

THE BOUNDED RC FACTOR ALGEBRA

MATTHEW J. SAMUEL

ABSTRACT. We present a self-contained account of the bounded RC factor algebra, an associative algebra whose basis elements are ordered tensors of full-rank Grassmannian RC graphs. The construction rests on three pillars: the ring of bounded RC graphs, equipped with a product defined via clipping and trimming; the Demazure crystal structure on RC graphs together with a family of crystal divided difference operators; and the squash product, which realizes the tensor product of crystals on the level of pipe dreams. After developing these ingredients, we define the algebra by means of confluent rewriting relations on tensors of Grassmannian factors and describe how to expand Schubert polynomials in this basis via the crystal elementary monomial representation.

1. INTRODUCTION

Schubert polynomials, introduced by Lascoux and Schützenberger, represent cohomology classes of Schubert varieties in the flag manifold and form a distinguished basis of the polynomial ring. Their structure constants, the Littlewood–Richardson coefficients for Schubert calculus, remain the subject of intense combinatorial investigation. RC graphs (also known as pipe dreams or reduced compatible sequences) provide a natural combinatorial indexing of Schubert monomials, and the additional structure one can impose on them has been a source of considerable progress.

The goal of this paper is to construct an associative algebra, here called the bounded RC factor algebra, whose elements are formal linear combinations of ordered tensors of full-rank Grassmannian RC graphs. We write a typical basis element as $G_1 \otimes G_2 \otimes \cdots \otimes G_k$ where each G_i is a Grassmannian pipe dream whose unique descent is equal to the number of rows in G_i . The algebra structure is defined by concatenation of tensors followed by a confluent normalization procedure. The resulting algebra maps surjectively onto the ring of bounded RC graphs via the squash product, and the Schubert element $\mathfrak{S}_w$ can be expressed as a finite linear combination of these tensor basis elements through what we call the crystal elementary monomial (CEM) representation.

We proceed as follows. In Section 2 we recall the definition of RC graphs and bounded RC graphs, and we define the ring structure on the free abelian group they span. Section 3 develops the Demazure crystal structure and the crystal divided difference operators. Section 4 treats the zeroing transition algorithm, which is the key technical ingredient for the ring product. Section 5 introduces the squash product and the squash decomposition of an RC graph into an S_n -factor and a Grassmannian factor. Section 6 defines the bounded RC factor algebra and its normalization relations. Finally, Section 7 describes the CEM representation of Schubert polynomials in this algebra.

2. BOUNDED RC GRAPHS AND THEIR RING STRUCTURE

2.1. RC graphs. Let $\mathbb{P} = \{1, 2, 3, \dots\}$ denote the positive integers and let S_∞ denote the group of permutations of $\mathbb{P}$ that fix all but finitely many elements, generated by the adjacent transpositions $s_i = (i, i+1)$ for $i \geq 1$.

Definition 2.1. Let $R \subseteq \mathbb{P} \times \mathbb{P}$ be a finite set. To each such set we associate a permutation $w_R \in S_\infty$ as follows. Given a pair (i, j) with $i, j > 0$, define $s(i, j) = s_{i+j-1}$. Totally order $\mathbb{P} \times \mathbb{P}$ by declaring $(i, j) < (a, b)$ if and only if $i < a$, or $i = a$ and $j > b$. Index the elements of R as $r_1, r_2, \dots, r_m$ in this order and set

$$w_R = s(r_1) \cdots s(r_m).$$

If $\ell(w_R) = m$, then R is called an *RC graph* (or a *pipe dream*).

Informally, an RC graph is a collection of crossing tiles placed in positions (i, j) of a grid, read from top to bottom and right to left within each row, such that the resulting product of transpositions is a reduced word for the associated permutation.

Definition 2.2. A *bounded RC graph* is an RC graph R together with a positive integer $n \geq \max\{i : (i, j) \in R \text{ for some } j\}$ that satisfies $\mathbf{m}(w_R) \leq n$, where $\mathbf{m}(w)$ denotes the largest descent of w . We write $\mathbf{ht}(R) = n$ for the *height* of the bounded RC graph and $\mathbf{wt}(R)$ for its *weight vector*, where $\mathbf{wt}_i(R)$ counts the number of elements in row i .

The *trim* operation removes the top row: $\mathbf{trim}(R)$ is the bounded RC graph of height $\mathbf{ht}(R) - 1$ with underlying set $\{(i - 1, j - 1) : (i, j) \in R, i \geq 2\}$. The *shift* operation acts by $\uparrow^m R = \{(i + m, j) : (i, j) \in R\}$.

Let $\mathcal{BRC}$ denote the free abelian group spanned by all bounded RC graphs. It admits a grading $\mathcal{BRC} = \bigoplus_{n \geq 0} \mathcal{BRC}_n$ where $\mathcal{BRC}_n$ is spanned by those R with $\mathbf{ht}(R) = n$.

2.2. The clipping and trimming operations. The ring product on $\mathcal{BRC}$ requires the ability to break a bounded RC graph into a “top part” and a “bottom part” and to reassemble it. This is accomplished by two operations. Given a bounded RC graph R with $\mathbf{ht}(R) = n$ and an integer $0 \leq p \leq n$, we define:

Definition 2.3. The *clip* of R to height p , denoted $\mathbf{clip}^p(R)$, is the bounded RC graph of height p obtained by taking the restriction of R to its first p rows and applying the zeroing transition (described in Section 4) to remove all descents larger than p .

The *trim* to depth p , denoted $\mathbf{trim}^p(R)$, is the bounded RC graph of height $n - p$ given by $\mathbf{trim}^p(R) = \{(i - p, j) : (i, j) \in R, i > p\}$.

These two operations satisfy the following associativity properties, which are crucial for the ring product.

Lemma 2.4. For a bounded RC graph R and integers $0 \leq p \leq p + q \leq \mathbf{ht}(R)$,

$$(2.1) \quad \mathbf{clip}^p(\mathbf{clip}^{p+q}(R)) = \mathbf{clip}^p(R),$$

$$(2.2) \quad \mathbf{trim}^p(\mathbf{clip}^{p+q}(R)) = \mathbf{clip}^q(\mathbf{trim}^p(R)),$$

$$(2.3) \quad \mathbf{trim}^{p+q}(R) = \mathbf{trim}^q(\mathbf{trim}^p(R)).$$

The key structural result is that the pair $(\mathbf{clip}^p(R), \mathbf{trim}^p(R))$ together with w_R uniquely determines R .

Theorem 2.1. For each integer $0 \leq p \leq n$, the map $\mathcal{BRC}_n \rightarrow \mathcal{BRC}_p \times \mathcal{BRC}_{n-p} \times S_\infty$ defined by $R \mapsto (\mathbf{clip}^p(R), \mathbf{trim}^p(R), w_R)$ is injective.

2.3. The ring product.

Definition 2.5. The *diamond product* of two bounded RC graphs R_1 with $\mathbf{ht}(R_1) = m$ and R_2 with $\mathbf{ht}(R_2) = n$ is

$$R_1 \diamond R_2 = \sum_{\substack{R' \in \mathcal{BRC}_{m+n} \\ \mathbf{clip}^m(R') = R_1 \\ \mathbf{trim}^n(R') = R_2}} R'.$$

Theorem 2.2. The product $\diamond$ turns $\mathcal{BRC}$ into an associative ring.

Proof. Associativity follows from the identities in Lemma 2.4. If R_w is any term in the expansion of $(R_1 \diamond R_2) \diamond R_3$ with $\mathbf{ht}(R_w) = p + q + r$, then equation (2.1) gives $\mathbf{clip}^p(\mathbf{clip}^{p+q}(R_w)) = \mathbf{clip}^p(R_w) = R_1$, equation (2.2) gives $\mathbf{trim}^p(\mathbf{clip}^{p+q}(R_w)) = \mathbf{clip}^q(\mathbf{trim}^p(R_w)) = R_2$, and equation (2.3) gives $\mathbf{trim}^{p+q}(R_w) = R_3$. Each term appears with multiplicity one, so the two bracketings produce identical sums. $\square$

The ring $\mathcal{BRC}$ admits a natural map to the divided difference algebra that is compatible with this product, but we will not need this here.

3. THE DEMAZURE CRYSTAL STRUCTURE

3.1. Crystal operators on RC graphs. Assaf and Schilling defined a Demazure crystal structure on the set $\mathcal{RC}(w)$ of all RC graphs for a fixed permutation w . For each $i \geq 1$, lowering operators $f_i : \mathcal{RC}(w) \rightarrow \mathcal{RC}(w) \cup \{0\}$ and raising operators $e_i : \mathcal{RC}(w) \rightarrow \mathcal{RC}(w) \cup \{0\}$ are defined via a pairing algorithm between rows i and $i + 1$.

Given an RC graph R , one pairs elements of row i with elements of row $i + 1$ greedily from right to left: the element (i, j) is paired with the element $(i + 1, k)$ where k is the smallest value such that $k \geq j$ and $(i + 1, k)$ is not yet paired. Unpaired elements in row i form the set R_i ; unpaired elements in row $i + 1$

form the set L_i . The lowering operator f_i removes the leftmost element of R_i and inserts an element in the corresponding position in row $i + 1$, while e_i removes the rightmost element of L_i and inserts one in row i . If the relevant set is empty, the operator returns 0.

These operators satisfy the Stembridge axioms for a Demazure crystal, and the connected components under crystal operators decompose $\mathcal{RC}(w)$ into Demazure crystal components. Each component has a unique *highest weight* element R^{hw} (characterized by $e_i(R^{\text{hw}}) = 0$ for all i) and a unique *lowest weight* element (characterized by $f_i = 0$ for all i).

3.2. Crystal divided differences.

Definition 3.1. Let R be an RC graph and let $i > 0$. Define $\mathfrak{F}^i R$ as follows. Set $\mathfrak{F}^i R = 0$ unless s_i is a right descent of w_R and there exists $(i, j) \in R$ with $\mathbf{rt}_R(i, j) = (i, i + 1)$. When these conditions hold, let $R' = R \setminus \{(i, j)\}$. If $e_i(R') \neq 0$, set $\mathfrak{F}^i R = 0$. Otherwise,

$$\mathfrak{F}^i R = \sum_{p=0}^{\varphi_i(R')} f_i^p R',$$

where $\varphi_i(R')$ is the string length in the i -direction of the crystal.

The crystal divided difference operators satisfy the nilCoxeter relations: $(\mathfrak{F}^i)^2 = 0$, $\mathfrak{F}^i \mathfrak{F}^j = \mathfrak{F}^j \mathfrak{F}^i$ for $|i - j| > 1$, and $\mathfrak{F}^i \mathfrak{F}^{i+1} \mathfrak{F}^i = \mathfrak{F}^{i+1} \mathfrak{F}^i \mathfrak{F}^{i+1}$.

Theorem 3.1. If i is a right descent of w , then $\mathfrak{F}^i \mathcal{S}_w(n) = \mathcal{S}_{ws_i}(n)$, where $\mathcal{S}_w(n) = \sum_{w_R=w, \text{ht}(R)=n} R$ is the Schubert element. If i is not a right descent, then $\mathfrak{F}^i \mathcal{S}_w(n) = 0$.

An important consequence is the following.

Proposition 3.2. Let $u \in S_n$ and suppose $(u_1, \dots, u_m)$ is a reduced word for $u^{-1}w_0$. Then there is a bijection between the nonzero terms

$$f_{u_1}^{p_1} \mathcal{X}^{u_1} \dots f_{u_m}^{p_m} \mathcal{X}^{u_m} R^0(w_0)$$

where $\mathcal{X}^i$ is the exchange-deletion operator and $p_i \geq 0$ are integers and the set $\mathcal{RC}(u)$.

4. THE ZEROING TRANSITION

The zeroing transition algorithm is the mechanism by which one removes an empty row from the bottom of a bounded RC graph, adjusting the crossings so that the resulting graph is a valid bounded RC graph of smaller height. It underlies both the clipping operation and the diamond product.

Definition 4.1. Let R be a bounded RC graph with $\text{ht}(R) = n$ and the last row of R empty. The *zeroing map* $\mathcal{Z}(R)$ produces a bounded RC graph R' with $\text{ht}(R') = n - 1$, $|R'| = |R|$, and $w_R \xrightarrow{n} w_{R'}$ (that is, $w_{R'}$ lies in the image of w_R under the parabolic projection at position n). The procedure is described in Algorithm 1.

Algorithm 1 The zeroing map $\mathcal{Z}(R)$

- 1: Let $n = \text{ht}(R)$ and suppose row n of R is empty.
 - 2: **if** removing row n from R yields a valid bounded RC graph **then**
 - 3: **return** R with row n removed.
 - 4: **else**
 - 5: Set $R_0 \leftarrow R$ and find the maximal sequence $(i_1, j_1), \dots, (i_p, j_p)$ in R such that $\mathbf{rt}_{R_{m-1}}(i_m, j_m) = (n + m - 1, n + m)$, where $R_m = R_{m-1} \setminus \{(i_m, j_m)\}$.
 - 6: Set $R^- = R \setminus \{(i_1, j_1), \dots, (i_p, j_p)\}$ and $I = (i_1, \dots, i_p)$.
 - 7: Form $D = R^- \cup \{(n, 1), (n, 2), \dots, (n, p)\}$.
 - 8: Apply pseudo-Pieri insertion: $D' = I \xrightarrow{n-1} D$.
 - 9: **return** $R' = \{(i, j) \in D' : i < n\}$ with $\text{ht}(R') = n - 1$.
 - 10: **end if**
-

The zeroing map is a weight-preserving bijection from the set of n -row bounded RC graphs R with empty row n to bounded RC graphs of height $n-1$ with appropriately related permutations. The clipping operation $\text{clip}^p(R)$ is defined by restricting R to its first p rows and then iteratively applying $\mathcal{Z}$ until all descents above p have been removed.

The correctness of this algorithm relies on a reconstruction lemma: given the “stripped” graph $\bar{R}$ and the row sequence $\text{strip}(R)$ obtained by removing crossings that carry descents at or above n , one recovers R by pseudo-Pieri insertion. The zeroing map also coincides, under the Edelman–Greene correspondence, with the Little bump operation, which preserves the recording tableau and hence the crystal structure.

5. THE SQUASH PRODUCT AND SQUASH DECOMPOSITION

5.1. Grassmannian RC graphs. A permutation w is called *n -Grassmannian* if its only descent is at position n . The corresponding Schubert polynomial $\mathfrak{S}_w(x_1, \dots, x_n)$ is a Schur polynomial in n variables. An RC graph for such a permutation, when viewed as a bounded RC graph of height n , is called a Grassmannian RC graph.

We say that a Grassmannian RC graph is *full rank* (or *full*) if its unique descent equals n , the number of rows.

Definition 5.1. For each n , let $\mathcal{G}_n$ denote the free abelian group generated by the RC graphs G with $\text{ht}(G) = n$ such that w_G is n -Grassmannian. Under the squash product (defined below), $\mathcal{G}_n$ is a ring isomorphic to the ring of tableaux under reverse Knuth insertion.

5.2. The squash product. Given two RC graphs R_1 and R_2 , their *disjoint union* $R_1 \star R_2$ is formed by shifting the columns of R_2 far enough to the right that no column overlaps with R_1 , then taking the union. Formally, choosing N larger than $i+j$ for all $(i, j) \in R_2$, one sets

$$R_1 \star R_2 = R_1 \cup \{(i, j + N) : (i, j) \in R_2\}.$$

Definition 5.2. The *squash product* of two bounded RC graphs R_1 and R_2 with $\text{ht}(R_1) = \text{ht}(R_2) = n$ is

$$R_1 \boxtimes R_2 = \text{clip}^n(R_1 \star R_2).$$

The squash product is associative and preserves the crystal structure in the following sense.

Theorem 5.1. Suppose (R_1, n) and (R_2, n) are bounded RC graphs. Then we have that there is a homomorphism of crystal graphs

$$\mathcal{C}(u, n) \otimes \mathcal{C}(v, n) \rightarrow \mathcal{C}(u) \star_n \mathcal{C}(v)$$

given by $R_1 \otimes R_2 \mapsto R_1 \star R_2$. If R_2 is n -Grassmannian, then this is a weight-preserving isomorphism of crystal graphs.

Proof. Clearly the map is a weight preserving bijection. We need only show that it commutes with the crystal operators. Let $R_1 \in \mathcal{C}(u, n)$ and $R_2 \in \mathcal{C}(v, n)$, and let $R = R_1 \star R_2$. Consider $f_i(R_1 \otimes R_2)$. If $\varphi_i(R_1) > \varepsilon_i(R_2)$, then the unpaired elements in row i in R_1 exhaust the unpaired elements in row $i+1$ in R_2 . Therefore, only elements of R_1 are affected by f_i , as in the definition of the tensor product. If instead $\varphi_i(R_1) \leq \varepsilon_i(R_2)$, then all unpaired elements of R_1 in row i are paired with elements of R_2 in row $i+1$, and so only elements of R_2 are affected by f_i . The same reasoning applies to e_i . $\square$

Theorem 5.2. Let $u, v \in S_\infty$ have height n , with v being n -Grassmannian. The map $A \otimes B \mapsto A \boxtimes B$ is a weight-preserving crystal isomorphism from $\mathcal{RC}(u) \otimes \mathcal{RC}(v)$ to $\bigoplus_{w: c_{u,v}^w > 0} \mathcal{RC}(w)^{\oplus c_{u,v}^w}$.

The squash product endows $\mathcal{G}_n$ with a ring structure and $\mathcal{BRC}_n$ with the structure of a right $\mathcal{G}_n$ -module.

Definition 5.3. Define

$$\text{Schub}_{\mathcal{RC}}(w, n) = \sum_{w_R = w, \text{ht}(R) = n} R$$

to be the Schubert element corresponding to w in $\mathcal{BRC}_n$. Also define

$$\text{DSchub}_{\mathcal{RC}}(w, n; y) = \sum_{uv = w, \ell(u) + \ell(v) = \ell(w)} \text{Schub}_{\mathcal{RC}}(v, n) \mathfrak{S}_{u^{-1}}(y)$$

Also define

$$\text{DElem}_{p,k}(n; y) = \sum_{\mathfrak{S}_v(x) = e_{p,k}(x)} \mathfrak{S}_{u^{-1}}(y) \text{DSchub}_{\mathcal{RC}}(v, n; y)$$

The following result is key to the construction of the bounded RC factor algebra.

Theorem 5.3. *Let λ be a partition, and let μ be the corresponding dominant permutation such that $\mathfrak{c}^*(\mu) = \lambda$. Let $\bar{\lambda} = (\lambda_{\ell(\lambda)}, \lambda_{\ell(\lambda)-1}, \dots, \lambda_1)$. Then*

$$\begin{aligned} & \bigotimes_{i=1}^{\ell(\lambda)} \text{DElem}_{\bar{\lambda}_i, \bar{\lambda}_i}(\bar{\lambda}_i; y_{\ell(\lambda)+1-i}) \\ &= \sum_{\ell(u\mu^{-1})=\ell(\mu)-\ell(u)} \text{Schub}_{\mathcal{RC}}(u, n) \mathfrak{S}_{u\mu^{-1}}(y) \end{aligned}$$

5.3. Squash decomposition. The squash decomposition reverses the squash product for a single RC graph. The initial step factors it into an S_n -piece and a Grassmannian piece.

Lemma 5.4. *Suppose $R \in \mathcal{BRC}_n$ is a bounded RC graph of height n . Then there is a unique decomposition*

$$R_* \boxtimes R^* = R$$

where $w_{R_*} \in S_n$ and $R^* \in \mathcal{G}_n$ with $|R^*|$ being maximal.

Proof. It is shown in [Mac91] that the polynomial ring $\mathbb{Z}[x_1, \dots, x_n]$ is a free module over the ring Λ_n of symmetric polynomials with basis given by the Schubert polynomials $\mathfrak{S}_w$ with $w \in S_n$. This is equivalent to saying that the set of polynomials $\mathfrak{S}_w(x)s_\lambda(x)$ with $w \in S_n$ and s_λ a Schur polynomial in n variables is a $\mathbb{Z}$ -basis for $\mathbb{Z}[x_1, \dots, x_n]$. By Theorem 5.2, this theorem also holds at the level of RC graphs. This means that the sums

$$\mathcal{S}_u(n) = \sum_{R \in \mathcal{RC}_n(u), u \in S_n} R$$

form a basis of a free right Λ_n -submodule of $\mathcal{BRC}_n$ under the squash product. This has a $\mathbb{Z}$ -basis of the form

$$\mathcal{S}_u \boxtimes G_\lambda$$

for distinct u and λ . Fixing a single $u \in S_n$, consider the $\mathbb{Z}$ -submodule generated by

$$A \boxtimes G$$

for $A \in \mathcal{RC}_n(u)$ and $G \in \mathcal{G}_n$. This subset has a well-defined crystal action and decomposes into distinct weight spaces, and hence is itself a crystal basis of the $\mathbb{Z}$ -submodule. Considering a fixed RC graph $R \in \mathcal{RC}(w)$ for some $w \in S_\infty$, there is a well defined integer $0 \leq m \leq \ell(w)$ such that for any representation of R in the form $A \boxtimes G$ with $A \in \mathcal{RC}_n(u)$ for some $u \in S_n$ and $G \in \mathcal{G}_n$ with $|G|$ maximal, we have that $|G| = m$. Restricting to the case that R is a highest weight, it follows that if we have two distinct such representations $A \boxtimes G$ and $A' \boxtimes G'$ with $|G| = |G'| = m$, then we have that $A \boxtimes G$ and $A' \boxtimes G'$ have the same weight and generate isomorphic Demazure crystals. Considering the crystal divided difference operator action, it follows that there are RC graphs B and B' such that $B \boxtimes G$ and $B' \boxtimes G'$ are RC graphs if to which we apply the exchange property for a common descent i we have that the results are

$$A \boxtimes G = A' \boxtimes G' = R$$

Raising to highest weights and applying this inductively, we reach the longest element of S_n , which has a unique RC graph, and we conclude that $A = A'$ and $G = G'$. After applying appropriate crystal operators, we conclude that the same holds for any $R \in \mathcal{RC}(w)$. $\square$

Definition 5.5 (Squash factorization). Let $R \in \mathcal{RC}_n(w)$ be an RC graph. Then the following algorithm, by the lemma above, produces a unique result. Let G_1 be the maximal-size Grassmannian RC graph and let R_1 be the unique corresponding S_n RC-graph such that $R_1 \boxtimes G_1 = R$. Resize R_1 to have height equal to its maximum descent. Recursively, if we have constructed G_{i-1} for some $i \geq 1$ and we have a non-identity RC graph R_{i-1} of full height, we construct G_i to be the maximal-size Grassmannian RC graph with $\text{ht}(R_{i-1})$ rows R_i the unique $S_{\text{ht}(R_{i-1})}$ -RC graph with $R_i \boxtimes G_i = R_{i-1}$. We resize R_i to have height equal to its maximum descent. We repeat this process until we reach the identity, at which point we stop. The resulting sequence of $(G_m, G_{m-1}, \dots, G_1)$ of full-rank Grassmannian RC graphs is called the *squash factorization* of R . By construction, we have

$$G_m \boxtimes G_{m-1} \boxtimes \dots \boxtimes G_1 = R.$$

Lemma 5.6. *Suppose $R \in \mathcal{RC}_n(w)$ is an RC graph with squash factorization $(G_m, G_{m-1}, \dots, G_1)$. Then we have that G_i is an RC graph for an elementary symmetric polynomial with $\text{hr}(G_i)$ rows for all $i > 1$.*

Theorem 5.4. *Let the squash factorization of an RC graph R be $G_1 \boxtimes G_2 \boxtimes \dots \boxtimes G_m$. Then the iterated squash product map from $\mathcal{RC}(w_{G_1}) \otimes \dots \otimes \mathcal{RC}(w_{G_m})$ to $\mathcal{RC}(w)$ is a weight-preserving crystal isomorphism.*

Proof. □

6. THE BOUNDED RC FACTOR ALGEBRA

6.1. Basis elements. Having developed the squash product, the squash decomposition, and the Demazure crystal structure, we are now in a position to define the algebra.

Definition 6.1. A *factor key* of size n is an ordered tuple $(G_1, G_2, \dots, G_k)$ of full-rank Grassmannian RC graphs, together with a positive integer n satisfying $n \geq \max_i \text{ht}(G_i)$, where each G_i is an n_i -Grassmannian RC graph with $\text{ht}(G_i) = n_i$ and n_i equal to the unique descent of w_{G_i} (the full-rank condition). We allow the following operations on such tensors to declare them equivalent.

- (1) *Sorting.* If two adjacent factors satisfy $\text{ht}(G_i) > \text{ht}(G_{i+1})$, they are transposed.
- (2) *Merging.* If two adjacent factors G_i and G_{i+1} satisfy $\text{ht}(G_i) = \text{ht}(G_{i+1})$, they are replaced by the single RC graph $G_i \boxtimes G_{i+1}$. This merged graph is again Grassmannian and has the same number of rows.
- (3) *Decomposition.* If a factor G_i represents a Grassmannian permutation u such that $\text{ht}(G_i) < n$ and $u \notin S_{\text{ht}(G_i)}$, then G_i can be resized to height $\text{ht}(G_i) + 1$ and decomposed via the squash factorization into a sequence $(G'_1, G'_2, \dots, G'_k)$ of full-rank Grassmannian factors, with $\text{ht}(G'_j) < \text{ht}(G'_{j+1})$ elementary symmetric and $\text{ht}(G'_k) = \text{ht}(G_i) + 1$, such that

$$G'_1 \boxtimes G'_2 \boxtimes \dots \boxtimes G'_k = G_i.$$

We can then replace G_i with the sequence $(G'_1, G'_2, \dots, G'_k)$, maintaining the order of the tensor factors.

In addition, any identity factor (an RC graph for the identity permutation) is dropped from the tensor.

Proposition 6.2. *For each n , every factor key of height at most n has a unique normal form $G_1 \otimes \dots \otimes G_n$ obtained by applying the above rules until none applies.*

The product on $\mathcal{E}$ is then defined by

$$(G_1 \otimes \dots \otimes G_p) \cdot (H_1 \otimes \dots \otimes H_q) = \text{normalize}(G_1 \otimes \dots \otimes G_p \otimes H_1 \otimes \dots \otimes H_q),$$

extended bilinearly to arbitrary elements.

Theorem 6.1. *The product on $\mathcal{E}$ is associative.*

Proof. Write elements as formal linear combinations of factor keys and expand. Associativity reduces to the statement that for factor keys α, β, γ , the normal form of $(\alpha \otimes \beta \otimes \gamma)$ does not depend on the order in which adjacent normalizations are performed. This follows from the confluence of the rewriting system. □

6.2. The squash map. There is a natural linear surjection $\mathcal{R} : \mathcal{E} \rightarrow \mathcal{BRC}$ that evaluates each factor key by iterated squash products:

$$\mathcal{R}(G_1 \otimes G_2 \otimes \dots \otimes G_k) = (\dots((G_1 \boxtimes G_2) \boxtimes G_3) \dots) \boxtimes G_k.$$

This map is well defined on normal forms because the normalization rules preserve the squash product: merging two equal-size factors via $\boxtimes$ is precisely the squash product, sorting commuting factors preserves the overall squash, and the decomposition rule is the inverse of a squash product and therefore also preserves it.

Remark. The squash map is not a ring homomorphism. Its kernel is not a two-sided ideal of $\mathcal{E}$. This fact, demonstrated by explicit examples already for the Schubert polynomial $\mathfrak{S}_{1432}$, is one of the main reasons the factor algebra is necessary: it retains more structure than the ring of RC graphs alone.

7. THE CRYSTAL ELEMENTARY MONOMIAL REPRESENTATION

7.1. Elementary symmetric RC graphs. For integers $0 \leq p \leq k$, the k -Grassmannian permutation with Lehmer code having a single nonzero entry p in position k (that is, the permutation sending $k \mapsto k + p$ and cyclically adjusting) corresponds to the elementary symmetric polynomial $e_p(x_1, \dots, x_k)$. Its RC graphs form a family $\{M_A^k\}$ indexed by subsets $A \subseteq [k]$ of size p . Each such graph M_A^k is a full-rank Grassmannian RC graph of height k .

The sum

$$\mathbf{e}_p^k = \sum_{\substack{A \subseteq [k] \\ |A|=p}} M_A^k$$

represents the elementary symmetric polynomial $e_p(x_1, \dots, x_k)$ at the level of the algebra $\mathcal{E}$.

7.2. CEM expansion of Schubert polynomials. Every Schubert polynomial can be written as a polynomial in the elementary symmetric polynomials, by a recursive procedure using divided differences. At the level of the factor algebra, this translates into the following.

Definition 7.1. The *crystal elementary monomial* (CEM) representation of a Schubert polynomial $\mathfrak{S}_w$ at size n is the element

$$\mathfrak{S}_w^{\text{CEM}}(n) = \sum_{(G_1, \dots, G_k)} c_{G_1, \dots, G_k} \cdot (G_1 \otimes \dots \otimes G_k) \in \mathcal{E}$$

whose image under the squash map satisfies $\mathcal{R}(\mathfrak{S}_w^{\text{CEM}}(n)) = \mathcal{S}_w(n)$, and where the coefficients $c_{G_1, \dots, G_k}$ are determined by expanding $\mathfrak{S}_w$ in the elementary monomial basis and tracking the crystal structure.

The CEM representation is computed algorithmically through the function `full_CEM`, which expands a Schubert polynomial in terms of elementary symmetric functions via a v -path dictionary (implementing the Lascoux–Schützenberger transition formula), then lifts each monomial $e_{a_1}(x_1, \dots, x_{k_1}) \cdots e_{a_m}(x_1, \dots, x_{k_m})$ to a tensor of Grassmannian RC graphs respecting the crystal structure.

Example 7.2. Consider the permutation $w = [2, 1, 3]$ with $\mathfrak{S}_w = x_1$. At size $n = 2$, the CEM representation consists of a single tensor: the Grassmannian RC graph $M_{\{1\}}^1$ of height 1 with a single crossing at position $(1, 1)$.

For $w = [3, 1, 2]$ with $\mathfrak{S}_w = x_1^2 + x_1x_2 = x_1e_1(x_1, x_2) = e_1(x_1) \cdot e_1(x_1, x_2)$ at size $n = 2$, the CEM representation is a sum of tensor products of full-rank Grassmannian factors whose squash products yield the RC graphs for $[3, 1, 2]$.

7.3. Dominant permutations. A permutation w is *dominant* if its Lehmer code is a partition (a weakly decreasing sequence). For a dominant permutation w_λ with $\mathbf{c}(w_\lambda) = \lambda = (\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m)$, the Schubert polynomial is a product of elementary symmetric polynomials:

$$\mathfrak{S}_{w_\lambda} = \prod_{i=1}^m e_{\lambda_i - \lambda_{i+1}}(x_1, \dots, x_i),$$

where $\lambda_{m+1} = 0$. In the factor algebra, the CEM representation of a dominant permutation is therefore a single tensor monomial with coefficient $+1$, factored as a product of at most one elementary symmetric function per rank, following a pattern dictated by the differences in the partition.

For a general permutation w , one finds a dominant permutation w_λ above w in the left weak Bruhat order, applies crystal divided differences along a reduced word from w_λ to w , and records the resulting expansion. This is the algorithmic content of the CEM representation.

7.4. Multiplication in the factor algebra. Given the CEM representations of $\mathfrak{S}_u$ and $\mathfrak{S}_v$, the product $\mathfrak{S}_u^{\text{CEM}}(n) \cdot \mathfrak{S}_v^{\text{CEM}}(n)$ is computed in $\mathcal{E}$ by distributing and normalizing each pair of tensor monomials. One then applies the squash map $\mathcal{R}$ to obtain an element of $\mathcal{BRC}_n$, and reads off the Littlewood–Richardson coefficients c_{uv}^w by identifying the Schubert expansion.

The hypothesis, verified computationally for permutations up to moderate length, is that this procedure yields non-negative integer coefficients, providing a combinatorial manifestation of the Littlewood–Richardson rule for Schubert calculus.

7.5. Formula for Schubert elements. Let μ be a dominant permutation with $\mathbf{c}^*(\mu) = (\lambda_1, \dots, \lambda_n)$ and let u be a permutation such that $\ell(u\mu) = \ell(\mu) - \ell(u)$. For a positive integer p , we say u and w are p -anchored if there exists a sequence of pairwise distinct positive integers $a_1, a_2, \dots, a_k$ with $a_i \neq p$ for any i such that $w = ut_{pa_1} \cdots t_{pa_k}$ and $\ell(ut_{pa_1} \cdots t_{pa_i}) = \ell(u) + i$ for all i . We have the following formula for Schubert elements:

$$\text{Schub}_{\mathcal{RC}}(u\mu, \lambda_1) = \sum (-1)^{\text{sign}(P)} E_{\lambda_n - \ell(u_{n-1}, u_n)}^{\lambda_n} \otimes \cdots \otimes E_{\lambda_2 - \ell(u_1, u_2)}^{\lambda_2} \otimes E_{\lambda_1 - \ell(u_0, u_1)}^{\lambda_1}$$

where $P = (u_0, \dots, u_n)$ is a sequence of permutations with $u_0 = 1$ and $u_n = u$ such that u_{i-1} and u_i are i -anchored for all i , and $\text{sign}(P)$ is the sum over all i of the number of $j < i$ such that $u_{i-1}(j) \neq u_i(j)$.

ACKNOWLEDGMENTS

Computational verification of the constructions described here was carried out using the Python package `schubmult`.

[Sam24]

REFERENCES

- [Mac91] I. G. Macdonald, *Notes on Schubert Polynomials*, Université Du Québec à Montréal, Dép. de mathématiques et d'informatique, 1991.
- [Sam24] Matthew J. Samuel, *A Molev-Sagan type formula for double Schubert polynomials*, J. Pure Appl. Algebra **228** (2024), no. 7, 107636.