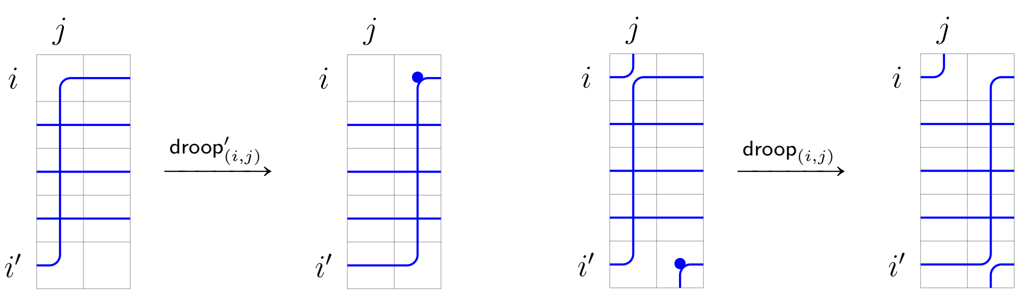
- The tile (i, j) contains a pipe entering from the bottom and exits from the right (i.e. It is a , , or a fake crossing).
- The tile $(i, j + 1)$ is a .
- Let $i' > i$ be the smallest such that (i', j) is not .Then (i', j) is a .

For each $i < r < i'$, we know (r, j) is a . A simple induction would imply that $(r, j + 1)$ has no pipe entering from the bottom. Thus, $(r, j + 1)$ is and $(i', j + 1)$ is , or . The operation $\mathsf{droop}(\cdot)$ does the following to column j and $j + 1$ between row i and row i'.

- Change (i, j) from or fake crossing to . Change (i, j) from or to .
- Change $(i, j + 1)$ from to .
- For $i < r < i'$,change (r, j) from to and change $(r, j + 1)$ from to .
- Change (i', j) from to .
- Change $(i', j + 1)$ from or to . Change $(i', j + 1)$ from to .

We also define $\mathsf{droop}'_{(i,j)}(M)$ on such (i, j) and M. It first performs $\mathsf{droop}_{(i,j)}$. Then notice that the pipe in $(i, j + 1)$ of $\mathsf{droop}_{(i,j)}(M)$ must have a in (r, j) for some $i < r \leqslant i'$. We may mark the pipe in $(i, j + 1)$, obtaining a valid MVPD $\mathsf{droop}'_{(i,j)}(M)$.

Example 6.8. We give two examples of the effect of $\mathsf{droop}'_{(i,j)}$ and $\mathsf{droop}_{(i,j)}$.



Lemma 6.9. *Take $M \in \mathsf{MVPD}(w)$. Then $\mathsf{droop}_{(i,j)}(M)$ and $\mathsf{droop}'_{(i,j)}(M)$ are both in $\mathsf{MVPD}(w)$ if they are defined.*

Proof. Let $i' > i$ be the smallest such that (i', j) is not . We just need to show that the same pipe exits from the right edge of $(r, j + 1)$ for $i \leqslant r < i'$ in M and $\mathsf{droop}_{(i,j)}(M)$. To prove this, we claim: For $i < r \leqslant i'$, if a pipe exits from the top of (r, j) in M, then the same pipe exits from the top of $(r, j + 1)$ in $\mathsf{droop}_{(i,j)}(M)$. We prove by induction on r. The base case when $r = i'$ is immediate. Now suppose $i < r < i'$. Say pipe p enters (r, j) from the left and pipe q enters (r, j) from the bottom in M. Then pipe $\max(p, q)$ exits from the the top of (r, j). The other pipe exits from the right of $(r, j + 1)$. By our inductive hypothesis, pipe p enters $(r, j + 1)$ from the left and pipe q enters $(r, j + 1)$ from the bottom in $\mathsf{droop}_{(i,j)}(M)$.

Pipe $\max(p,q)$ exits from the the top of $(r, j+1)$, and the other pipe exits from the right. Our inductive step is finished. □

Finally, we study a special family of MVPDs.

Definition 6.10. A $M \in \mathsf{MVPD}(w)$ is called *saturated* if it satisfies both of the following.

- For any in M, the pipe in it does not have before.
- For any , the two pipes in it do not cross in M.

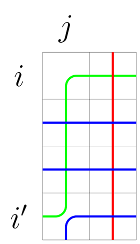
In other words, an $M \in \mathsf{MVPD}(w)$ is not saturated if we can turn one of its to or to and still remain in $\mathsf{MVPD}(w)$.

Lemma 6.11. *Take a saturated $M \in \mathsf{MVPD}(w)$. Say a pipe p enters the tile (i,j) from the bottom and exits from the right. Then $(i, j+1)$ cannot be a real crossing.*

Proof. Suppose there exists such (i,j). We pick one such (i,j) where i is maximal. Say pipe p enters from the bottom of (i,j) and say it crosses with pipe q in $(i+1, j)$. We know these two pipes have not crossed before $(i, j+1)$. Moreover, since M is saturated, there is no in M involving pipe p and pipe q. As a conclusion, under row i, there is no tile containing both pipe p and pipe q.

Find $i' > i$ such that pipe p enters on the left edge of (i', j). We know pipe p goes from bottom to top of (r, j) for $i < r < i'$. Say pipe q enters on the left edge of $(i'', j+1)$. We have $i'' > i$ since otherwise, (i'', j) would be a tile containing both pipe p and pipe q. Pipe q goes from bottom to top of $(i'', j+1)$, so this tile is a real crossing. Consider the tile (i', j). It contains pipe p which enters on the left and exits on the top. Thus, it also must contains a pipe entering from the bottom and exits on the right. We reach a contradiction since we picked the maximal i. □

Here is an illustration of the proof of Lemma 6.11. We make pipe p green and pipe q red.



6.3. **Construction.** Fix inverse fireworks w in throughout this section. For each $M \in \mathsf{MVPD}(w) \backslash \widehat{\mathsf{MVPD}(w)}$, our goal is to construct M' such that $\mathsf{wt}_{M'}(\mathbf{x}) = \mathsf{wt}_M(\mathbf{x}) x_i$ for some i. If M is not saturated, we can find the M' easily: Say M has an and the pipe in it has a before, we simply mark the and obtain M'. Otherwise, say M has a where the two pipes in it cross somewhere else. We may turn this into and the resulting MVPD is still in $\mathsf{MVPD}(w)$. It remains to consider saturated M. Our construction relies on the operator $\mathsf{droop}'_{i,j}(\cdot)$, which requires us to find an occurrence of or in M. That is, a or with a immediately on its right.

Lemma 6.12. *Take a saturated $M \in \mathsf{MVPD}(w) \backslash \widehat{\mathsf{MVPD}(w)}$. In M, there exists or .*

Proof. Since $M \notin \widehat{\mathsf{MVPD}}(w)$, by Lemma 5.16, M must have a or . Let (i, j) be the highest, or one of the highest, such tile. We prove $(i, j + 1)$ must be by contradiction. Suppose $(i, j + 1)$ is not a . Let pipe p be the pipe that enters (i, j) from the bottom and exits on the right. By Lemma 6.11, $(i, j+1)$ cannot be a real crossing. Thus, $(i, j+1)$ can be a fake crossing, a , or a . In any case, pipe p must exits on the top of $(i, j+1)$. Then we present two different arguments based on whether (i, j) is or . Both arguments follow the following three steps:

- Step 1: Show pipe p must exits column $j + 1$. Say it exits from the right edge of $(i', j + 1)$ for some $i' < i$.
- Step 2: We know $(i', j + 1)$ cannot be or by how we picked (i, j). We show $(i', j + 1)$ cannot be an , so it must be a fake crossing.
- Step 3: Find a contradiction.

We start with the case where (i, j) is a bump. Let pipe q be the pipe exiting from the top of (i, j). Since M is saturated, we know pipe p and pipe q never cross in M, so $q < p$. Now we perform the three steps and eventually show pipe p and pipe q must cross, which would be a contradiction.

- Step 1: Suppose pipe p does not exit column $j + 1$. Since pipe p and q cannot cross, pipe q does not exit column j. In other words, $\alpha(j + 1) = p$ and $\alpha(j) = q$. Then $w^{-1}(j + 2) = p$ and $w^{-1}(j + 1) = q$. Since $q < p$, p is actually the first number in its decreasing run in w^{-1}, so p cannot appear in $\alpha(w)$. We reach a contradiction, so pipe p must exit column $j + 1$.
- Step 2: We know pipe p goes from the bottom to top in $(r, j + 1)$ for $i' < r < i$. Since pipe q cannot cross with pipe p, it must also go from bottom to top in (r, j) for $i' < r < i$. Thus, pipe q enters (i', j) from the bottom. The tile (i', j) must have some pipe exiting from the right. Thus, $(i', j + 1)$ has a pipe entering from the left, so it cannot be . It must be a fake crossing.
- Step 3: Let pipe t be the pipe that enters $(i', j + 1)$ from the left. Since $(i', j + 1)$ is a fake crossing, pipe t and pipe p must have a real crossing under row i. Then pipe t must exits row i on the left of pipe p. Since pipe p exits row i on column $j + 1$ and pipe q exits row i on column j, we know pipe t exits row i on the left of column j. Now consider the region enclosed by pipe p and t from their real crossing to $(i', j + 1)$. Pipe q appears in this region. By Lemma 6.6, pipe q crosses with pipe p. Contradiction.

Now assume (i, j) is . By M is saturated, we know pipe p does not have a before (i, j). We perform the three steps.

- Step 1: If pipe p does not exits column $j + 1$, then it does not have a in M, contradicting Lemma 6.5.
- Step 2: In (i', j), the pipe p still does not have a yet, so (i', j) cannot be . It must be a fake crossing.
- Step 3: Say $(i', j+1)$ is a fake crossing between pipe p and pipe t. Pipe p must have a real crossing under row i where pipe p goes horizontally. In other words, we can find a real crossing (r_+, c_+) where pipe p goes horizontally with $r_+ > i$. Take the (r_+, c_+) where c_+ is maximal. Thus, from (r_+, c_+) to $(i, j + 1)$, the pipe p is not allowed to

travel horizontally in any tile. In other words, if pipe p enters a tile from the left, it must exit from the top.

Next, we argue for $i \leqslant r \leqslant r_+$, when pipe p exits row r, there is a pipe exiting from the cell on its left, which has already crossed with pipe p.

We prove our claim by induction. The base case is when $r = r_+$. We know (r_+, c_+) is a real crossing. Since pipe p enters $(r_+, c_+ + 1)$ from the left, it exits row r_+ from $(r_+, c_+ + 1)$. Indeed, (r_+, c_+) has a pipe exiting from the top, which just crossed with pipe p. Now take $i \leqslant r < r_+$. Say pipe p enters from the bottom of (r, c). By our inductive hypothesis, another pipe enters $(r, c-1)$ from the bottom. Say it is pipe s If pipe p goes vertically in (r, c), pipe s must go vertically in $(r, c-1)$ since if it exits on the right, (r, c) would be a fake crossing. Now suppose pipe p exits (r, c) on the right. Since $(r, c-1)$ has a pipe entering from the bottom, it must has a pipe exiting from the right. Then (r, c) can be a fake crossing or a . Consider the region enclosed by pipe t and pipe p from their real crossing to (i', j). The other pipe in (r, c) is either pipe t, or lies in this region. In either case, it must cross with pipe p. Since M is saturated, (r, c) is not a , so it is a fake crossing. Pipe p exits from the top of $(r, c+1)$ and some pipe that has crossed with it exits from the top of (r, c).

Finally, our claim implies when pipe p exits $(i, j+1)$ from the top, there must be a pipe that exits (i, j) from the top. This contradicts to our assumption that (i, j) is . □

Now we describe our algorithm. Take a saturated $M \in \mathsf{MVPD}(w)\backslash\widehat{\mathsf{MVPD}(w)}$. By Lemma 6.12, we know M must have a or . We let (i, j) and $(i, j+1)$ be the lowest such occurrence where we first maximize i, and then j. We check $\mathsf{droop}'_{(i,j)}(M)$ is defined. The first two conditions in Definition 6.7 are immediate. For the last condition, we let $i' > i$ be the smallest such that (i', j) is not . It can be or . Assume it is a toward contradiction. Since $(i'-1, j+1)$ is , we know $(i', j+1)$ is also a . This contradicts the maximality of i. Thus, (i', j) is a and $\mathsf{droop}_{(i,j)}(M)$ is well-defined.

Next, the algorithm computes $\mathsf{droop}'_{(i,j)}(M)$, which is in $\mathsf{MVPD}(w)$ by Lemma 6.9. We compare the weighty tiles of M and $\mathsf{droop}'_{(i,j)}(M)$:

- The tile (i, j) is not weighty in M and $\mathsf{droop}'_{i,j}(M)$. The tile $(i, j+1)$ is weighty in M and $\mathsf{droop}'_{i,j}(M)$.
- For $i < r < i'$, the tile (r, j) and $(r, j+1)$ are weighty in M and $\mathsf{droop}'_{i,j}(M)$.
- The tile (i', j) is not weighty in M but becomes weighty in $\mathsf{droop}'_{i,j}(M)$. The tile $(i', j+1)$ could be either weighty or not in M, but is not weighty in $\mathsf{droop}'_{i,j}(M)$.

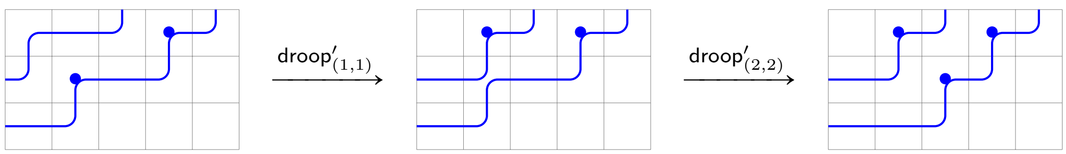
If $\mathsf{droop}'_{i,j}(M)$ has one more weighty tile than M, we let $M' = \mathsf{droop}'_{i,j}(M)$ and terminate. Then $\mathsf{wt}_{M'}(\mathbf{x}) = \mathsf{wt}_M(\mathbf{x}) x_{i'}$. Otherwise,

$$\mathsf{wty}(\mathsf{droop}'_{i,j}(M)) = (\mathsf{wty}(M)\backslash\{(i', j+1\}) \cup \{(i', j)\},$$

so $\mathsf{droop}'_{i,j}(M)$ and M have the same number of weighty tiles. If $\mathsf{droop}'_{i,j}(M)$ is not saturated, then we change a or a into a weighty tile and obtain M'. Otherwise, we update the variable M into $\mathsf{droop}'_{(i,j)}(M)$ and repeat the algorithm.

It remains to show the algorithm eventually terminates. Let $M_1, M_2, \cdots$ be the MVPDs in the start of each iteration. We know $\mathsf{wty}(M_k)$ is obtained from $\mathsf{wty}(M_{k-1})$ by turning an (r, c) into $(r, c-1)$. Thus, the algorithm must terminate.

Example 6.13. The following is an example of the algorithm. We start with a saturated $M \in \mathsf{MVPD}(w) \backslash \widehat{\mathsf{MVPD}(w)}$ where w^{-1} has one-line notation 14253. We first apply $\mathsf{droop}'_{(1,1)}$ and obtain another $M_2 \in \mathsf{MVPD}(w)$. Notice that M_2 is also saturated and $\mathsf{wt}_M(\mathbf{x}) = \mathsf{wt}_{M_2}(\mathbf{x})$. We then apply $\mathsf{droop}'_{(2,2)}$ and obtain M'. Notice that $\mathsf{wt}_{M'}(\mathbf{x}) = \mathsf{wt}_M(\mathbf{x})x_3$.



7. Conclusion

In this thesis, we introduced important polynomials in Schubert calculus, including Schubert and Grothendieck polynomials. We studied their support through combinatorial objects called pipedreams. We now provide some conjectures on the support of Grothendieck polynomials and future directions for related topics.

7.1. Conjectures.

In section 6, we introduced three conjectures on the support of Grothendieck polynomials by Mészáros, Setiabrata, and St. Dizier [MSSD22]. We now introduce other conjectures on Grothendieck polynomials and their supports.

It is conjectured by Huh, Matherne, Mészáros, and St. Dizier [HMMSD22] that homogenized Grothendieck polynomials are Lorentzian. In particular, this would imply that the support of homogenized Grothendieck polynomials are M-convex. Or equivalently, the support forms a saturated Newton polytope. It is already proven for several families of special permutations that their homogenized Grothendieck polynomials are M-convex [EY17, MSD20, HMSSD23, CCRMM24]. However, the general case remains open. M-convexity of homogenized Grothendieck polynomials would imply all three conjectures on the support of Grothendieck mentioned in section 6.

Ross and Yong [RY13] conjectured a K-Kohnert rule for Grothendieck polynomials on Rothe diagrams of permutations. Robichaux [Rob24] showed that this conjecture fails by providing a counter example and provided an updated version of the conjecture.

7.2. Future directions.

Schubert and Grothendieck polynomials can be extended to a "quantum" direction. Brenti, Fomin, and Postnikov [BFP99] defined quantum bruhat graphs to study the 3-point Gromov-Witten invariants of the flag variety, which are the structure constants of the small quantum cohomology ring. Quantum bruhat graphs are also used to study path Schubert polynomials [Pos05] and tilted Richardson varities [GGG23]. Quantum double Schubert polynomials are generalizations of double Schubert polynomials. They are polynomial representatives of Schubert classes in the torus-equivariant quantum cohomology of the complete flag variety. Le, Ouyang, Tao, Restivo, and Zhang [LOT+24] generalized bumpless pipedreams to give a combinatorial formula for quantum double Schubert polynomials. They are also interested in a potential quantum version of pipedreams as well as a quantum version of the Gao-Huang bijection [GH23]. A quantum version of non-reduced bumpless pipedream (or pipedreams) that computes quantum Grothendieck polynomials is also yet to be discovered.

Lascoux polynomials, defined by Alain Lascoux [Las04], are polynomials labeled by compositions. They form a basis for the ring of polynomials. Shimozono and Yu [SY23] gave an expansion of Grothendieck polynomials into Lascoux polynomials using increasing tableaux. It is believed that conjecture 6.2 is true for the support of Lascoux polynomials. Therefore, one might want to studying the Lascoux polynomials that appear in a Grothendieck polynomial's expansion.

References

[BB93] Nantel Bergeron and Sara Billey. RC-graphs and Schubert polynomials. *Experimental Mathematics*, 2(4):257–269, 1993.

[BFP99] Francesco Brenti, Sergey Fomin, and Alexander Postnikov. Mixed bruhat operators and yang-baxter equations for weyl groups. *International Mathematics Research Notices*, 8:419–441, 1999.

[BJS93] Sara C Billey, William Jockusch, and Richard P Stanley. Some combinatorial properties of Schubert polynomials. *Journal of Algebraic Combinatorics*, 2(4):345–374, 1993.

[CCRMM24] Federico Castillo, Yairon Cid-Ruiz, Fatemeh Mohammadi, and Jonathan Montaño. K-polynomials of multiplicity-free varieties. *arXiv preprint arXiv:2212.13091*, 2024.

[CY23] Chen-An Chou and Tianyi Yu. Constructing maximal pipedreams of double Grothendieck polynomials. *arXiv preprint arXiv:2312.01250*, 2023.

[CY24] Chen-An Chou and Tianyi Yu. Grothendieck polynomials for inverse fireworks permutations. *arXiv preprint arXiv:2403.12186*, 2024.

[DMSD22] Matt Dreyer, Karola Mészáros, and Avery St. Dizier. On the degree of Grothendieck polynomials. *arXiv preprint arXiv:2209.00687*, 2022.

[EY17] Laura Escobar and Alexander Yong. Newton polytopes and symmetric grothendieck polynomials. *Comptes Rendus Mathematique*, 355(8):831–834, 2017.

[FK94] Sergey Fomin and Anatol N Kirillov. Grothendieck polynomials and the Yang-Baxter equation. In *Proc. formal power series and alg. comb*, pages 183–190, 1994.

[FUL92] W FULTON. Flags, Schubert polynomials, degeneracy loci, and determinantal formulas. *Duke Math. J.*, 65:381–420, 1992.

[GGG23] Jiyang Gao, Shiliang Gao, and Yibo Gao. Quantum bruhat graphs and tilted richardson varieties. *arXiv preprint arXiv:2309.01309*, 2023.

[GH23] Yibo Gao and Daoji Huang. The canonical bijection between pipe dreams and bumpless pipe dreams. *International Mathematics Research Notices*, 2023:18629–18663, 2023.

[Haf22] Elena S Hafner. Vexillary Grothendieck polynomials via bumpless pipe dreams. *arXiv preprint arXiv:2201.12432*, 2022.

[HMMSD22] June Huh, Jacob Matherne, Karola Mészáros, and Avery St. Dizier. Logarithmic concavity of Schur and related polynomials. *Transactions of the American Mathematical Society*, 375(6):4411–4427, 2022.

[HMSSD23] Elena S Hafner, Karola Mészáros, Linus Setiabrata, and Avery St. Dizier. M-convexity of Grothendieck polynomials via bubbling. *arXiv preprint arXiv:2306.08597*, 2023.

[KM05] Allen Knutson and Ezra Miller. Gröbner geometry of Schubert polynomials. *Annals of Mathematics*, pages 1245–1318, 2005.

[KMY09] Allen Knutson, Ezra Miller, and Alexander Yong. Gröbner geometry of vertex decompositions and of flagged tableaux. 2009.

[Las04] Alain Lascoux. Schubert grothendieck: Un bilan bidécennal. *Séminaire Lotharingien de Combinatoire*, 50:1–32, 2004.

[LLS21] Thomas Lam, Seung Jin Lee, and Mark Shimozono. Back stable Schubert calculus. *Compositio Mathematica*, 157(5):883–962, 2021.

[LLS23] Thomas Lam, Seung Jin Lee, and Mark Shimozono. Back stable K-theory Schubert calculus. *International Mathematics Research Notices*, 2023(24):21381–21466, 2023.

[LOT$^+$24] Tuong Le, Shuge Ouyang, Leo Tao, Joseph Restivo, and Angelina Zhang. Quantum bumpless pipe dreams. *arXiv preprint arXiv:2403.16168*, 2024.

[LRS06] Cristian Lenart, Shawn Robinson, and Frank Sottile. Grothendieck polynomials via permutation patterns and chains in the Bruhat order. *American Journal of Mathematics*, 128(4):805–848, 2006.

[LS82] Alain Lascoux and Marcel-Paul Schützenberger. Structure de Hopf de l'anneau de cohomologie et de l'anneau de Grothendieck d'une variété de drapeaux. *C. R. Acad. Sci. Paris Sér. I Math.*, 295(11):629–633, 1982.

[MSD20] Karola Mészáros and Avery St. Dizier. From generalized permutahedra to grothendieck polynomials via flow polytopes. *Algebraic Combinatorics*, 3(5):1197–1229, 2020.

[MSSD22] Karola Mészáros, Linus Setiabrata, and Avery St. Dizier. On the support of Grothendieck polynomials. *arXiv preprint arXiv:2201.09452*, 2022.

[Pos05] Alexander Postnikov. Quantum bruhat graph and schubert polynomials. *Proceedings of the American Mathematical Society*, 133(3):699–709, 2005.

[PSW21] Oliver Pechenik, David E Speyer, and Anna Weigandt. Castelnuovo-Mumford regularity of matrix Schubert varieties. *arXiv preprint arXiv:2111.10681*, 2021.

[PY23] Jianping Pan and Tianyi Yu. Top-degree components of Grothendieck and Lascoux polynomials. *arXiv preprint arXiv:2302.03643*, 2023.

[Ram85] Annamalai Ramanathan. Schubert varieties are arithmetically Cohen-Macaulay. *Inventiones mathematicae*, 80(2):283–294, 1985.

[Rob24] Colleen Robichaux. A counterexample to the ross–yong conjecture for grothendieck polynomials. *arXiv preprint arXiv:2403.14538*, 2024.

[RRR$^+$21] Jenna Rajchgot, Yi Ren, Colleen Robichaux, Avery St. Dizier, and Anna Weigandt. Degrees of symmetric Grothendieck polynomials and Castelnuovo-Mumford regularity. *Proceedings of the American Mathematical Society*, 149(4):1405–1416, 2021.

[RRW23] Jenna Rajchgot, Colleen Robichaux, and Anna Weigandt. Castelnuovo-Mumford regularity of ladder determinantal varieties and patches of Grassmannian Schubert varieties. *J. Algebra*, 617:160–191, 2023.

[RY13] Colleen Ross and Alexander Yong. Combinatorial rules for three bases of polynomials. *arXiv preprint arXiv:1302.0214*, 2013.

[SY23] Mark Shimozono and Tianyi Yu. Grothendieck-to-lascoux expansions. *Transactions of the American Mathematical Society*, 2023.

[WY18] Anna Weigandt and Alexander Yong. The prism tableau model for Schubert polynomials. *Journal of Combinatorial Theory, Series A*, 154:551–582, 2018.