A crystal on decreasing factorizations
in the 0-Hecke monoid

Jennifer Morse*
Department of Mathematics
University of Virginia
Charlottesville, VA 22904, U.S.A.
morsej@virginia.edu

Jianping Pan Wencin Poh Anne Schilling†
Department of Mathematics
University of California
One Shields Ave., Davis, CA, 95616-8633, U.S.A.
{jppan,wpoh,anne}@math.ucdavis.edu

Submitted: Nov 25, 2019; Accepted: May 6, 2020; Published: May 29, 2020
© The authors. Released under the CC BY-ND license (International 4.0).

Abstract

We introduce a type $A$ crystal structure on decreasing factorizations of fully-commutative elements in the 0-Hecke monoid which we call $\star$-crystal. This crystal is a $K$-theoretic generalization of the crystal on decreasing factorizations in the symmetric group of the first and last author. We prove that under the residue map the $\star$-crystal intertwines with the crystal on set-valued tableaux recently introduced by Monical, Pechenik and Scrimshaw. We also define a new insertion from decreasing factorization to pairs of semistandard Young tableaux and prove several properties, such as its relation to the Hecke insertion and the uncrowding algorithm. The new insertion also intertwines with the crystal operators.

Mathematics Subject Classifications: 05E05, 05E10, 14N15, 20G42

1 Introduction

Grothendieck polynomials were introduced by Lascoux and Schützenberger [LS82, LS83] as representatives for the Schubert classes in the $K$-theory of the flag manifold. Their

*Supported by NSF grant DMS–1833333.
†Supported by NSF grants DMS–1760329 and DMS–1764153.
stabilizations were studied by Fomin and Kirillov [FK94]. The stable Grothendieck polynomials, labeled by permutations \( w \in \mathbb{S}_n \), are defined as

\[
G_w(x_1, \ldots, x_m; \beta) = \sum_{(k, h)} \beta^{\ell(h) - \ell(w)} x^k,
\]

where the sum is over decreasing factorizations \([k, h]\) of \( w \) in the 0-Hecke algebra. When \( \beta = 0 \), \( G_w \) specializes to the Stanley symmetric function \( F_w \) [Sta84].

A robust combinatorial picture has been developed for the special case of Grothendieck polynomials indexed by Grassmannian permutations. Buch [Buc02] showed that the Grassmannian Grothendieck polynomials can be realized as the generating functions of semistandard set-valued tableaux:

\[
G_\lambda(x_1, \ldots, x_m; \beta) = \sum_{T \in \text{SVT}^m(\lambda)} \beta^{\text{ex}(T)} x^{\text{wt}(T)},
\]

where \( \text{SVT}^m(\lambda) \) is the set of semistandard set-valued tableaux of shape \( \lambda \) in the alphabet \([m] := \{1, 2, \ldots, m\}\) and \( \text{ex}(T) \) is the excess of \( T \). Recently, Monical, Pechenik and Scrimshaw [MPS18] provided a type \( A_{m-1} \)-crystal structure on \( \text{SVT}^m(\lambda) \) which, in particular, implies that

\[
G_\lambda(x_1, \ldots, x_m; \beta) = \sum_{\mu} \beta^{\mu|\mu| - |\lambda|} M^\mu_\lambda s_\mu(x_1, \ldots, x_m),
\]

where \( M^\mu_\lambda \) is the number of highest-weight set-valued tableaux of weight \( \mu \) in the crystal \( \text{SVT}^m(\lambda) \). Their approach recovers a Schur expansion formula for Grassmannian Grothendieck polynomials given by Lenart [Len00, Theorem 2.2] in terms of flagged increasing tableaux.

In this paper, we define a type \( A \) crystal structure on decreasing factorizations of \( w \) in the 0-Hecke algebra of (1), when \( w \) is fully-commutative [Ste96] (or equivalently 321-avoiding). A permutation \( w \) is fully-commutative if its reduced expressions do not contain any braids. The number of fully-commutative elements of \( \mathbb{S}_n \) is the \( n \)-th Catalan number. The residue map (see Section 2.4) shows that fully-commutative permutations correspond to skew shapes. We call our crystal \( * \)-crystal. It is local in the sense that the crystal operators \( f_i \) and \( e_i \) only act on the \( i \)-th and \((i + 1)\)-th factors of the decreasing factorization. It generalizes the crystal of Morse and Schilling [MS16] for Stanley symmetric functions (or equivalently reduced decreasing factorizations of \( w \)) in the fully-commutative case. We show that the \( * \)-crystal and the crystal on set-valued tableaux intertwine under the residue map (see Theorem 17). We also show that the residue map and the Hecke insertion [BKS+08] are related (see Theorem 24), thereby resolving [MPS18, Open Problem 5.8] in the fully-commutative case. In addition, we provide a new insertion algorithm, which we call \( * \)-insertion, from decreasing factorizations on fully-commutative elements in the 0-Hecke monoid to pairs of (transposes of) semistandard Young tableaux of the same shape (see Definition 27 and Theorem 35), which intertwines with crystal operators.
(see Theorem 57). This recovers the Schur expansion of $\mathfrak{S}_w$ of Fomin and Greene [FG98] when $w$ is fully-commutative, stating that

$$\mathfrak{S}_w = \sum_\mu \beta_{|\mu|-\ell(w)} g^\mu_w s_\mu,$$

where

$$g^\mu_w = |\{ T \in \text{SSYT}^\mu \mid w_C(T) \equiv w \}|,$$

and $w_C(T)$ is the column reading word of $T$ (see Remark 58). We also show that the composition of the residue map with the $\star$-insertion is related to the uncrowding algorithm [Buc02] (see Theorem 64). Other insertion algorithms have recently been studied in [CP19].

The paper is organized as follows. In Section 2, we introduce the $\star$-crystal on decreasing factorizations in the 0-Hecke monoid and show that it intertwines with the crystal on semistandard set-valued tableaux [MPS18] under the residue map. In Section 3, we discuss two insertion algorithms for decreasing factorizations. The first is the Hecke insertion introduced by Buch et al. [BKS+08] and the second is the new $\star$-insertion. In Section 4, properties of the $\star$-insertion are discussed. In particular, we prove that it intertwines with the crystal operators and that it relates to the uncrowding algorithm. We conclude in Section 5 with some discussions about the non-fully-commutative case.

## 2 The $\star$-crystal

In this section, we define the $K$-theoretic generalization of the crystal on decreasing factorizations by Morse and Schilling [MS16] when the associated word is fully-commutative. The underlying combinatorial objects are decreasing factorizations in the 0-Hecke monoid introduced in Section 2.1. The $\star$-crystal on these decreasing factorizations is defined in Section 2.2. We review the crystal structure on set-valued tableaux introduced by Monical, Pechenik and Scrimshaw [MPS18] in Section 2.3. The residue map and the proof that it intertwines the $\star$-crystal and the crystal on set-valued tableaux is given in Section 2.4.

### 2.1 Decreasing factorizations in the 0-Hecke monoid

The symmetric group $\mathfrak{S}_n$ for $n \geq 1$ is generated by the simple transpositions $s_1, s_2, \ldots, s_{n-1}$ subject to the relations

$$s_i s_j = s_j s_i, \quad \text{if } |i - j| > 1,$$

$$s_i s_{i+1} s_i = s_{i+1} s_i s_{i+1}, \quad \text{for } 1 \leq i < n - 1,$$

$$s_i^2 = 1, \quad \text{for } 1 \leq i \leq n - 1.$$

A reduced expression for an element $w \in \mathfrak{S}_n$ is a word $a_1 a_2 \ldots a_\ell$ with $a_i \in [n - 1] := \{1, 2, \ldots, n - 1\}$ such that

$$w = a_1 \cdots a_\ell$$  \hspace{1cm} (3)

and $\ell$ is minimal among all words satisfying (3). In this case, $\ell$ is called the length of $w$. 

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), \#P2.29
Definition 1. The 0-Hecke monoid $\mathcal{H}_0(n)$, where $n \geq 1$ is an integer, is the monoid of finite words generated by positive integers in the alphabet $[n - 1]$ subject to the relations

$$
pq = qp \quad \text{if } |p - q| > 1,
pqp = qpq \quad \text{for all } p, q,
pp = p \quad \text{for all } p.
$$

We may form an equivalence relation $\equiv_{\mathcal{H}_0}$ on all words in the alphabet $[n - 1]$ based on the relations (4). The equivalence classes are infinite since the last relation changes the length of the word. We say that a word $a = a_1a_2\ldots a_\ell$ is reduced if $\ell \geq 0$ is the smallest among all words in $\mathcal{H}_0(n)$ equivalent to $a$. In this case, $\ell$ is the length of $a$. Note that $\mathcal{H}_0(n)$ is in bijection with $S_n$ by identifying the reduced word $a_1a_2\ldots a_\ell$ in $\mathcal{H}_0(n)$ with $s_{a_1}s_{a_2}\ldots s_{a_\ell} \in S_n$. We say $w \in \mathcal{H}_0(n)$ or $S_n$ is fully-commutative or 321-avoiding if none of the reduced words equivalent to $w$ contain a consecutive braid subword of the form $i i + 1 i$ or $i i - 1 i$ for any $i \in [n - 1]$.

Remark 2. Any (not necessarily reduced) word $w \in \mathcal{H}_0(0)$ containing a consecutive braid subword is not fully-commutative.

Definition 3. A decreasing factorization of $w \in \mathcal{H}_0(n)$ into $m$ factors is a product of the form

$$
h = h^m \ldots h^2 h^1,
$$

where the sequence in each factor

$$
h^i = h^i_1h^i_2\ldots h^i_{\ell_i}
$$

is either empty (meaning $\ell_i = 0$) or strictly decreasing (meaning $h^i_1 > h^i_2 > \cdots > h^i_{\ell_i}$) for each $1 \leq i \leq m$ and $h \equiv_{\mathcal{H}_0} w$ in $\mathcal{H}_0(n)$.

The set of all possible decreasing factorizations into $m$ factors is denoted by $\mathcal{H}_m^\ast$ or $\mathcal{H}^m(n)$ if we want to indicate the value of $n$. We call $\text{ex}(h) = (\text{len}(h^1), \text{len}(h^2), \ldots, \text{len}(h^m))$ the excess of $h$, where $\text{len}(h)$ is the number of letters in $h$ and $\ell$ is the length of $w$. We say $h$ is fully-commutative (or 321-avoiding) if $w$ is fully-commutative.

2.2 The $\ast$-crystal

Let $\mathcal{H}^{m,\ast}$ be the set of fully-commutative decreasing factorizations in $\mathcal{H}^m$. We introduce a type $A_{m-1}$ crystal structure on $\mathcal{H}^{m,\ast}$, which we call the $\ast$-crystal. This generalizes the crystal for Stanley symmetric functions [MS16] (see also [Len04]).

Definition 4. For any $h = h^m \ldots h^2 h^1 \in \mathcal{H}^{m,\ast}$, we define crystal operators $e_i^\ast$ and $f_i^\ast$ for $i \in [m - 1]$ and a weight function $\text{wt}(h)$. The weight function is determined by the length of the factors

$$
\text{wt}(h) = (\text{len}(h^1), \text{len}(h^2), \ldots, \text{len}(h^m)).
$$

To define the crystal operators $e_i^\ast$ and $f_i^\ast$, we first describe a pairing process:
• Start with the largest letter \( b \) in \( h^{i+1} \), pair it with the smallest \( a \geq b \) in \( h^i \). If there is no such \( a \), then \( b \) is unpaired.

• The pairing proceeds in decreasing order on elements of \( h^{i+1} \) and with each iteration, previously paired letters of \( h^i \) are ignored.

If all letters in \( h^i \) are paired, then \( f_i^* \) annihilates \( h \). Otherwise, let \( x \) be the largest unpaired letter in \( h^i \). The crystal operator \( f_i^* \) acts on \( h \) in either of the following ways:

1. If \( x + 1 \in h^i \cap h^{i+1} \), then remove \( x + 1 \) from \( h^i \), add \( x \) to \( h^{i+1} \).
2. Otherwise, remove \( x \) from \( h^i \) and add \( x \) to \( h^{i+1} \).

If all letters in \( h^{i+1} \) are paired, then \( e_i^* \) annihilates \( h \). Let \( y \) be the smallest unpaired letter in \( h^{i+1} \). The crystal operator \( e_i^* \) acts on \( h \) in either of the following ways:

1. If \( y - 1 \in h^i \cap h^{i+1} \), then remove \( y - 1 \) from \( h^{i+1} \), add \( y \) to \( h^i \).
2. Otherwise, remove \( y \) from \( h^{i+1} \) and add \( y \) to \( h^i \).

It is not hard to see that \( e_i^* \) and \( f_i^* \) are partial inverses of each other.

**Example 5.** Let \( h = (7532)(621)(6) \), then

\[
\begin{align*}
f_1^*(h) &= 0, & e_1^*(h) &= (7532)(62)(61), \\
f_2^*(h) &= (75321)(61)(6), & e_2^*(h) &= (753)(6321)(6).
\end{align*}
\]

**Remark 6.** Compared to [MS16], one pairs a letter \( b \) in \( h^{i+1} \) with the smallest letter \( a \geq b \) in \( h^i \) rather than \( a > b \).

**Proposition 7.** Let \( h = h^m \ldots h^1 \in \mathcal{H}^{m,*} \) such that \( f_i^*(h) \neq 0 \). Then \( f_i^*(h) \in \mathcal{H}^{m,*} \), \( f_i^*(h) \equiv_{h_0} h \), and \( \text{ex}(f_i^*(h)) = \text{ex}(h) \). Furthermore, the \( j \)-th factor in \( f_i^*(h) \) and \( h \) agrees for \( j \notin \{i, i+1\} \). Analogous statements hold for \( e_i^* \).

**Proof.** Suppose \( \tilde{h} := f_i^*(h) \neq 0 \). Then by definition of \( f_i^* \), \( \tilde{h} = h^m \ldots h^{i+2} \tilde{h}^{i+1} \tilde{h}^i h^{i-1} \ldots h^1 \) and \( h^i \) is unchanged for \( j \notin \{i, i+1\} \). In addition, the number of factors does not change.

To see \( h \equiv_{h_0} \tilde{h} \), it suffices to show that \( h^{i+1} h^i \equiv_{h_0} \tilde{h}^{i+1} \tilde{h}^i \). Let \( x \) be the largest unpaired letter in \( h^i \). By the bracketing procedure this implies that \( x \notin h^{i+1} \). We can write \( h^{i+1} \) as \( w_1 w_2 \), where \( w_1 \) is a word containing only letters greater than \( x \), and \( w_2 \) is a word containing only letters smaller than \( x \). We can write \( h^i \) as \( w_3 x w_4 \), where \( w_3 \) contains only letters greater than \( x \) and \( w_4 \) contains only letters smaller than \( x \).

The pairing process will result in one of the following two cases:

1. If \( x + 1 \in h^i \cap h^{i+1} \), then obtain \( \tilde{h}^i \) by removing \( x + 1 \) from \( h^i \), and \( \tilde{h}^{i+1} \) by adding \( x \) to \( h^{i+1} \).
2. Otherwise, obtain \( \tilde{h}^i \) by removing \( x \) from \( h^i \) and obtain \( \tilde{h}^{i+1} \) by adding \( x \) to \( h^{i+1} \).
We first argue that in either case we must have $x - 1 \notin w_2$. Assume $x - 1 \in w_2$ and let $k$ be the largest number such that the interval $[x - k, x - 1] \subseteq w_2$. By assumption $k \geq 1$. In order for $x$ to be the largest unpaired letter in $h^i$, $[x - k, x - 1]$ must be contained in $w_4$. We can write $w_2 = (x - 1) \ldots (x - k)w'_2$ and $w_4 = (x - 1) \ldots (x - k)w'_4$, where all letters in $w'_2$ are smaller than $x - k - 1$. When $k = 1$, we have the following subword
\[(x - 1)w'_2w_3(x - 1) \equiv_h 1_0 w_2'w_3(x - 1)x(x - 1),\]
which contains a braid $(x - 1)x(x - 1)$. When $k > 1$, we also have the following subword
\[(x - k)w'_2w_3(x - 1) \ldots (x - k + 1)(x - k) \equiv_h w_2'w_3(x - 1) \ldots (x - k + 2)(x - k)(x - k + 1)(x - k),\]
which also contains a braid.

**Case (1):** Let $k$ be the largest letter such that $[x + 1, x + k] \subseteq w_3$. Clearly $k \geq 1$. Suppose $k > 1$, then we can write $w_3 = w'_3(x + k) \ldots (x + 1)$. Since $x$ is the largest unpaired letter in $h^i$, everything in $[x + 1, x + k] \subseteq w_3$ must be paired. The letter $x + 1$ in $w_3$ is paired with $x + 1 \in w_1$, which implies that $x + i$ in $w_3$ is paired with $x + i \in w_1$ for all $1 \leq i \leq k$. This implies that $[x + 1, x + k] \subseteq w_1$. Then we have the following subword
\[(x + 1)w_2w'_3(x + k) \ldots (x + 2)(x + 1) \equiv_h w_2w'_3(x + k) \ldots (x + 1)(x + 2)(x + 1)\]
which contains a braid. Thus, we must have $k = 1$, which implies that $x + 2 \notin w_3$. Write $w_1 = w'_1(x + x)$. Then by direct computation
\[h^{i+1}h^i \equiv_h (w'_1(x + 1)w_2w'_3(x + 1)xw_4 \equiv_h w'_1(x + 1)(x + 1)w_2w'_3xw_4) \equiv_h (w'_1(x + 1)xw_4)(w'_3xw_4) = h^{i+1}h^i.\]

**Case (2):** We claim that if $x + 1 \notin h^{i+1}$, then $x + 1 \notin h^i$. Otherwise the $x + 1 \in h^i$ must be paired with some $z \in h^{i+1}$, so we have $z \leq x + 1$. But $x$ is unpaired, which implies $z > x$, that gives us a contradiction. Hence $x + 1 \notin w_3$. Recall that $x - 1 \notin w_2$. Therefore, by a straightforward computation
\[h^{i+1}h^i = w_1w_2w_3xw_4 \equiv_h (w_1xw_2)(w_3w_4) \equiv_h h^{i+1}h^i.\]
The above arguments show that $h^{i+1}h^i \equiv_h h^{i+1}h^i$, thus $h \equiv_h h$, and the total length of the decreasing factorization are unchanged under $f_i^*$. Furthermore, the excess remains unchanged under $f_i^*$.

Similar arguments hold for $e_i^*$. \qed

**Remark 8.** Here we summarize several results from the proof that will be needed later. Namely, if $x$ is the largest unpaired letter in $h^i$, then

- $x - 1 \notin h^{i+1}$.

- One and only one of the three statements hold: $x + 1 \in h^{i+1} \cap h^i$, $x + 1 \notin h^{i+1} \cup h^i$, and $x + 1 \in h^{i+1}, x + 1 \notin h^i$.

It will be shown in Section 2.4 that $H^{m,*}$ is indeed a Stembridge crystal of type $A_{m-1}$ (for an introduction to crystal and terminology, see [BS17]).
2.3 The crystal on set-valued tableaux

In this section, we review the type $A$ crystal structure on set-valued tableaux introduced in [MPS18]. In fact, in [MPS18] the authors only considered the crystal structure on straight-shaped set-valued tableaux. Here we consider the crystal on skew shapes as well, see Theorem 11.

We use French notation for partitions $\lambda = (\lambda_1, \lambda_2, \ldots)$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq 0$, that is, in the Ferrers diagram for $\lambda$, the largest part $\lambda_1$ is at the bottom.

**Definition 9 ([Buc02]).** A semistandard set-valued tableau $T$ is the filling of a skew shape $\lambda/\mu$ with nonempty subsets of positive integers such that:

- for all adjacent cells $A$, $B$ in the same row with $A$ to the left of $B$, we have $\max(A) \leq \min(B)$,
- for all adjacent cells $A$, $C$ in the same column with $A$ below $C$, we have $\max(A) < \min(C)$.

The weight of $T$, denoted by $\text{wt}(T)$, is the integer vector whose $i$-th component counts the number of $i$'s that occur in $T$. The excess of $T$ is defined as $\text{ex}(T) = |\text{wt}(T)| - |\lambda|$. We denote the set of all semistandard set-valued tableaux of shape $\lambda/\mu$ by $\text{SVT}(\lambda/\mu)$. Similarly, if the maximum entry is restricted to $m$, the set is denoted by $\text{SVT}^m(\lambda/\mu)$.

We now review the crystal structure on semistandard set-valued tableaux given in [MPS18]. We state the definition on skew shapes rather than just straight shapes.

**Definition 10.** Let $T \in \text{SVT}^m(\lambda/\mu)$. We employ the following pairing rule for letters $i$ and $i+1$. Assign $-$ to every column of $T$ containing an $i$ but not an $i+1$. Similarly, assign $+$ to every column of $T$ containing an $i+1$ but not an $i$. Then, successively pair each $+$ that is to the left of and adjacent to a $-$, removing all paired signs until nothing can be paired.

The operator $f_i$ changes the $i$ in the rightmost column with an unpaired $-$ (if this exists) to $i+1$, except if the cell $b$ containing that $i$ has a cell to its right, denoted $b^\rightarrow$, that contains both $i$ and $i+1$. In that case, $f_i$ removes $i$ from $b^\rightarrow$ and adds $i+1$ to $b$. Finally, if no unpaired $-$ exists, then $f_i$ annihilates $T$.

Similarly, the operator $e_i$ changes the $i+1$ in the leftmost column with an unpaired $+$ (if this exists) to $i$, except if the cell $b$ containing that $i+1$ has a cell to its left, denoted $b^\leftarrow$, that contains both $i$ and $i+1$. In that case, $e_i$ removes $i+1$ from $b^\leftarrow$ and adds $i$ to $b$. Finally, if no unpaired $+$ exists, then $e_i$ annihilates $T$.

Based on the pairing procedure above, $\varphi_i(T)$ is the number of unpaired $-$ while $\varepsilon_i(T)$ is the number of unpaired $+$.

One can easily show that the crystal on $\text{SVT}^m(\lambda/\mu)$ of Definition 10 defines a semi-normal crystal (for definitions see [BS17]). It was proved in [MPS18, Theorem 3.9] that the above described operators $e_i$ and $f_i$ define a type $A_{m-1}$ Stembridge crystal structure on $\text{SVT}^m(\lambda)$. We claim that their proof goes through also for skew shapes.
Theorem 11. The crystal $\text{SVT}^m(\lambda/\mu)$ of Definition 10 is a Stembridge crystal of type $A_{m-1}$.

Proof. Since the proof is exactly the same as in [MPS18, Theorem 3.9], we just state the outline and give a brief description. For details we refer to [MPS18].

First note that the signature rule given by column-reading is compatible with the signature rule given by row-reading (top to bottom, left to right, and arrange the letters in the same cell by descending order) by semistandardness. Hence we may consider the crystal to live inside the tensor product of its rows. A single-row semistandard set-valued tableaux of a fixed shape is isomorphic to a Stembridge crystal, as shown in [MPS18, Proposition 3.5]:

$$\Phi_s: \text{SVT}^m(s\Lambda_1) \to \bigoplus_{k=1}^{m} B((s-1)\Lambda_1 + \Lambda_k),$$

where $\Lambda_k$ are the fundamental weights of type $A_{m-1}$.

Let $\lambda = (\lambda_1, \ldots, \lambda_\ell)$ and $\mu = (\mu_1, \ldots, \mu_\ell)$ (the last couple $\mu_i$ could be zero) be two partitions such that $\mu \subseteq \lambda$. Construct the map below, which is a strict crystal embedding:

$$\Psi: \text{SVT}^m(\lambda/\mu) \to \text{SVT}^m((\lambda_1 - \mu_1)\Lambda_1) \otimes \text{SVT}^m((\lambda_2 - \mu_2)\Lambda_1) \otimes \cdots \otimes \text{SVT}^m((\lambda_\ell - \mu_\ell)\Lambda_1).$$

Thus, we have a strict crystal embedding:

$$(\Phi_{\lambda_1-\mu_1} \oplus \cdots \oplus \Phi_{\lambda_\ell-\mu_\ell}) \circ \Psi: \text{SVT}^m(\lambda/\mu) \to \bigotimes_{j=1}^{\ell} \left( \bigoplus_{k=1}^{m} B((\lambda_j - \mu_j)\Lambda_1 + \Lambda_k) \right).$$

Since $\text{SVT}^m(\lambda/\mu)$ is a seminormal crystal, we can conclude that it is a Stembridge crystal.

2.4 The residue map

In this section, we define the residue map from set-valued tableaux of skew shape to fully-commutative decreasing factorizations in the 0-Hecke monoid. We then show in Theorem 17 that the residue map intertwines with the crystal operators, proving that $\mathcal{H}^{m,*}$ is indeed a crystal of type $A_{m-1}$ (see Corollary 18).

Definition 12. Given $T \in \text{SVT}^m(\lambda/\mu)$, we define the residue map $\text{res}: \text{SVT}^m(\lambda/\mu) \to \mathcal{H}^m$ as follows. Associate to each cell $(i, j)$ in $\lambda/\mu$ its content $\ell(\lambda) + j - i$, where $\ell(\lambda)$ is the number of parts in $\lambda$. Produce a decreasing factorization $\mathbf{h} = h^m h^{m-1} \cdots h^2 h^1$ by declaring $h^k$ to be the (possibly empty) sequence formed by taking the contents of all cells in $T$ containing the entry $k$ and then arranging the contents in decreasing order. This defines $\text{res}(T) := \mathbf{h}$.

Example 13. Let $T$ be the set-valued tableau of skew shape $(2, 2)/(1)$

$$T = \begin{array}{ccc}
23 & 3 \\
12 & &
\end{array}$$

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29
The content of each cell in $T$ is denoted by a subscript as follows:

\[
\begin{array}{|c|c|}
\hline
23 & 3_2 \\
\hline
12 & 3 \\
\hline
\end{array}
\]

To read off the third factor, we search for all cells with an entry 3; these cells have contents 1 and 2, so we have 21 in the third factor. Altogether, we obtain $\text{res}(T) = (21)(31)(3) \in \mathcal{H}^3$.

The image of the residue map $\text{res}$ is $\mathcal{H}^{m,*}$, the set of fully-commutative decreasing factorizations into $m$ factors. In fact, $\text{res}$ is a bijection from semistandard set-valued skew tableaux on the alphabet $[m]$ to $\mathcal{H}^{m,*}$ up to shifts in the skew shape.

For this purpose, let us describe the inverse of the residue map. Let $h = h^m h^{m-1} \ldots h^2 h^1 \in \mathcal{H}^{m,*}$. Begin by filling the diagonals of content that appear in $h^m$ by the entry $m$. As the resulting $T$ is supposed to be of skew shape, the cells containing $m$ along increasing diagonals need to go weakly down from left to right. If these diagonals are consecutive, then the cells have to be in the same row of $T$ since $T$ is semistandard. Continue the procedure above by putting entry $i$ into the diagonals specified by $h^i$ for all $i = m - 1, m - 2, \ldots, 1$, applying the condition that the resulting filling should be semistandard.

**Proposition 14.** If $h = h^m h^{m-1} \ldots h^2 h^1 \in \mathcal{H}^{m,*}$, then the above algorithm is well-defined up to shifts along diagonals. It produces a skew semistandard set-valued tableau $T$ such that $\text{res}(T) = h$.

**Proof.** We shall show more generally that at any given stage in the algorithm for the inverse of the residue map above, the tableau $T$ produced is of skew shape if and only if $h$ is fully-commutative.

Assume that $T$ is not of skew shape. Consider the earliest stage in the algorithm when the produced tableau is not of skew shape. Then, either one of the following cases must have occurred for the first time.

**Case 1:** There are adjacent cells with nonempty sets $A$ and $B$ (where $\max(A) \leq \min(B)$) in the same row on diagonals $i$ and $i+1$ respectively with no cells appearing directly below these cells, as illustrated on the left side of Figure 1. Moreover, by minimality, we have an integer $x$ with the following properties:

1. $i + 1 \in h^x$ and $x < \min(A)$,

2. there does not exist a $y$ with $x \leq y < \min(B)$ and $i + 2 \in h^y$.

By applying semistandardness, a cell containing $x$ is created directly below the cell containing the set $A$ as in the right side of Figure 1. Furthermore, by (2), for all $x \leq y < \min(B)$, we have that every letter in $h^y$ is either at most $i + 1$ or at least $i + 3$. It follows that, after possibly applying commutativity ($i + 1$ with letters at most $i - 1$ or at least $i + 3$) and the idempotent relation, $h_{\min(B)} \ldots h_{x+1} h^x$ is equivalent to one
containing the braid subword $i + 1$ $i$ $i + 1$. This implies that $h$ is equivalent to a Hecke word containing the same braid subword.

**Case 2:** There are adjacent cells with nonempty sets $A$ and $B$ in the same column on diagonals $i + 1$ and $i$ respectively with no cells appearing directly to the left of these cells, as illustrated on the left side of Figure 2. Moreover, by minimality, we have an integer $x$ with the following properties:

1. $i \in h^x$ and $x \leq \min(A)$,
2. there does not exist a $y$ with $x < y \leq \min(B)$ and $i - 1 \in h^y$.

By applying semistandardness, a cell containing $x$ is created directly to the left of the cell containing the set $A$ as in the right side of Figure 2. Furthermore, by (2), for all $x < y \leq \min(B)$, we have that every letter in $h^y$ is either at most $i - 2$ or at least $i$. Similar to the argument in Case 1, $h^{\min(B)} \ldots h^{x+1}h^x$ is equivalent to one containing the braid subword $i$ $i + 1$ $i$. This implies that $h$ is equivalent to a word in $H_0(n)$ containing the same braid subword.

The above arguments imply that the image of $\text{res}$ is contained in $H^{m,*}$. Conversely, if $h$ is fully-commutative, then the algorithm for $\text{res}^{-1}$ does not produce Case 1 or Case 2 above and hence the resulting tableau $T$ is of skew shape which in turn implies that the algorithm is well-defined (up to shifts along the diagonal if a gap of size at least 3 occurs in the labels).

If the skew shape $\lambda/\mu$ of the tableau $T$ is known, then one may simplify the procedure above noting that the filling of $i$ specified by letters in $h^i$ must occur along a horizontal strip for all $i = m, m - 1, \ldots, 1$. In this case, the recovered tableau $T$ is unique and there is no shift ambiguity if a gap of size at least 3 occurs in the labels.

**Example 15.** Let $h = (61)(752)(75)(762)$ be a decreasing factorization of $w = 651762$. 

Figure 1: A forbidden case while inverting the residue map.

![Figure 1](image1)

Figure 2: Another forbidden case while inverting the residue map.

![Figure 2](image2)
In the algorithm for the inverse of the residue map, the entry 4 is placed on diagonal 1 and 6, respectively. Due to semistandardness, the entry 3 in diagonal 2 must be placed below the 4 in diagonal 1, while the 3's in diagonals 5 and 7 are respectively to the left and below the 4 in diagonal 6. Continuing with the remaining fillings, we have two possibilities:

\[
T_1 = \begin{array}{cccc}
4 & 1 & 3 & 4 \\
13 & 2 & 3 & 4 \\
23 & 5 & 4 & 6 \\
1 & 6 & 12 & 3 \\
\end{array},
\]

or

\[
T_2 = \begin{array}{cccc}
4 & 1 & 3 & 4 \\
13 & 2 & 3 & 4 \\
23 & 5 & 4 & 6 \\
1 & 6 & 12 & 3 \\
\end{array},
\]

where \( T_1 \in \text{SVT}^4((4, 4, 1, 1)/(2, 2)) \) and \( T_2 \in \text{SVT}^4((3, 3, 1, 1)/(1, 1, 1)) \). Note that they indeed just differ by a shift along diagonals as stated in Proposition 14.

**Example 16.** Let \( h = (8431)(863)(8654)(941) \) be a decreasing factorization of \( w = 84396541 \). Suppose that \( h = \text{res}(T) \), where \( T \in \text{SVT}^4(\lambda/\mu) \) with

\[
\lambda/\mu = (5, 5, 4, 2, 1)/(4, 4, 1, 1).
\]

Then, we fill in 4 along the diagonals with labels 1, 3, 4, 8 respectively, noting that the 4 in diagonal 4 is to the right of the 4 in diagonal 3 (due to the semistandardness of \( T \)). Continuing with the remaining fillings, we have

\[
T = \begin{array}{cccc}
14 & 3 & 4 & 4 \\
12 & 4 & 2 & 3 \\
23 & 4 & 2 & 3 \\
23 & 4 & 2 & 3 \\
\end{array}.
\]
Theorem 17. The crystal on set-valued tableaux $\text{SVT}^m(\lambda/\mu)$ and the crystal on decreasing factorizations $\mathcal{H}^{m,*}$ intertwine under the residue map. That is, the following diagrams commute:

\[
\begin{array}{ccc}
\text{SVT}^m(\lambda/\mu) & \xrightarrow{\text{res}} & \mathcal{H}^{m,*} \\
\downarrow f_k & & \downarrow f^*_k \\
\text{SVT}^m(\lambda/\mu) & \xrightarrow{\text{res}} & \mathcal{H}^{m,*}
\end{array}
\]

\[
\begin{array}{ccc}
\text{SVT}^m(\lambda/\mu) & \xrightarrow{\text{res}} & \mathcal{H}^{m,*} \\
\downarrow e_k & & \downarrow e^*_k \\
\text{SVT}^m(\lambda/\mu) & \xrightarrow{\text{res}} & \mathcal{H}^{m,*}
\end{array}
\]

Proof. Let $T \in \text{SVT}^m(\lambda/\mu)$, $h = \text{res}(T)$ and $\ell = \ell(\lambda)$. We prove the following three statements associated to $f_k(T)$ and $f^*_k(h)$.

(1) We claim that if there is no unpaired $k$ in $T$, then $f^*_k$ annihilates $h$. Furthermore, if the rightmost unpaired $k$ in cell $b$ of $T$ has content $x$, then $x$ is also the largest unpaired letter in $h^k$.

For the proof of (1) it suffices to notice that the signature rule on tableaux is equivalent to the pairing process for decreasing factorizations of $\mathcal{H}_0(n)$. We rephrase the pairing procedure for decreasing factorizations on tableaux:

- At the beginning, no letter is paired.
- Then start with the rightmost column and work westward.
- Successively, for each $k + 1$, compute its content $a$, then pair it with the $k$ of smallest content weakly greater than $a$ that is yet unpaired.

Next, we argue that the signature rule yields the same result on the rightmost unpaired letter. Assume we are looking at cell $b$ containing the current $k + 1$ with content $a$.

Case (a): Suppose there is no unpaired $k$ with content $a$ but at least one unpaired $k$ with strictly greater content(s). Then pair it with the current $k + 1$. This is the direct signature rule.

Case (b): Suppose there is no unpaired $k$ with content weakly greater than $a$, then this $k + 1$ is unpaired. This is also the direct signature rule.

Case (c): Suppose there is an unpaired $k$ with content $a$. Then it must be either in the same cell $b$, or one row below and one column to the left of $b$ on the diagonal labeled $a$. If they are in the same cell, then the pairing is the direct signature rule.

Otherwise, there must be cells to the left and below $b$ since the shape is skew. Suppose cell $b$ is in row $r$. Consider the rightmost entry in cell $(r, j)$ in row $r$ containing a $k + 1$, and the leftmost entry in cell $(r - 1, q)$ in row $r - 1$ containing a $k$. Considering this as the first of a consecutive occurrence, cell $b$ is cell $(r, j)$, so we have $\ell + j - r = a$. By semistandardness and the condition that the shape is skew, we can partially fill out the involved subtableau of $T$ for rows $r - 1, r$ from column $q$ to $j$.
All the cells \((s,t)\) with \(q < t < j\) and \(s \in \{r,r−1\}\) and the cells \((r,q)\) and \((r−1,j)\) are single-valued by semistandardness as shown in the above figure.

From the \(k+1\) in \((r,j)\), we start the pairing process. First, we claim that the \(k\) in cell \((r−1,j)\) must be unpaired at this point. Suppose that there is a \(k+1\) to the east of cell \((r,j)\) with content smaller or equal to \(ℓ+j−r+1\), then it must be cell \((r,j+1)\), which violates that \((r,j)\) is the rightmost cell in row \(r\) containing a \(k+1\). Then the pairing says the \(k+1\) in cell \((r,t)\) pairs with the \(k\) in cell \((r−1,t−1)\) for \(q < t \leq j\). Lastly, the \(k+1\) in cell \((r,q)\) has to pair with the previously unpaired \(k\) with label greater or equal to \(ℓ+q−r\) and smaller than \(ℓ+j−r+1\).

Although the pairing is different than the usual signature rule pairing, which pairs \(k+1,k\) in the same column, the \(2(j−q+1)\) letters end up being paired. Since it will not influence which one will be the rightmost unpaired letter, it is still equivalent to the signature rule.

So in any case, the pairing is equivalent to the signature rule. Thus, the rightmost unpaired \(k\) in \(T\) corresponds to the largest unpaired letter in \(h^k\).

(2) We claim that if \(f_k\) changes the rightmost unpaired \(k\) in \(T\) to a \(k+1\) (with content \(x\)) without moving it, then \(f_k^*\) moves a letter \(x\) from \(h^k\) to \(h^{k+1}\).

Since \(f_k\) does not need to move any letter, it means the cell to the right of \(b\), denoted by \(b^\rightarrow\), does not contain a \(k\). It is the only cell with content \(x+1\) that could contain a \(k\). This implies that \(x+1 \notin h^k\). By Definition 4, \(f_k^*\) moves \(x\) from \(h^k\) to \(h^{k+1}\).

(3) We claim the following. If \(f_k\) changes a \(k\) from \(b^\rightarrow\) into a \(k+1\) and moves to cell \(b\), then \(f_k^*\) removes an \(x+1\) from \(h^k\) and changes it to an \(x\) in \(h^{k+1}\).

That \(f_k\) needs to move a number means that \(k\) and \(k+1\) are in \(b^\rightarrow\), which implies that \(x+1 \in h^k \cap h^{k+1}\). By Definition 4, \(f_k^*\) removes the \(x+1\) from \(h^k\) and adds an \(x\) to \(h^{k+1}\).

We have proved the three statements and they complete the proof that \(f_k\) and \(f_k^*\) intertwine under the residue map. The proof is similar for \(e_k\) and \(e_k^*\).

**Corollary 18.** The set \(H^{m,*}\), together with crystal operators \(e_i^*\) and \(f_i^*\) for \(1 \leq i < m\) and weight function \(wt\) defined in Definition 4, is a Stembridge crystal.

**Proof.** By Theorem 17 and the fact that the residue map preserves the weight and is invertible, this follows from the fact that \(SVT^m(λ/µ)\) is a Stembridge crystal proven in [MPS18, Theorem 3.9] (see also Theorem 11).

**Example 19.** Consider the tableau \(T\) (with labels in red) given by

\[
T = \begin{array}{c}
3 \\
12
\end{array}
\begin{array}{c}
13
\end{array}
\]

with \(\text{res}(T) = (31)(3)(32)\).

For the crystal operators on set-valued tableaux we obtain

\[
f_1(T) = \begin{array}{c}
3 \\
12
\end{array}
\begin{array}{c}
23
\end{array}
\]

\[\text{the electronic journal of combinatorics 27(2) (2020), #P2.29 13}\]
with \( \text{res}(f_1(T)) = (31)(32)(2) \). Then it can be easily checked that the following diagram commutes:

\[
\begin{array}{ccc}
T &=& \begin{array}{c}
3_1 \\
1_2 \\
123_3
\end{array} \\
\downarrow{f_1} & \downarrow{\text{res}} & \downarrow{f_1^*} \\
\begin{array}{c}
3_1 \\
12_3 \\
123_3
\end{array} &=& (31)(3)(32)
\end{array}
\]

3 Insertion algorithms

In this section, we discuss two insertion algorithms for decreasing factorizations in \( H^m \) (resp. \( H^{m,\star} \)). The first is the Hecke insertion introduced by Buch et al. [BKS+08], which we review in Section 3.1. We prove a relationship between Hecke insertion and the residue map (see Theorem 24). In particular, this proves [MPS18, Open Problem 5.8] for fully-commutative permutations. The second insertion is a new insertion, which we call \( \star \)-insertion, introduced in Section 3.2. It goes from fully-commutative decreasing factorizations in the 0-Hecke monoid to pairs of (transposes of) semistandard tableaux of the same shape and is well-behaved with respect to the crystal operators.

3.1 Hecke insertion

Hecke insertion was first introduced in [BKS+08] as column insertion. Here we state the row insertion version as in [PP16]. In this section, we represent a decreasing factorization \( h = h_m h_{m-1} \cdots h_1 \), where \( h_i = h_i^1 h_i^2 \cdots h_i^{\ell_i} \), by a decreasing Hecke biword

\[
\begin{bmatrix} k \\ h \end{bmatrix} = \begin{bmatrix} m & \ldots & m & \ldots & 1 & \ldots & 1 \\ h_m & \ldots & h_{m-1} & \ldots & h_1 & \ldots & h_1^{\ell_1} \end{bmatrix}.
\]

In addition, we say that \([k, h]^t\) is fully-commutative if \( h \) is fully-commutative.

Example 20. Consider the decreasing factorization \( h = (1)(2)(31)( ) (32) \). Then the corresponding biword \([k, h]^t\) is

\[
\begin{bmatrix} k \\ h \end{bmatrix} = \begin{bmatrix} 5 & 4 & 3 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 & 3 & 2 \end{bmatrix}.
\]

Definition 21. Starting with a decreasing Hecke biword \([k, h]^t\), we define Hecke row insertion from the right. The insertion sequence is read from right to left. Suppose there are \( n \) columns in \([k, h]^t\).
Start the insertion with \((P_0, Q_0)\) being both empty tableaux. We recursively construct \((P_{i+1}, Q_{i+1})\) from \((P_i, Q_i)\). Suppose the \((n - i)\)-th column in \([k, h]^t\) is \([y, x]^t\).

We describe how to insert \(x\) into \(P_i\), denoted \(P_i \leftarrow x\), by describing how to insert \(x\) into a row \(R\). The insertion may modify the row and may produce an output integer, which will be inserted into the next row. First, we insert \(x\) into the first row \(R\) of \(P_i\) following the rules below:

1. If \(x \geq z\) for all \(z \in R\), the insertion terminates in either of the following ways:
   
   (a) If we can append \(x\) to the right of \(R\) and obtain an increasing tableau, the result \(P_{i+1}\) is obtained by doing so; form \(Q_{i+1}\) by adding a box with \(y\) in the same position where \(x\) is added to \(P_i\).
   
   (b) Otherwise row \(R\) remains unchanged. Form \(Q_{i+1}\) by adding \(y\) to the existing corner of \(Q_i\) whose column contains the rightmost box of row \(R\).

2. Otherwise, there exists a smallest \(z\) in \(R\) such that \(z > x\).
   
   (a) If replacing \(z\) with \(x\) results in an increasing tableau, then do so. Let \(z\) be the output integer to be inserted into the next row.
   
   (b) Otherwise, row \(R\) remains unchanged. Let \(z\) be the output integer to be inserted into the next row.

The entire Hecke insertion terminates at \((P_n, Q_n)\) after we have inserted every letter from the Hecke biword. The resulting insertion tableau \(P_n\) is an increasing tableau, meaning that both rows and columns of \(P_n\) are strictly increasing. If \(k = (n, n - 1, \ldots, 1)\), the recording tableau \(Q_n\) is a standard set-valued tableau.

**Example 22.** Take \([k, h]^t\) from Example 20. Following the Hecke row insertion, we compute its insertion tableau and recording tableau:

\[
\emptyset \rightarrow \begin{array}{c}
2 \\
1 \ 3
\end{array} \rightarrow \begin{array}{c}
2 \\
1 \ 3
\end{array} \rightarrow \begin{array}{c}
2 \ 3 \\
1 \ 2
\end{array} \rightarrow \begin{array}{c}
3 \\
2 \ 3 \\
1 \ 2
\end{array} = P,
\]

\[
\emptyset \rightarrow \begin{array}{c}
1 \\
1 \ 1
\end{array} \rightarrow \begin{array}{c}
3 \\
1 \ 1
\end{array} \rightarrow \begin{array}{c}
3 \\
1 \ 1 \ 3
\end{array} \rightarrow \begin{array}{c}
3 \ 4 \\
1 \ 1 \ 3
\end{array} \rightarrow \begin{array}{c}
5 \\
3 \ 4 \\
1 \ 1 \ 3
\end{array} = Q.
\]

**Example 23.** Note that the recording tableau for the Hecke insertion of Definition 21 is not always a semistandard set-valued tableau. For example, for \(h = (21)(41)\) we have

\[
\begin{bmatrix}
k \\ h
\end{bmatrix} = \begin{bmatrix}
2 & 2 & 1 & 1 \\ 2 & 1 & 4 & 1
\end{bmatrix}
\]
and
\[ P = \begin{array}{c}
4 \\
1 & 2
\end{array} \quad \text{and} \quad Q = \begin{array}{c}
22 \\
1 & 1
\end{array}. \]

However, in Theorem 24 below we will see that in certain cases it is.

**Theorem 24.** Let \( T \in \text{SVT}(\lambda) \) and \([k, h]^t = \text{res}(T)\). Apply Hecke row insertion from the right on \([k, h]^t\) to obtain the pair of tableaux \((P, Q)\). Then \( Q = T \).

**Remark 25.** Combining Theorems 24 and 17 shows that Hecke insertion from right to left (as opposed to left to right in [PP16]) intertwines the crystal on set-valued tableaux and the \( \star \)-crystal, even though in general it is not always well-defined (see Example 23). This resolves [MPS18, Open Problem 5.8] when the decreasing factorizations are fully-commutative. Even when \( h \) is fully-commutative, but does not correspond to a straight-shaped tableau under \( \text{res}^{-1} \) as in Example 23, one can fill the skew part with small enough numbers and apply the Hecke insertion on this tableau. In the above example

\[
\begin{bmatrix}
k \\
h
\end{bmatrix} = \begin{bmatrix}
2 & 2 & 1 & 1 & 0 & 0 \\
2 & 1 & 4 & 1 & 3 & 2
\end{bmatrix} \quad \text{with} \quad Q = T = \begin{bmatrix}
12 & 2 & 0 \\
0 & 0 & 1
\end{bmatrix}.
\]

Note, however, that unlike in [MPS18] we use row Hecke insertion from right to left rather than column insertion from left to right (in analogy to [MS16] for Edelman–Greene insertion).

Since \( k \in T(i, j) \) if and only if \( \ell + j - i \in h^k \) under the residue map, where \( \ell = \ell(\lambda) \) and \( h^k \) is the \( k \)-th factor of \( h \), the statement of Theorem 24 is equivalent to applying Hecke insertion on the entries of \( T \) sorted first by ascending order of entries, followed by ascending diagonal content.

**Example 26.** Let \( T \) be the semistandard set-valued tableau

\[
T = \begin{array}{c}
2 & 4 \\
1 & 23
\end{array}.
\]

The insertion sequence by entry is listed in the table below:

<table>
<thead>
<tr>
<th>Cell</th>
<th>(1,1)</th>
<th>(2,1)</th>
<th>(1,2)</th>
<th>(1,2)</th>
<th>(2,2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content</td>
<td>2</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Entry</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

We will prove Theorem 24 by induction by considering all subtableaux of \( T \), obtained by adding the entries in \( T \) one by one in the order above:

\[
\emptyset \rightarrow \begin{array}{c}
1_2 \\
1_2
\end{array} \rightarrow \begin{array}{c}
2_1 \\
1_2 2_3
\end{array} \rightarrow \begin{array}{c}
2_1 \\
1_2 2_3 3_3
\end{array} \rightarrow \begin{array}{c}
2_1 4_2 \\
1_2 2_3 3_3
\end{array} = T.
\]
In addition, the corresponding sequence of insertion tableaux and recording tableaux is listed here:

\[
\emptyset \rightarrow \begin{array}{c|c}
2 \\
1 
\end{array} \rightarrow \begin{array}{c|c|c}
2 & 1 \\
1 & 3 
\end{array} \rightarrow \begin{array}{c|c|c|c}
2 & 1 & 3 \\
1 & 2 & 
\end{array} = P.
\]

\[
\emptyset \rightarrow \begin{array}{c|c}
1 \\
2 & 1 
\end{array} \rightarrow \begin{array}{c|c|c}
2 & 1 & 2 \\
2 & 1 & 3 
\end{array} \rightarrow \begin{array}{c|c|c|c}
2 & 4 & 1 & 23 \\
1 & 23 & 
\end{array} = Q.
\]

**Proof of Theorem 24.** We prove the theorem by proving the following more specific statement.

For a given step in the insertion process, suppose that the entries of \( T \) that are involved so far form a nonempty subtableau \( T' \) of \( T \) with shape \( \mu \) containing cell \((1,1)\), and the insertion tableau and recording tableau at the corresponding step are \( P(T') \) and \( Q(T') \). Then, they both have shape \( \mu \), and the entry of cell \((i,j)\) of \( P(T') \) is \( \ell + j - \mu'_j + i - 1 \), and \( Q(T') = T' \), where \( \mu' \) is the transpose of the partition \( \mu \) and \( \ell := \lambda_1' = \ell(\lambda) \).

We prove this by induction on subtableaux of \( T \).

**Base step:** Suppose \( T' \) only contains a single cell \((1,1)\) and \( T'(1,1) = S \), where \( S \) is a subset of \( T(1,1) \) with cardinality \( d \). Then \( P(T') \) is obtained by inserting \( d \) times the number \( \ell \). So we have \( P(T') = [\ell] \) and \( Q(T') = T' \). Here \( \mu = (1) \), so for \((i,j) = (1,1)\), we have \( \ell + j - \mu'_j + i - 1 = \ell \).

**Inductive step:** Suppose that the statements hold for some subtableau \( T' \) of shape \( \mu \). Assume the next insertion step involves adding the entry \( k \) in cell \((p,q)\) of \( T \) to \( T' \) to obtain \( T'' \). There are two cases: (1) the cell \((p,q)\) is already in \( T' \), or (2) the cell \((p,q)\) is not in \( T' \).

**Case (1):** We must have \((p,q)\) to be an inner corner of \( T' \) (no cell is to its right or above it), so \( p = \mu'_q \) and \( p > \mu'_q + 1 \). In this case, \( k \) is recorded in \( Q(T') \). Then by the induction on \( T' \), every cell \((i,j)\) of \( P(T') \) has value \( \ell + j - \mu'_j + i - 1 \). To determine the insertion path of \( P(T') \leftarrow \ell + q - p \), we compute the columns \( q \) and \( q + 1 \) of \( P(T') \) as follows:

<table>
<thead>
<tr>
<th>row number</th>
<th>( q )-th column</th>
<th>( (q + 1) )-st column</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p )</td>
<td>( \ell + q - 1 )</td>
<td></td>
</tr>
<tr>
<td>( \mu'_q + 1 &lt; p )</td>
<td>( \ell + q - p + \mu'_q + 1 )</td>
<td>( \ell + q )</td>
</tr>
<tr>
<td>( 2 )</td>
<td>( \ell + q - p + 1 )</td>
<td>( \ell + q + 2 - \mu'_q + 1 )</td>
</tr>
<tr>
<td>( 1 )</td>
<td>( \ell + q - p )</td>
<td>( \ell + q + 1 - \mu'_q + 1 )</td>
</tr>
</tbody>
</table>

Following Case 2(b) of Hecke insertion, the insertion path is vertically up column \( q + 1 \). At the top of the column, \( \ell + q \) is inserted into row \( \mu'_q + 1 \). Furthermore, \( \ell + q \) is greater than \( \ell + q - p + \mu'_q + 1 \) in cell \((\mu'_q + 1, q)\) because \( p > \mu'_q + 1 \). By Hecke insertion Case 1(b), the insertion ends in row \( \mu'_q + 1 \). Also \( P(T') \) is unchanged, and \( k \) is recorded in cell \((p,q)\) of \( Q(T') \) since it is the corner whose column contains the rightmost box of row.
In this case, we get $Q(T''') = T''$. Since the shape $\mu$ is unchanged, we have that $P(T''') = P(T')$ also satisfies the statement.

**Case (2):** If cell $(p, q)$ is not in $T'$, then it must be an outer corner of $T'$, so $\mu'_q = p - 1$ and $\mu'_{q-1} > p - 1$. Specifically, two cases can happen: (a) $p = 1$ and $(1, q - 1) \in T'$, (b) both $(p - 1, q), (p, q - 1) \in T'$, or $q = 1$ and $(p - 1, 1) \in T$.

**Case 2(a):** The first row of $P(T')$ is $\ell + 1 - \mu'_1, \ldots, \ell + j - \mu'_j, \ldots, \ell + (q - 1) - \mu'_{q-1}$. Since $\ell + q - p = \ell + q - 1 > \ell + (q - 1) - \mu'_{q-1}$, it is appended to the end of the first row which is the cell $(1, q)$. The letter $k$ is recorded in the same new cell of $Q(T')$. In this case, the only entry in $P$ that is changed is $(1, q)$, and its entry $\ell + q - 1$ satisfies the statement. Also $Q(T''')$ equals $T'''$.

**Case 2(b):** Since entry $(i, q - 1)$ of $P(T')$ is $\ell + q - 1 - \mu'_{q-1} + i - 1$ and entry $(i, q)$ of $P(T')$ is $\ell + q - \mu'_q + i - 1$, the number $q - p + \ell$ is in-between the two when $i = 1$. So the insertion starts by bumping $(1, q)$. To get the insertion path, we compute columns $q - 1$ and $q$ as follows:

<table>
<thead>
<tr>
<th>row number</th>
<th>$(q - 1)$-st column</th>
<th>$q$-th column</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mu'_{q-1}$</td>
<td>$\ell + q - 2$</td>
<td>...</td>
</tr>
<tr>
<td>$p - 1$</td>
<td>$\ell + q + p - \mu'_{q-1} - 3$</td>
<td>$\ell + q - 1$</td>
</tr>
<tr>
<td>2</td>
<td>$\ell + q - \mu'_{q-1}$</td>
<td>$\ell + q - p + 2$</td>
</tr>
<tr>
<td>1</td>
<td>$\ell + q - 1 - \mu'_{q-1}$</td>
<td>$\ell + q - p + 1$</td>
</tr>
</tbody>
</table>

By Hecke insertion Case 2(a), $\ell + q - p$ is placed in cell $(1, q)$ and the original column $q$ is shifted one position higher. By Hecke insertion Case 1(a), the insertion terminates at row $p$ and the original entry in cell $(p - 1, q)$ is appended at the rightmost box of row $p$. Thus, $\mu'_q$ increases by 1. The updated entries in column $q$ still satisfy the statement. Since the entries in other columns of $P(T')$ are unchanged and $\mu'_q$ is unchanged for $j \neq q$, they also satisfy the statement. So we have $P(T''')$ satisfies the statement. The letter $k$ is inserted into the new cell $(p, q)$ of $Q(T')$, which makes $Q(T''') = T'''$.

Thus, the statement holds, proving the theorem.

3.2 The $\star$-insertion

We define a new insertion algorithm, which we call $\star$-insertion, from fully-commutative decreasing Hecke biwords $[k, h]^t$ to pairs of tableaux $P$ and $Q$, denoted by $\star([k, h]^t) = (P, Q)$, as follows.

**Definition 27.** Fix a fully-commutative decreasing Hecke biword $[k, h]^t$. The insertion is done by reading the columns of this biword from right to left.

Begin with $(P_0, Q_0)$ being a pair of empty tableaux. For every integer $i \geq 0$, we recursively construct $(P_{i+1}, Q_{i+1})$ from $(P_i, Q_i)$ as follows. Let $[q, x]^t$ be the $i$-th column (from the right) of $[k, h]^t$. Suppose that we are inserting $x$ into row $R$ of $P_i$. 

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29 18
Case 1 If $R$ is empty or $x > \text{max}(R)$, then form $P_{i+1}$ by appending $x$ to row $R$ and form $Q_{i+1}$ by adding $q$ in the corresponding position to $Q_i$. Terminate and return $(P_{i+1}, Q_{i+1})$.

Case 2 Otherwise, if $x \notin R$, locate the smallest $y$ in $R$ with $y > x$. Bump $y$ with $x$ and insert $y$ into the next row of $P_i$.

Case 3 Otherwise, if $x \in R$, locate the smallest $y$ in $R$ with $y \leq x$ and interval $[y, x]$ contained in $R$. Row $R$ remains unchanged and $y$ is to be inserted into the next row of $P_i$.

Denote $(P, Q) = (P_\ell, Q_\ell)$ if $[k, h]^t$ has length $\ell$. We define the $\ast$-insertion by $\ast([k, h]^t) = (P, Q)$.

Furthermore, denote by $P \leftarrow x$ the tableau obtained by inserting $x$ into $P$. The collection of all cells in $P \leftarrow x$, where insertion or bumping has occurred is called the insertion path for $P \leftarrow x$. In particular, in Case 1 the newly added cell is in the insertion path, in Case 2 the cell containing the bumped letter $y$ is in the insertion path, and in Case 3 the cell containing the same entry as the inserted letter is in the insertion path.

Example 28. Let

$$\begin{pmatrix} k \\ h \end{pmatrix} = \begin{pmatrix} 4 & 4 & 2 & 2 & 1 & 1 \\ 4 & 2 & 4 & 2 & 3 & 1 \end{pmatrix}.$$ 

The corresponding sequence of insertion tableaux and recording tableaux under the $\ast$-insertion is listed here:

$$\emptyset \rightarrow 1 \rightarrow 1 3 \rightarrow 3 1 2 \rightarrow 3 1 2 4 \rightarrow 3 1 4 1 2 4 \rightarrow 3 1 4 1 2 4 = P.$$ 

$$\emptyset \rightarrow 1 \rightarrow 1 1 \rightarrow 2 1 1 \rightarrow 2 1 1 2 \rightarrow 4 2 1 1 2 \rightarrow 4 2 1 1 2 = Q.$$ 

Then we have $\ast([k, h]^t) = (P, Q)$, and the cells in the insertion paths at each step are highlighted in yellow.

Lemma 29. Let $[k, h]^t$ be a fully-commutative decreasing Hecke biword. Suppose that $\ast([k, h]^t) = (P, Q)$. Then, the following statements hold:

1. $P^k$ is semistandard and $Q$ has the same shape as $P$.

2. Let $x$ be an integer such that $x \cdot h$ is fully-commutative. Then the insertion path for $P \leftarrow x$ goes weakly to the left.
Proof. We will prove (1) by induction on the number of cells of \( P \). Statement (2) will follow by some results in the proof of statement (1).

Consider the leftmost column \([q, x]^t\) of \([k, h]^t\) and let \([k', h']^t\) be the Hecke biword formed by taking the remaining columns in the same order. If the \(*\)-insertion of \([k', h']^t\) yields \((P', Q')\), note that we have \( P = P' \leftarrow x \). For all integers \( j \geq 1 \), denote by \( R_j \) the (possibly empty) \(-j\)-th row of \( P' \). Denote by \( u \) the entry to be inserted into \( R_j \) and \( B_j \) as the cell in the insertion path at \( R_j \), where \( 1 \leq j \leq k \). Additionally, if bumping occurs at \( R_j \), denote the entry bumped out as \( y \).

(1) We will prove that if \((P')^t\) is semistandard, then the transpose of the updated tableau is semistandard.

**Case (a)** Suppose that the insertion terminates at \( R_1 \). Then Case 1 of the \(*\)-insertion has occurred, with a cell containing \( x \) appended at the end of the row. If \( R_1 \) is nonempty, then \( x > \max(R_1) \). Additionally, as \((P')^t\) is semistandard, integers strictly increase along \( R_1 \) but weakly increase along the column containing \( B_1 \). Hence, the transpose of the resulting tableau \( P \) is semistandard.

**Case (b)** Suppose that insertion terminates at \( R_k \), where \( k > 1 \). We will show that for all \( 1 \leq j \leq k \), the changes introduced at row \( R_j \) of \( P' \) maintain the property that the transpose of the updated tableau is semistandard.

**Case (b)(i)** Suppose that \( j = k \). In this case, a new cell containing \( u \) is appended at the end of \( R_k \) and \( u > \max(R_k) \) if the row is nonempty, proving that the integers increase strictly along \( R_k \).

If Case 2 occurs at \( R_{k-1} \), then \( u \) is the entry bumped out of \( R_{k-1} \) with the property that when \( u' \) is inserted into \( R_{k-1} \), \( u \in R_{k-1} \) is the smallest entry with \( u > u' \). Let \( z \) be the entry below cell \( B_k \). We claim that \( z \leq u \). If we assume instead that \( z > u \), then the cell containing \( z \) is strictly to the right of \( B_{k-1} \). However, the cell above \( B_{k-1} \) has value greater than \( u \) since \((P')^t\) is semistandard and \( u \notin R_k \). This contradicts the minimality of \( u' \), as \( u' \) is greater than this value, hence proving the claim.

If Case 3 occurs at \( R_{k-1} \), then \( u \) is bumped out of \( R_{k-1} \) with the property that when \( u' \) is inserted into \( R_{k-1} \), \( u \in R_{k-1} \) is the smallest entry with \( [u, u'] \subseteq R_{k-1} \). Let \( z \) be the entry below cell \( B_k \). Then, similar to the argument immediately before, \( z \leq u' \). Hence, we have established that the integers weakly increase along the column containing \( B_k \) after \( u \) is appended at the end of \( R_k \).

**Case (b)(ii)** Suppose that \( 1 \leq j < k \) and Case 2 occurs at \( R_j \). Then \( y \) is the entry bumped out of \( R_j \) with the property that when \( u \) is inserted into \( R_j \), \( y \in R_j \) is the smallest entry with \( y > u \). Thus, as \( u \notin R_j \), for all entries \( z \) and \( z' \) respectively to the left and to the right of \( B_j \), we have \( z < u < y < z' \).

If Case 2 occurs at \( R_{j-1} \), then \( u \) is bumped out of \( R_{j-1} \) with the property that when \( u' \) is inserted into \( R_{j-1} \), \( u \in R_{j-1} \) is the smallest entry with \( u > u' \). Let \( z \) be the
entry below cell \(B_j\). Then by repeating the same argument as in the first subcase of in Case (b)(i), we obtain \(z \leq u\).

If Case 3 occurs at \(R_{j-1}\), then \(u\) was bumped out of \(R_{j-1}\) with the property that when \(u'\) is inserted into \(R_{j-1}\), \(u \in R_{j-1}\) is the smallest entry with \([u, u'] \subseteq R_{j-1}\).

Let \(z\) be the entry below cell \(B_j\). Then by repeating the same argument as in the second subcase of in Case (b)(i), we obtain \(z \leq u'\).

Hence, we have established that integers increase weakly along the column containing \(B_j\) but increase strictly along \(R_j\) after \(u\) bumps out \(y\).

**Case (b)(iii)** Suppose that \(1 \leq j < k\) and Case 3 occurs at \(R_j\). In this case, there are no changes to row \(R_j\) after inserting \(u\) and bumping \(y\). Hence, it is trivial that integers increase weakly along the column containing \(B_j\) but increase strictly along \(R_j\) after \(u\) bumps out \(y\).

In all cases, we have shown that if \((P')^t\) is semistandard, then the transpose of the updated tableau remains semistandard. Therefore, by induction on the number of added cells, we have proved that the insertion tableau \(P\) under \(\star\)-insertion satisfies the property that \(P^t\) is semistandard.

Finally, note that the shape of the recording tableau is modified only when Case 1 of the \(\star\)-insertion has occurred. In this case, a cell is added to form \(Q\) at the same position as the cell added to form \(P\). Since we always begin with a pair of empty tableaux, by inducting on the number of added cells, the shapes of \(P\) and \(Q\) are the same.

(2) Suppose that the insertion terminates at \(R_k\), where \(k \geq 1\). We shall prove that \(B_j\) is weakly to the left of \(B_{j-1}\) for all \(1 < j \leq k\) by revisiting the cases explored in the proof of part (1) (note that \(P\) should replace the role of \(P'\)).

If Case 2 occurs at \(R_{j-1}\), then \(u\) is the entry bumped out of \(R_{j-1}\) with the property that when \(u'\) is inserted into \(R_{j-1}\), \(u \in R_{j-1}\) is the smallest entry with \(u > u'\). As in the proof of the first subcase of Case (b)(i) in part (1), we conclude that the entry \(z\) of the cell below \(B_k\) satisfies \(z \leq u\), showing that \(B_j\) is weakly to the left of \(B_{j-1}\).

If Case 3 occurs at \(R_{j-1}\), then \(u\) was bumped out of \(R_{j-1}\) with the property that when \(u'\) is inserted into \(R_{j-1}\), \(u \in R_{j-1}\) is the smallest entry with \([u, u'] \subseteq R_{j-1}\). As in the proof of the second subcase of Case (b)(i) in part (1), we conclude that the entry \(z\) of the cell below \(B_j\) satisfies \(z \leq u'\), \(B_j\) is weakly to the left of \(B_{j-1}\).

This completes the proof. \(\square\)

For the following results, given a tableau \(P\) with positive integer entries, \(\text{row}(P)\) denotes its row reading word, obtained by reading these entries row-by-row starting from the top row (in French notation), reading from left to right. We will consider \(\text{row}(P)\) as an element in a fixed 0-Hecke monoid.

**Lemma 30.** Let \(P\) be a tableau such that \(P^t\) is semistandard and \(\text{row}(P)\) is fully-commutative. Let \(x\) be an integer such that \(\text{row}(P) \cdot x\) is fully-commutative. Then,

\[
\text{row}(P \leftarrow x) \equiv_{H_0} \text{row}(P) \cdot x.
\]
Proof. To prove (5), let us first prove the following statements for all row tableaux $P$:

- With the assumptions in lemma, if insertion terminates at row $P$ while computing $P \leftarrow x$, then
  $$\text{row}(P \leftarrow x) \equiv_{H_0} \text{row}(P) \cdot x.$$ 

- With the assumptions in lemma, if $y$ is bumped from row $P$ and $P$ changes to $P'$ while computing $P \leftarrow x$, then
  $$\text{row}(P \leftarrow x) \equiv_{H_0} y \cdot \text{row}(P').$$

Assume that insertion terminates at row $P$ while computing $P \leftarrow x$. Then, Case 1 must have occurred and $P$ changes to $P'$, where $P'$ is $P$ appended by a cell containing $x$. Hence, we have

$$\text{row}(P \leftarrow x) \equiv_{H_0} \text{row}(P') \equiv_{H_0} \text{row}(P) \cdot x.$$ 

Assume that $y$ is bumped from row $P$ and $P$ changes to $P'$ while computing $P \leftarrow x$. Then, either Case 2 or Case 3 must have occurred.

If Case 2 occurs at $P$, then $x \not\in P$ and there is a $y \in P$ with $y > x$; furthermore, $y$ is the smallest value with such property. Write $P$ as $AyB$, where $A$ and $B$ are the row subtableaux of $P$ formed by entries to the left and to the right of $y$, respectively. Then, $P \leftarrow x$ is the tableau with row $Azb$ followed by row $y$. As $x \not\in P$, we have $\max(A) < x < y < \min(B)$. Hence by commutativity relations, for all $z \in B$, we have $z \cdot x \equiv_{H_0} x \cdot z$ and for all $z \in A$, we have $z \cdot y \equiv_{H_0} y \cdot z$, so that regarding $A$ and $B$ as words in $H_0(n)$, we obtain

$$A \cdot y \equiv_{H_0} y \cdot A, \quad B \cdot x \equiv_{H_0} x \cdot B.$$ 

It follows that

$$\text{row}(P) \cdot x \equiv_{H_0} \text{row}(AyB) \cdot x \equiv_{H_0} A \cdot y \cdot B \cdot x \equiv_{H_0} y \cdot A \cdot x \cdot B \equiv_{H_0} y \cdot \text{row}(AxB) \equiv_{H_0} \text{row}(P \leftarrow x).$$ 

If Case 3 occurs at $P$, then $x, y \in P$ with $y$ being the smallest value such that $[y, x] \subseteq P$. Write $P$ as $ABC$, where $B = [y, x]$, $A$ and $C$ are respectively the row subtableaux of $P$ formed by entries to the left and to the right of $B$. Then, $P \leftarrow x$ is the tableau with row $ABC$ followed by row $y$. As $\text{row}(P) \cdot x$ was assumed to be fully-commutative, $x + 1 \not\in P$. Furthermore, by minimality of $y$, $y > \max(A) + 1$. Hence, by commutativity relations, for all $z \in A$, we have $z \cdot y \equiv_{H_0} y \cdot z$ and for all $z \in C$, we have $x \cdot z \equiv_{H_0} z \cdot x$, so that

$$A \cdot y \equiv_{H_0} y \cdot A, \quad C \cdot x \equiv_{H_0} x \cdot C.$$ 

Moreover, by using the relations $p - 1 \equiv_{H_0} p - 1$, we have $y \cdot B \equiv_{H_0} B \cdot x$. It follows that

$$\text{row}(P) \cdot x \equiv_{H_0} \text{row}(ABC) \cdot x \equiv_{H_0} A \cdot B \cdot C \cdot x \equiv_{H_0} A \cdot y \cdot B \cdot C \equiv_{H_0} y \cdot \text{row}(ABC) \equiv_{H_0} \text{row}(P \leftarrow x).$$
Hence, the two statements above hold for all row tableaux $P$.

We are now ready to prove (5) in full generality. The result follows once we prove by induction on the number of rows of $P$, with the given setup above, that the following statements hold:

- If the insertion terminates within tableau $P$ while computing $P \leftarrow x$, then
  
  \[ \text{row}(P \leftarrow x) \equiv_{\mathcal{H}_0} \text{row}(P) \cdot x. \]

- If $y$ is bumped from tableau $P$ and $P$ changes to $P'$ while computing $P \leftarrow x$, then
  
  \[ \text{row}(P \leftarrow x) \equiv_{\mathcal{H}_0} y \cdot \text{row}(P'). \]

Indeed, if $P$ is a (possibly empty) row tableau, then we are done by the two previous statements that have been proved. Let $k \geq 1$ be an arbitrary integer. Assume that both statements mentioned above hold for all such tableaux $P$ with $k$ rows.

Let $P$ be a tableau with $k + 1$ rows with the setup as above. Then, we may consider the subtableau $P^*$ formed from its first $k$ rows and denote the final row as $R$. Note that $\text{row}(P) = \text{row}(R) \cdot \text{row}(P^*)$ and $\text{row}(R)$ is fully-commutative.

Assume that the changes from $P$ to $P \leftarrow x$ involve at most the first $k$ rows of $P$. Then $P \leftarrow x$ is the same tableau as $P^* \leftarrow x$ with an extra row $R$, so that by the inductive hypothesis,

\[ \text{row}(P \leftarrow x) \equiv_{\mathcal{H}_0} \text{row}(R) \cdot \text{row}(P^*) \quad \text{and} \quad \text{row}(P) \equiv_{\mathcal{H}_0} \text{row}(P \leftarrow x) \cdot x. \]

Now assume that the changes from $P$ to $P \leftarrow x$ involves all $k + 1$ rows of $P$. Let $P'$ be the resulting tableau after performing these changes on $P^*$ and let $y$ be the entry bumped from the final row of $P^*$. Then, $P \leftarrow x$ is the tableau obtained by concatenating tableau $R \leftarrow y$ after $P'$.

If the insertion terminates at row $R$, then by the previous statements for all row tableaux and the inductive hypothesis, we obtain

\[ \text{row}(P \leftarrow x) \equiv_{\mathcal{H}_0} \text{row}(R) \cdot \text{row}(P^* \leftarrow x) \equiv_{\mathcal{H}_0} \text{row}(R) \cdot \text{row}(P^*) \cdot x \equiv_{\mathcal{H}_0} \text{row}(P) \cdot x. \]

Otherwise, if the insertion bumps $z$ from $R$ and $R$ changes to $R'$ while computing $R \leftarrow y$, then it holds that the insertion bumps $z$ from $P$ while computing $P \leftarrow x$. In this case, if we denote $P''$ as the tableau $P'$ concatenated by row $R'$, then

\[ \text{row}(P) \cdot x \equiv_{\mathcal{H}_0} \text{row}(P \leftarrow y) \cdot \text{row}(P') \equiv_{\mathcal{H}_0} z \cdot \text{row}(R') \cdot \text{row}(P') \equiv_{\mathcal{H}_0} z \cdot \text{row}(P'') \equiv_{\mathcal{H}_0} \text{row}(P \leftarrow x). \]

This completes the induction. \qed
Remark 31. Observe that the assumption that \( \text{row}(P) \) is fully-commutative implies that \( \text{row}(R) \) is fully-commutative for each row \( R \) of \( P \). Moreover, in the proof of Lemma 30, if \( x \) is to be inserted into row \( R \) of \( P \) when computing \( P \leftarrow y \) and \( x \in R \), then the extra assumption that \( \text{row}(P) \cdot x \) is fully-commutative implies that \( R \) does not contain \( x + 1 \).

Lemma 32. Let \( P \) be a tableau such that \( P^t \) is semistandard and \( \text{row}(P) \) is fully-commutative. Let \( x, x' \) be integers such that \( \text{row}(P) \cdot x \) and \( \text{row}(P) \cdot xx' \) are fully-commutative.

Denote the insertion paths of \( P \leftarrow x \) and \( (P \leftarrow x) \leftarrow x' \) as \( \pi \) and \( \pi' \) respectively. Also, suppose that \( P \leftarrow x \) and \( (P \leftarrow x) \leftarrow x' \) introduce boxes \( B \) and \( B' \) respectively. Then the following statements about \( * \)-insertion are true:

1. If \( x < x' \), then \( \pi' \) is strictly to the right of \( \pi \). Moreover, \( B' \) is strictly to the right of and weakly below \( B \).

2. If \( x \geq x' \), then \( \pi' \) is weakly to the left of \( \pi \). Moreover, \( B' \) is weakly to the left of and strictly above \( B \).

Proof. Similar to Fulton’s proof [Ful96] of the Row Bumping Lemma, we will keep track of the entries as they are bumped from a row. Consider a row \( R \) of tableau \( P \) and suppose that \( u \) and \( u' \) are to be inserted into \( R \) when computing \( P \leftarrow x \) and \( (P \leftarrow x) \leftarrow x' \) respectively, where \( u < u' \). Denote by \( C \) (similarly \( C' \)) the box in \( \pi \) (similarly \( \pi' \)) that is also in \( R \).

Case 1: \( x < x' \). We will prove that the following assertions hold for \( R \):

(a) If the insertion terminates at \( R \) while computing \( P \leftarrow x \), then the insertion terminates at \( R \) while computing \( (P \leftarrow x) \leftarrow x' \).

(b) \( C' \) is strictly to the right of \( C \).

Note that the insertion terminates at \( R \) when computing \( P \leftarrow x \) precisely when Case 1 of the \(*\)-insertion occurs at \( R \). Box \( C \) containing \( u \) is appended at the end of \( R \). As \( u' > u \), Case 1 occurs again at \( R \) with box \( C' \) containing \( u' \) appended to the right of \( C \), so bumping does not occur at \( R \) when computing \( (P \leftarrow x) \leftarrow x' \). This proves (a) and simultaneously, (b) for this case.

Let us assume that bumping occurs at \( R \) with \( y \) bumped out when computing \( P \leftarrow x \).

Case A If \( y \) is bumped from \( R \) because Case 2 occurs, the insertion at row \( R \) introduced to box \( C' \) occurs strictly to the right of \( C \) (containing \( u \)) because:

(i) If \( u' > \max(R) \), then box \( C' \) containing \( u' \) is appended to the end of \( R \) by Case 1. In particular, \( C' \) appears strictly to the right of \( C \).

(ii) Otherwise, since \( u' > u \), the letter \( u' \) is inserted into a box \( C' \) strictly to the right of \( C \) with \( y \) bumped out. If \( u' \notin R \), Case 2 occurs and \( y' > y \) because \( C' \) and \( C \) originally contained \( y' \) and \( y \) respectively. Else, \( u' \in R \) and Case 3 occurs. Suppose that \( [y', u'] \) is the longest interval of consecutive integers contained in \( R \). Since box \( C \) that originally contained \( y \) is strictly to the left of \( C' \), we have \( u < y < u' \). Therefore, \( [u, u'] \) cannot be contained in \( R \), so \( y > y' \).
Case B Otherwise, $y$ is bumped from $R$ because Case 3 occurs when computing $P \leftarrow x$ and $[y, u]$ is the longest interval of consecutive integers contained in $R$ by Remark 31. The insertion at row $R$ introduced to box $C'$ occurs strictly to the right of $C$ (containing $u$) because:

(i) If either $u' > \text{max}(R)$ or $u' \notin R$, then by similar arguments as in Case A(i) and Case A(ii), $C'$ appears to the right of $C$. Furthermore, in the latter situation, by a similar argument in Case A(ii), we have $y < y'$.

(ii) Otherwise, $u' \in R$ and Case 3 occurs. As $u' > u$, $u'$ is inserted into box $C'$ strictly to the right of $C$ with $y'$ bumped out. In addition, $[y', u']$ is the longest interval of consecutive integers contained in $R$. As row $(R)$ is fully-commutative before computing $P \leftarrow x, u+1 \notin R$. Hence $[u, u']$ cannot be contained in $R$. It follows that $y \leq u < u' < y'$.

Note that in the arguments above, we have also shown that if $y$ and $y'$ are bumped from $R$ when computing $P \leftarrow x$ and $(P \leftarrow x) \leftarrow x'$ respectively, then $y < y'$. It follows that we may apply similar arguments in the rows following $R$. Since assertion (b) now holds for all rows, we conclude that $\pi'$ is strictly to the right of $\pi$. In addition, $\pi'$ cannot continue after $\pi$ ends because of assertion (a). Considering that $\pi'$ goes weakly left by Lemma 29, we conclude that box $B'$ is strictly to the right of and weakly below $B$.

Case 2: $x \geq x'$. We will prove that the following assertions hold for $R$:

1. If the insertion terminates at $R$ while computing $P \leftarrow x$, then bumping occurs at $R$ while computing $(P \leftarrow x) \leftarrow x'$.

2. $C'$ is weakly to the left of $C$.

If the insertion terminates at row $R$ when computing $P \leftarrow x$, then Case 1 occurs and box $C$ containing $u$ is appended at the end of $R$. If $u' \in R$, Case 3 occurs at $R$ with $y' \leq u' \leq u$ bumped out. Furthermore, box $C'$ containing $u'$ is weakly to the left of $C$. If $u' \notin R$, Case 2 occurs at $R$ with $y' > u'$ bumped out and $u' < u$. We have $y' \leq u$ by minimality of $y'$, so that box $C'$ is weakly to the left of $C$. In either of the subcases, bumping occurs at $R$ when computing $(P \leftarrow x) \leftarrow x'$. This proves (a) and simultaneously, (b) for this case.

Let us assume that bumping occurs at $R$ with $y$ bumped out when computing $P \leftarrow x$.

Case A If $y$ is bumped from $R$ because Case 2 occurs when computing $P \leftarrow x$, the insertion at row $R$ introduced to box $C'$ occurs weakly to the left of $C$ (containing $u$) because:

(i) If $u' \notin R$, then $u'$ is inserted into box $C'$ containing $y'$ by Case 2, while bumping out this $y'$. As $u' < u$, we have $y' \leq u < y$ and that $C'$ appears weakly to the left of $C$. 

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29 25
(ii) Otherwise, \( u' \in R \) and Case 3 occurs. The letter \( u' \) is inserted into box \( C' \) weakly to the left of \( C \) as \( u' \leq u \). In addition, if \([y', u']\) is the longest interval of consecutive integers in \( R \), then \( y' \) is bumped out. Furthermore, we have \( y' < y \) as \( C \), which originally contained \( y \) before computing \( P \leftarrow x \), is to the right of the box containing \( y' \).

**Case B** Otherwise, \( y \) is bumped from \( R \) because Case 3 occurs when computing \( P \leftarrow x \).

Let \([y, u]\) be the longest interval of consecutive integers that is contained in \( R \). The insertion at row \( R \) introduced to box \( C' \) occurs weakly to the left of \( C \) because:

(i) If \( u' \notin R \), then \( u' < u \), \( u' \) is inserted into box \( C' \) containing \( y' \) and \( y' \) is bumped out by Case 2. As \( \text{row}(P) \cdot x \) is fully-commutative, in particular \( \text{row}(R) \) is fully-commutative. Hence \( u' < y \), so that \( C'' \) is weakly to the left of box containing \( y \) (hence also weakly to the left of \( C \)). Furthermore, we have \( y' \leq y \) by the minimality of \( y' \).

(ii) If \( u' \in R \), then either \( u' = u \) or \( u < u' \). The former case is easy as Case 3 occurs again with \( u' \) inserted into \( C'' = C \) and \( y' = y \) is bumped out. If \( u < u' \), then as \( \text{row}(P) \cdot x \) is fully-commutative, \( \text{row}(R) \) is fully-commutative, so that \( u' < y - 1 \). It follows that \( C' \) is strictly to the left of box containing \( y \) (hence also strictly to the left of \( C \)). Furthermore, we have \( y' \leq u' < y - 1 < y \).

Note that in the arguments above, we have also shown that if \( y \) and \( y' \) are bumped from \( R \) when computing \( P \leftarrow x \) and \((P \leftarrow x) \leftarrow x'\) respectively, then \( y \geq y' \). It follows that we may apply similar arguments in the rows following \( R \). Since assertion (b) now holds for all rows, we conclude that \( \pi' \) is weakly to the left of \( \pi \). In addition, \( \pi' \) must continue after \( \pi \) ends because of assertion (a). Considering that \( \pi' \) goes weakly left by Lemma 29, we conclude that box \( B' \) is weakly to the left of and strictly above \( B \).

Let \( U \) be a tableau such that \( U' \) is semistandard and \( \text{row}(U) \) is fully-commutative. We describe the reverse row bumping for \( \ast \)-insertion of \( U \) as follows. Locate an inner corner of \( U \) and remove entry \( y \) from that row. Perform the following operations until an entry is bumped out of the bottommost row. Suppose that we are reverse bumping \( y \) into a row \( R \). If \( y \notin R \), find the largest \( x \in R \) with \( x < y \); insert \( y \) and bump out \( x \). Otherwise, \( y \in R \), so find the largest \( x \in R \) such that \([y, x]\) is the longest interval of consecutive integers. In this case, row \( R \) remains unchanged but \( x \) is bumped out. Then reverse bump \( x \) into the next row below unless there is no further row below. In this case, terminate and return the resulting tableau as \( T \) along with the bumped entry \( x \). It is straightforward to see that reverse row bumping specified above reverses the bumping process specified by the \( \ast \)-insertion.
**Example 33.** Let $U$ be the tableau

\[
U = \begin{array}{c}
5 \\
2 \\
2 & 5 \\
2 & 3 & 5 \\
1 & 2 & 4
\end{array}
\]

By performing reverse row bumping on the topmost 5 in $U$, we obtain

\[
T = \begin{array}{c}
5 \\
2 & 5 \\
2 & 3 & 5 \\
1 & 3 & 4
\end{array}
\]

and entry 2. It is also straightforward to check that $U = T \leftarrow 2$.

**Corollary 34.** Let $T$ be a tableau of shape $\lambda$ such that $T^t$ is semistandard and row$(T)$ is fully-commutative. Let $k$ be a positive integer.

Let $x_1 < x_2 < \cdots < x_k$ (similarly $x_k \leq \cdots \leq x_2 \leq x_1$) be integers such that row$(T) \cdot x_1 \cdots x_i$ is fully-commutative for all $1 \leq i \leq k$. Then, the collection of boxes added to $T$ to form the tableau

\[
U = ((T \leftarrow x_1) \leftarrow x_2) \cdots \leftarrow x_k
\]

has the property that no two boxes are in the same column (similarly row).

Conversely, if $U$ is a tableau of shape $\mu$ such that $\lambda \subseteq \mu$ and $\mu/\lambda$ consists of $k$ boxes with no two boxes in the same column, i.e., a horizontal strip of size $k$ (similarly row, i.e., a vertical strip of size $k$), then there is a unique tableau $T$ of shape $\lambda$ and unique integers $x_1 < x_2 < \cdots < x_k$ (similarly $x_k \leq \cdots \leq x_2 \leq x_1$) such that

\[
U = ((T \leftarrow x_1) \leftarrow x_2) \cdots \leftarrow x_k.
\]

In particular, if $(P,Q) = \ast([k,h]^t)$, where $[k,h]^t$ is a fully-commutative decreasing Hecke biword, then $Q$ is semistandard.

**Proof.** Assume that $x_1 < x_2 < \cdots < x_k$. By statement (1) of Lemma 32, the sequence of added boxes in $U = ((T \leftarrow x_1) \leftarrow x_2) \cdots \leftarrow x_k$ moves weakly below and strictly to the right when computing $U$. In particular, no two of the added boxes can be in the same column.

To recover the required tableau $T$ and integers $x_1 < x_2 < \cdots < x_k$, perform reverse row bumping on the boxes specified by the shape $\mu/\lambda$ within $U$ starting from the rightmost box, working from right to left. The tableau $T$ and the integers $x_1, x_2, \ldots, x_k$ are uniquely determined by the operations. Moreover, by Lemma 32, the integers $x_k, x_{k-1}, \ldots, x_1$ obtained in the given order of operations satisfy $x_1 < x_2 < \cdots < x_k$. 
Now assume $x_k \leq \cdots \leq x_2 \leq x_1$. By statement (2) of Lemma 32, the sequence of added boxes moves strictly above and weakly to the right when computing $U$. In particular, no two of the added boxes can be in the same row.

Similarly, one may perform reverse row bumping on the boxes specified by the shape $\mu/\lambda$ within $U$ starting from the topmost box, working from top to bottom. Again, the operations uniquely determine the tableau $T$ and the integers $x_1, x_2, \ldots, x_k$. Moreover, by Lemma 32, the integers $x_k, x_{k-1}, \ldots, x_1$ obtained in the given order of operations satisfy $x_k \leq \cdots \leq x_2 \leq x_1$.

Finally, note that in a decreasing Hecke biword $[k, h]^t$, where $h = h^m \ldots h^2 h^1$, entries within a fixed $a^t$ are inserted in increasing order. It follows that the collection of all boxes with label $i$ form a horizontal strip within the tableau $Q$. Collecting all these horizontal strips with values $i$ from $m$ to $i$ in order by using the converse recovers $Q$, implying that $Q$ is semistandard.

**Theorem 35.** The $\star$-insertion is a bijection from the set of all fully-commutative decreasing Hecke biwords to the set of all pairs of tableaux $(P, Q)$ of the same shape, where both $P^t$ and $Q$ are semistandard and $\text{row}(P)$ is fully-commutative.

**Proof.** By successive applications of Lemma 30, if $(P, Q) = \star([k, h]^t)$, then as $h$ is fully-commutative, $\text{row}(P)$ is also fully-commutative. Hence, using Lemma 29 and Corollary 34, $\star$-insertion is a well-defined map from the set of all fully-commutative decreasing Hecke biwords to the set of all pairs of tableaux $(P, Q)$ of the same shape with both $P^t, Q$ semistandard and $\text{row}(P)$ being fully-commutative.

It remains to show that the $\star$-insertion is an invertible map. Assume that $P$ and $Q$ are tableaux of the same shape with both $P^t, Q$ semistandard and $\text{row}(P)$ being fully-commutative. Since $Q$ is semistandard, the collection of boxes with the same entry form a horizontal strip. Starting with the largest such entry $m$, perform reverse row bumping with the boxes in the strip from right to left. By Lemma 32, this recovers the entries in $h^m$ in decreasing order. Repeating this procedure in decreasing order of entries recovers $h = h^m \ldots h^2 h^1$, which automatically yields a decreasing Hecke biword $[k, h]^t$. Furthermore, by repeated applications of Lemma 30, since $\text{row}(P)$ was fully-commutative, then the reverse word of $h$ is fully-commutative, so that $h$ is fully-commutative too. Finally, by repeated applications of the converse stated in Corollary 34, the recovered decreasing Hecke biword $[k, h]^t$ is unique. $\square$

4 Properties of the $\star$-insertion

In this section, we show that the $\star$-insertion intertwines with the crystal operators. More precisely, the insertion tableau remains invariant on connected crystal components under the $\star$-insertion as shown in Section 4.1 by introducing certain micro-moves. In Section 4.2, it is shown that the $\star$-crystal on $H^m: \star$ intertwines with the usual crystal operators on semistandard tableaux on the recording tableaux under the $\star$-insertion. In Section 4.3, we relate the $\star$-insertion to the uncrowding operation.
4.1 Micro-moves and invariance of the insertion tableaux

In this section, we introduce certain equivalence relations of the \(\star\)-insertion in order to establish its relation with the \(\star\)-crystal. From now on we are focusing on the sequence in the insertion order. Since each decreasing factorization \(h\) is inserted from right to left, we look at \(h\) read from right to left.

**Definition 36.** We define an equivalence relation through **micro-moves** on fully-commutative words in \(H_0(n)\).

1. **Knuth moves**, for \(x < z < y\):
   
   (I1) \(xyz \sim yxz\)
   
   (I2) \(zxy \sim zyx\)

2. **Weak Knuth moves**, for \(y > x + 1\):

   (II1) \(xyy \sim yxy\)
   
   (II2) \(xxy \sim xyx\)

3. **Hecke move**, for \(y = x + 1\):

   (III) \(xxy \sim yxy\)

Note that the micro-moves preserve the relation \(\equiv_{H_0}\).

Similar relations have appeared in [FG98, Eq. (1.2)].

**Example 37.** The 13242 \(\in H_0(5)\) is equivalent to 31242, 13422, 13224, 31224, and itself.

Next, we use the following notation on \(\star\)-insertion tableaux. For a single-row increasing tableau \(R\), let \(R^x\) denote the first row of the tableau \(R \leftarrow x\) and let \(R(x)\) denote the output of the \(\star\)-insertion from the first row. If the \(\star\)-insertion outputs a letter, then denote it by \(R(x)\); if \(x\) is appended to the end of the row \(R\), then the output \(R(x)\) is 0, which can be ignored. We always have \(x \cdot 0 \sim x \sim 0 \cdot x\).

**Example 38.** Let \(R = \begin{array}{ccccccc}1 & 3 & 4 & 6 & 7 & 8\end{array}\), then the first row of \(R \leftarrow 7\) is

\[
R^7 = \begin{array}{ccccccc}1 & 3 & 4 & 6 & 7 & 8\end{array}
\]

and \(R(7) = 6\). Furthermore, the first row of \(R^7 \leftarrow 9\) is \(R^{7,9} = \begin{array}{ccccccccc}1 & 3 & 4 & 6 & 7 & 8 & 9\end{array}\) and \(R^{8}(9) = 0\).

**Lemma 39.** Let \(R\) be a single-row increasing tableau, and \(x, y, z\) be letters such that \(\text{row}(R) \cdot x \cdot y \cdot z\) is fully-commutative. Let \(x', y', z'\) be letters such that \(xyz \sim x'y'z'\). Following the above notation, we have

\[
R^{x'y'z'} = R^{x'y'z'} \quad \text{and} \quad R(x)R^z(x')R^{y'y'(z')} \sim R(x')R^z(x')R^{y'y'}(z').
\]
Proof. Let \( R \) be a single-row increasing tableau and \( M \) be the largest letter in \( R \). First note that if \( a \in R \) and \( \text{row}(R) \cdot a \) is fully-commutative, then \( a + 1 \notin R \), see also Remark 31.

There are five types of equivalence triples, so we discuss them in 3 groups.

1. Cases (I1) and (II1): We have \( x < z < y \), or \( x < z = y \) and \( y > x + 1 \). In both cases \( x' = y, y' = x, z' = z \).

Case (1A): \( M < x < z \). In this case, the first resulting tableau is \( R^{xyz} = \begin{bmatrix} R \ x \ z \end{bmatrix} \) and the outputs are \( R(x) = R^x(y) = 0 \) and \( R^{xy}(z) = y \). The second resulting tableau is \( R^{xyz} = \begin{bmatrix} R \ x \ z \end{bmatrix} \) and the outputs are \( R(y) = 0 = R^{yx}(z) \) and \( R^y(x) = y \). So we have \( R^{xyz} = R^{xyz} \) and also \( 0 \cdot 0 \cdot y \sim 0 \cdot y \cdot 0 \).

Case (1B): \( x \leq M < z \leq y \). In this case, we have \( R^{xy} = R^{yx} \) and \( R(x) = R^y(x) \) since \( y \) is just appended to the end of \( R \). Thus, \( R(x) \cdot 0 \cdot y \sim 0 \cdot R(x) \cdot y \).

Case (1C): \( x < z \leq M < y \). In this case, we also have that \( R^{xy} = R^{yx} \) and \( R(x) = R^y(x) \), for the same reason as case (1B). Thus, we have \( R^{xyz} = R^{xyz} \) and \( R^{xy}(z) = R^{yx}(z) \). Since we have \( R^{y}(y) = R(y) = 0 \), \( R(x) \cdot 0 \cdot R^{yx}(z) \sim 0 \cdot R^{y}(x) \cdot R^{yx}(z) \).

Case (1D): \( x < z \leq y \leq M \). If \( x \) is the maximal letter in \( R^x \), then it follows as case (1B). Otherwise, this case needs further separation into subcases.

Case 1D-(i): \( x, y \notin R \). Then \( x < R(x), y < R(y) \) and \( R(x) \neq y \).

(1) If \( R(y) \leq y \), then \( R^x(y) = R(y) \) and \( R^y(x) = R(x) \), which implies \( R^{xy} = R^{yx} \), thus \( R^{xyz} = R^{xyz} \) and \( R^{xy}(z) = R^{yx}(z) \). Hence \( R(x)R^{x}(y)R^{xy}(z) = R^{y}(x)R(y)R^{yx}(z) \). Since \( R(x) < R^{xy}(z) \leq y < R(y) \), we have \( R(y) > R^{y}(x) + 1 \) and for the outputs \( R(x)R^{x}(y)R^{xy}(z) = R^{y}(x)R(y)R^{yx}(z) \sim R(y)R^{y}(x)R(y)R^{yx}(z) \) by move type (1I) or (I1). (2) If \( R(x) > y \), let the letter to the right of \( R(x) \) in \( R \) be \( R(x)^{\rightarrow} \). Then both \( R^{xyz} \) and \( R^{xyz} \) are obtained by replacing \( R(x) \) with \( x \) and \( R(x)^{\rightarrow} \) with \( z \). For the output, we have \( R(x) = R(y), R^{x}(y) > R(y), R^{y}(x) = y, R^{yx}(z) = R^{yx}(z) \) and \( R^{yx}(z) = y \). Since \( y < R(x) < R^{x}(y) \), we have that \( R^{x}(y) = R(x)^{\rightarrow} \sim y + 1 \). Hence the outputs \( R(x)R^{x}(y)R^{xy}(z) = R(x)R(x)^{\rightarrow}y \sim R(y)R^{y}(x)R^{yx}(z) \) by move of type (I2).

Case 1D-(ii): \( x \in R, y \notin R \). Then \( R(x) \leq x, R(y) > y \) and \( x + 1 \notin R \). In this case, we have \( R^{x}(y) = R(y) \) and \( R(y) = R(x) \), thus \( R^{xy} = R^{yx} \), \( R^{xy}(z) = R^{yx}(z) \) and \( R^{xyz} = R^{xyz} \). Since \( x + 1 \notin R \), we have \( R^{xy}(z) > x + 1 \). This implies \( R(x) \leq x < R^{xy}(z) \leq y < R(y) \), thus \( R(x)R^{x}(y)R^{xy}(z) \sim R(y)R^{y}(x)R^{yx}(z) \) as it is a type (I1) move.

Case 1D-(iii): \( x \notin R, y \in R \). Then \( x < R(x), y \geq R(y), y + 1 \notin R, R(x) - 1 \notin R, R(x) \leq y, R(y) \leq R^{x}(y) \) and \( R^{y} = R \).

(1) If \( R(x) = y \), denote the box to the right of \( y \) as \( y^{\rightarrow} \). Note that \( y^{\rightarrow} > y + 1 \). Then \( R^{x}(y) = y^{\rightarrow}, R^{xy}(z) = y, R^{y}(x) = y \) and \( R^{yx}(z) = y^{\rightarrow} \). Note \( y - 1 \notin R \), otherwise \( R(x) \leq y - 1 \). Thus, \( R(y) = y \). Both \( R^{xyz} \) and \( R^{xyz} \) are obtained by replacing \( y \in R \) with \( x \) and \( y^{\rightarrow} \) with \( z \), so \( R^{xyz} = R^{xyz} \). The outputs \( R(x)R^{x}(y)R^{xy}(z) = yy^{\rightarrow}y \sim yy^{\rightarrow}y = R(y)R^{y}(x)R^{yx}(z) \) as it is a type (II2) move.
(2) Suppose $R(x) < y$ and $R(x) = R(y)$. Then $[R(x), y] \subset R$ and $R^z(y) = R(x) + 1$. Since $R^y = R$ and $R^{xy} = R^x$, we have that both $R^{xy}$ and $R^{yz}$ equal $R^x$ and furthermore $R^y(x) = R(x)$. Note that $z$ can either be equal to $y$ or $z < R^y(y)$, otherwise $z \in R^{xy}$ and $z+1 \in R^{xy}$, which will give us a braid from row($R^{xy}$).z. Thus, we have $R^{xy}(z) = R^{yz}(z) = R(x) + 1$. In either case, the outputs are $R(x)R^y(y)R^{xy}(z) = R(x)(R(x)+1)(R(x)+1) \sim R(x)R(x)(R(x)+1) = R(y)R^y(x)R^{yz}(z)$ as they are type (III) moves.

(3) Suppose $R(x) < y$ and $R(x) < R(y)$. Then $R(y) > R(x) + 1$ and $R^x(y) = R(y)$. Similar to the previous case, both $R^{xy}$ and $R^{yz}$ are equal to $R^x$, and $z$ is either $y$ or $z < R(y)$. In either case, $R^{yz}(z) \leq R(y)$.

Then the outputs are $R(x)R^y(y)R^{xy}(z) = R(x)R(y)R^z(z) \sim R(y)R(x)R^z(z) = R(y)R^y(x)R^{yz}(z)$ as they are type (II) or (III) moves.

Case 1D-(iv): $x, y \in R$. In this case $x \geq R(x), y \geq R(y), x+1 \notin R$ and $y+1 \notin R$. Since $x+1 \notin R$, $[x, y]$ is not contained in $R$ and hence $R(y) > x + 1 > x \geq R(x)$.

Then $R^y(y) = R(y), R^y(x) = R(x)$ and $R^{xy} = R^{yz} = R$. Since $z > x$ and $x+1 \notin R$, we have $R(z) > x + 1 \geq R(x) + 1$. By similar reasons to the previous two subcases of Case 1D-(iii), $z$ can either be $y$ or $z < R(y)$ in order to avoid a braid in row($R^{xy}$).z. So, we have $R^{xy}(z) \leq R(y)$. Then the outputs are $R(x)R^y(y)R^{xy}(z) = R(x)R(y)R(z) \sim R(y)R(x)R(z) = R(y)R^y(x)R^{yz}(z)$ as they are type (II) moves.

2. Cases (I2) and (II2): We have $z < x < y$, or $z = x < y$ and $y > x + 1$. In both cases $x' = x, y' = z, z' = y$. By definition, $x \in R^x$.

Case 2A: $M < x < y$, then $R(x) = R^x(y) = 0$. $R^{xy} = R_x \setminus y$ is obtained by appending $x$ and $y$ to the end of $R$. Since $x \in R^x$ and $z \leq x < y$, we have $R^{xy}(z) = R^z(z)$. Moreover, $R^{yz}$ is obtained by appending $y$ to the end of $R^{xz}$ and hence $R^{xyz} = R^{xz}$. The outputs are $R(x)R^y(y)R^{xy}(z) = 00R^z(z) \sim 0R^y(z)0 = R(x)R^z(z)R^{xz}(y)$.

Case 2B: $z \leq x \leq M < y$, then $R^x(y) = R^{xz}(y) = 0$. Since $R^{xy} = R_x \setminus y$, $x \in R^x$ and $z \leq x$, we have $R^{xy}(z) = R^z(z)$, thus $R^{xyz} = R^{xy}y$. The output $R(x)R^y(y)R^{xy}(z) = R(x)0R^z(z) \sim R(x)R^z(z)0 = R(x)R^z(z)R^{xz}(y)$.

Case 2C: $z < x < y \leq M$, then we have $R^z(z) \leq x$. We discuss the following subcases.

Case 2C-i): $x, y \notin R$, then we have $R(x) > x$ and $R^x(y) > y$. Since $y > x$ and $x$ replaces $R(x)$ in $R$, we have $R^x(y) > R(x)$ from row strictness. Since $R^y(x) > R(x)$ and $R^x(z) \leq x$, we have $R^{xy}(z) = R^x(z)$ and $R^{xz}(y) = R^x(y)$. Furthermore, $R^{xyz} = R^{xy}$. Moreover, we have $R^x(y) \leq x < R^y(x) < R^x(y)$, which implies $R^x(y) > R^y(x) + 1$. Hence $R(x)R^y(y)R^{xy}(z) = R(x)R^y(y)R^{xy}(z) \sim R(x)R^z(z)R^{yz}(y) = R(x)R^z(z)R^{xz}(y)$ by type (II) moves.

Case 2C-ii): $x \in R, y \notin R$. Then $R^x = R, R(x) \leq x, R^x(y) > y$. Since $z \leq x$ and $[R(x), x] \subset R^x$, we have that $R^x(z) \leq R(x)$. Since $R^y(x) > y > x$ and $R^y(z) \leq R(x)$, we have that $R^{xy}(z) = R^x(z)$ and $R^{yz}(y) = R^x(y)$, thus $R^{xyz} = R^{xy}$. Since $R^z(z) \leq R(x) \leq x < y < R^x(y)$, we have $R^x(y) > R^x(z) + 1$. The outputs are $R(x)R^x(y)R^{xy}(z) = R(x)R^x(y)R^{xy}(z) \sim R(x)R^x(z)R^y(y) = R(x)R^z(z)R^{xz}(y)$ by type (II) or (II2) moves.

Case 2C-iii): $x \notin R, y \in R$. Then $R^{xy} = R^x$, $R(x) > x$ and $R^x(y) \leq y$. Let the
letter to the right of $R(x)$ in $R$ be $R(x)^\rightarrow$. Then $R(x)^\rightarrow > R(x) > x$ implies $R(x)^\rightarrow > x + 1$. This also shows that $x + 1 \notin R^x$ and thus $R^x(y) > R(x)$. Since $R^{xy} = R^x$ and $R^{xxy} = R^{xz}$, we have $R^{xyz} = R^{xz} = R^{xzy}$. Since $R^x(z) \leq x < R(x) < R^x(y)$, we have $R^x(y) > R^x(z) + 1$. Since $z \leq x$, we also have that $R^x(y) = R^{xz}(y)$. Thus, the outputs are $R(x)R^x(y)R^{xy}(z) = R(x)R^x(y)R^x(z) \sim R(x)R^x(z)R^{xg}(y)$ by a type (I2) move.

Case 2C-iv: $x \in R, y \in R$. Then $R^x = R, R^{xy} = R, R(x) \leq x, R^x(y) \leq y, x + 1 \notin R$ and $y + 1 \notin R$. Thus, $R^x(y) > x + 1$. Since $z \leq x$ and $[R(x), x] \subset R^x$, we have that $R^x(z) \leq R(x)$. Since $R^{xy} = R, R^{xzy} = R^x$. Since $R^x(z) \leq x, R^{xxy}(y) = R(y)$ and thus $R^{xzy} = R^x$. This implies $R^{xy} = R^{xzy}$. Now we have $R(z) \leq R(x) < x + 1 < R(y)$. Therefore, the outputs are $R(x)R^x(y)R^{xy}(z) = R(x)R^x(y)R(z) \sim R(x)R(z)R(y) = R(x)R^x(z)R^{xg}(y)$ by type (I2) or (II2) moves.

3. Case (III): We have $y = x, z = x + 1$ and hence $x' = x, y' = x + 1$ and $z' = x + 1$.

Case (3A): $x > M$. Then $R^x$ is obtained by appending $x$ to the end of $R$ and $R(x) = 0$. Also $R^{xx} = R^x$ with output $R^x(x)$. Note $R^{x, x+1}(x + 1) = R^x(x)$. Both $R^{xx, x+1}$ and $R^{xx, x+1, x+1}$ are obtained by appending $x + 1$ to the end of $R^x$, thus they are the same. The outputs are $R(x)R^x(x)R^{xx}(x + 1) = 0R^x(x)0 \sim 00R^x(x) = R(x)R^x(x + 1)R^x(x + 1)(x + 1)$.

Case (3B): $x \leq M, x + 1 > M$. Both $R^{xx, x+1}$ and $R^{xx, x+1, x+1}$ are obtained by appending $x + 1$ to the end of $R^x$, so they are equal. Since $x \in R^x$, we have $R^{xx, x+1}(x + 1) = R^x(x)$. Thus, the outputs are $R(x)R^x(x)R^{xx}(x + 1) = R(x)R^x(x)0 \sim R(x)0R^{xx, x+1}(x + 1)$.

Case (3C): $x + 1 \leq M$. It is clear that $x \in R^x$. If $x$ is the maximal letter in $R^x$, then the rest follows as case (3B).

Otherwise, let $x^\rightarrow$ be the letter to the right of $x$ in $R^x$. Since $x \in R^x$, we must have $x + 1 \notin R^x$, thus $x^\rightarrow > x + 1$. Moreover, we have $R^{xx} = R^x, R^x(x + 1) = R^{xx}(x + 1) = x^\rightarrow$. Since $R^{xx, x+1}$ is obtained from $R^x$ by replacing $x^\rightarrow$ with $x + 1$ and $x, x + 1 \in R^{xx, x+1}$, we have $R^{xx, x+1}(x + 1) = R^x(x)$. Both $R^{xx, x+1}$ and $R^{xx, x+1, x+1}$ are obtained from $R^x$ by replacing $x^\rightarrow$ with $x + 1$, thus they are the same. Furthermore, since $R^x(x) \leq x$ and $x^\rightarrow > x + 1$, we have that $R(x)R^x(x)R^{xx}(x + 1) = R(x)R^x(x)R^x(x) \sim R(x)R^x(x)R^x(x) = R(x)R^x(x + 1)R^{xx, x+1}(x + 1)$ by a type (I2) or (II2) move.

Proposition 40. If two words in $H_0(n)$ have the property that their reverse words are equivalent according to Definition 36, then they have the same insertion tableau under $\ast$-insertion (inserted from right to left).

Proof. Let $P$ be a $\ast$-insertion tableau. By Lemma 29, $P'$ is a semistandard tableau. Let the rows of $P$ be $R_1, \ldots, R_\ell$. Then each row is strictly increasing. The row $R_j$ is considered to be empty for $j > \ell$.

Let $x_1, y_1, z_1$ and $x'_1, y'_1, z'_1$ be letters such that $x_1y_1z_1 \sim x'_1y'_1z'_1$ and row$(P) \cdot x_1 \cdot y_1 \cdot z_1$ is fully-commutative. Let the output of the $\ast$-insertion algorithm of $P$ be $x_1 \leftarrow y_1 \rightarrow z_1$ (resp. $P \leftarrow x'_1 \leftarrow y'_1 \leftarrow z'_1$) from the row $i$ be $x_{i+1}, y_{i+1}, z_{i+1}$ (resp. $x'_{i+1}, y'_{i+1}, z'_{i+1}$). That is:

- $R_i^{x_1y_1z_1}$ is the first row of $[(R_i \leftarrow x_i) \leftarrow y_i] \leftarrow z_i$ and the outputs in order are $x_{i+1}, y_{i+1}, z_{i+1}$.
• $R_i^{x_i'y_i'z_i'}$ is the first row of $[(R_i \leftarrow x'_i) \leftarrow y'_i] \leftarrow z'_i$ and outputs in order are $x_{i+1}', y_{i+1}', z_{i+1}'$.

By Lemma 39, we have that $R_i^{x_i'y_i'z_i'} = R_i^{x_i'y_i'z_i'}$ and $x_{i+1}y_{i+1}z_{i+1} \sim x_{i+1}'y_{i+1}'z_{i+1}'$ for all $i$ (possibly some extra rows exceeding $\ell$). Thus, we have the desired result.

**Example 41.** The four words in $\mathcal{H}_0(5)$ of Example 37 all have the same $\ast$-insertion tableau:

$$
\begin{array}{cccc}
3 \\
1 \\
2 \\
4 \\
\end{array}
$$

In the next couple of lemmas, we prove that the crystal operators $f_k^\ast$ act by a composition of micro-moves as given in Definition 36. More precisely, for a fully-commutative decreasing factorization $h$, we have $h^{rev} \sim f_k^\ast(h)^{rev}$ as long as $f_k^\ast(h) \neq 0$, where $h^{rev}$ is the reverse of $h$.

**Remark 42.** By Definition 4 and Remark 8, there are two cases for the $k$-th and $(k+1)$-st factors under the crystal operator $f_k^\ast$, where $x$ is the largest unpaired letter in the $k$-th factor, $w_i, v_i \succ x$ and $u_i, b_i \prec x$:

1. $(w_1 \ldots w_p u_1 \ldots u_q)(v_1 \ldots v_s a b_1 \ldots b_t) f_k^\ast \rightarrow (w_1 \ldots w_p x u_1 \ldots u_q)(v_1 \ldots v_s b_1 \ldots b_t), \quad \text{where } v_s \neq x + 1.$

2. $(w_1 \ldots w_p u_1 \ldots u_q)(v_1 \ldots v_s a b_1 \ldots b_t) f_k^\ast \rightarrow (w_1 \ldots w_p x u_1 \ldots u_q)(v_1 \ldots v_s -1 x b_1 \ldots b_t), \quad \text{where } v_s = w_p = x + 1.$

In both cases, $u_i \prec x - 1$ since if $u_1 = x - 1$ then $b_1 = x - 1$ due to the fact that $x$ is unbracketed; but this would mean that the word is not fully-commutative. We also notice that since all $u_i$ are paired with some $b_j$, we have that $t \succ q$ and $b_i \succ u_i$. Similarly, all $v_i$ are paired with some $w_j$, so we have that $p \succ s$ and $v_i \succ w_{p-s+i}$. Let $u$ denote the sequence $u_1 \ldots u_q$ and let $b$ denote the sequence $b_1 \ldots b_t$.

**Lemma 43.**

1. For $2 \leq i \leq q, b_{i-1} > u_i + 1.$

2. For $1 \leq i < s, v_i > w_{p-s+i+1} + 1.$

**Proof.** (1): When $b_{i-1} > b_i + 1$ or $u_i \prec b_i$, the result follows directly.

Consider the case that $u_i = b_i = a$ and $b_{i-1} = b_i + 1 = a + 1$ for some letter $a$. Since $a = u_i < u_{i-1} \leq b_{i-1} = a + 1$, we must have $u_{i-1} = a + 1$. Let $c$ be the largest letter such that $[a, c] \subseteq b$. Then $c \geq a + 1$ and $c + 1 \notin b$. Moreover, since all $u_i$ are paired, $u_i \leq b_i$ and $u_{j-1} > u_j$, it is not hard to see that $[a, c] \subseteq u$ and $c, c - 1 \in u$. Since $c + 1 \notin b$, we can use commutativity to move $c \in b$ to the left and obtain a subword $c(c - 1)c$, which contradicts that the original word is fully-commutative.

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29

33
Lemma 46. We have that $b_q \ldots b_1 x v_s \ldots v_1 \sim u_q \ldots u_1$ is equivalent to $b_q u_q \ldots b_2 u_2 b_1 u_1 x v_s \ldots v_1$. 

(2): The proof is almost identical to the first part. When $w_{p-s+i} > w_{p-s+i+1} + 1$ or $v_i > w_{p-s+i}$, the result follows.

Consider the case $w_{p-s+i} = w_{p-s+i+1} + 1 = a + 1$ and $v_i = w_{p-s+i} = a + 1$ for some letter $a$. Since $a = w_{p-s+i+1} \leq v_{i+1} < v_i = a + 1$, we must that $v_{i+1} = a$. Let $c$ be the smallest letter such that $[c, a + 1] \subseteq w$. Then $c \leq a$ and $c - 1 \not\in w$. Moreover, since all $v_i$ are paired, $v_j \geq w_{p-s+j}$ and $v_{j+1} < v_j$, we can see that $[c, a + 1] \subseteq v$ and $c, c + 1 \in v$. Since $c - 1 \not\in w$, we can use commutativity to move $c \in w$ to the right and form a subword $c(c + 1)c$, which contradicts that the original word is fully-commutative. 

We now summarize several observations that will be used later.

Remark 44. For both types of actions of $f_k^*$ as in Remark 42, we have the following equivalence relations:

1. For $1 \leq i \leq q, 1 \leq j \leq s - 1$, $v_{j+1} v_j u_i \sim v_{j+1} u_i v_j$, since $u_i < v_{j+1} < v_j$.

2. For $1 \leq i \leq q$, $x v_s u_i \sim x u_i v_s$, since $u_i < x < v_s$.

3. For $1 \leq i \leq q$, $b_i x u_i \sim b_i u_i x$, since $u_i \leq u_i \leq b_i < x$, and $u_i < x - 1$.

4. For $1 \leq j < i - 1, 1 \leq i \leq q$, $b_{j+1} b_j u_i \sim b_{j+1} u_i b_j$, since $u_i \leq b_i < b_{j+1} < b_j$.

5. For $2 \leq i \leq q$, $b_i b_{i-1} u_i \sim b_i u_i b_{i-1}$, since $u_i \leq b_i < b_{i-1}$ and $b_{i-1} > u_i + 1$ by Lemma 43.

6. For $1 \leq i \leq s$, $p - s + i - 1 \leq j \leq p - 1$, $w_{j+1} v_i w_j \sim v_i w_{j+1} w_j$, since $w_{j+1} < w_j < w_{p-s+i} \leq v_i$.

7. For $1 \leq i \leq s - 1$, $w_{p-s+i+1} v_i w_{p-s+i} \sim v_i w_{p-s+i+1} w_{p-s+i}$, since $w_{p-s+i+1} < w_{p-s+i} \leq v_i$ and $v_i > w_{p-s+i+1} + 1$ by Lemma 43.

8. For all $1 \leq j \leq s - 1, 1 \leq i \leq q$, $v_{j+1} u_i v_j \sim v_{j+1} v_j u_i$, since $u_i < v_{j+1} < v_j$.

9. For $1 \leq i \leq q$, $b_1 u_i v_s \sim b_1 v_s u_i$, since $u_i < u_i \leq b_1 < v_s$.

10. For $1 \leq i \leq q$, $1 \leq j \leq s$, $x u_i v_j \sim x v_j u_i$, since $u_i < x < v_j$.

11. For $1 \leq j \leq s - 1$, $x v_j w_p \sim v_j x w_p$, since $x < w_p \leq v_s < v_j$.

Remark 45. When $v_s \neq x + 1$, we have the following equivalence relations:

1. $1 \leq i \leq s$, $x v_i x w_p \sim v_i x w_p$, since $x < w_p \leq v_s$ and $v_s > x + 1$.

2. $b_1 u_1 v_s \sim b_1 v_s u_1$, since $u_i \leq b_1 < v_s$ and $v_s > x + 1 > u_i + 1$.

Lemma 46. We have that $b_q \ldots b_1 x v_s \ldots v_1 u_q \ldots u_1$ is equivalent to $b_q u_q \ldots b_2 u_2 b_1 u_1 x v_s \ldots v_1$. 

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29 34
Proof. With the equivalence relations from Remark 44 (1)-(5), we can make the sequences of equivalence moves as follows:

\[ b_q \ldots b_1 x v_s \ldots v_2 v_1 u_q u_{q-1} \ldots u_1 \sim b_q \ldots b_1 x v_s \ldots v_2 u_q v_1 u_{q-1} \ldots u_1 \sim \]
\[ b_q \ldots b_1 x v_s u_q \ldots v_2 v_1 u_{q-1} \ldots u_1 \sim b_q \ldots b_1 x u_q v_1 \ldots v_2 v_1 u_{q-1} \ldots u_1 \sim \]
\[ b_q \ldots b_1 u_q x v_s \ldots v_2 v_1 u_{q-1} \ldots u_1 \sim b_q u_q \ldots b_1 x v_s \ldots v_2 v_1 u_{q-1} \ldots u_1 \sim \]
\[ b_q u_q b_{q-1} u_{q-1} \ldots b_1 u_1 x v_s \ldots v_2 v_1. \]

Lemma 47. We have that \( v_s \ldots v_1 w_p \ldots w_{p-1} \) is equivalent to \( v_s w_p v_s \ldots w_{p-1} \).

Proof. With the equivalence relations from Remark 44 (6)-(7), we can make the following equivalence moves:

\[ v_s \ldots v_2 v_1 w_p w_{p-1} \ldots w_{p-1} \sim v_s \ldots v_2 w_p w_{p-1} \ldots w_{p-1} \sim v_s \ldots v_3 w_p w_{p-1} \ldots w_{p-1} \sim v_s w_p \ldots v_1 w_{p-1}. \]

Lemma 48. We have

\[ x w_p v_{s-1} v_{p-1} \ldots v_2 w_{p-1} v_{p-1} \ldots w_{p-1} \sim v_{s-1} \ldots v_1 x w_p \ldots w_{p-1}. \]

Proof. With the equivalence relations from Remark 44 (6),(7) and (11), we can make the following equivalent moves:

\[ x w_p v_{s-1} v_{p-1} \ldots v_2 w_{p-1} v_{p-1} \ldots w_{p-1} \sim x v_{s-1} w_p w_{p-1} \ldots v_2 w_{p-1} v_{p-1} \ldots w_{p-1} \sim v_{s-1} x w_p w_{p-1} \ldots v_2 w_{p-1} v_{p-1} \ldots w_{p-1} \sim v_{s-1} w_p w_{p-1} \ldots v_2 w_{p-1} v_{p-1} \ldots w_{p-1}. \]

Lemma 49. When \( v_s \neq x + 1 \), we have

\[ x v_s w_p v_{s-1} v_{p-1} \ldots v_1 w_{p-1} \sim v_s \ldots v_1 x w_p \ldots w_{p-1}. \]

Proof. With the equivalence relations from Remark 44 (6)-(7) and Remark 45 (1), we can make the following equivalence moves:

\[ x v_s w_p v_{s-1} v_{p-1} v_{s-2} \ldots v_{1} w_{p-1} \sim v_s x w_p v_{s-1} v_{p-1} v_{s-2} \ldots v_{1} w_{p-1} \sim v_s v_{s-1} x w_p w_{p-1} v_{s-2} \ldots v_{1} w_{p-1} \sim v_s v_{s-1} v_{s-2} \ldots v_1 x w_p w_{p-1} \ldots w_{p-1}. \]

Lemma 50. When \( v_s \neq x + 1 \), we have \( b_q u_q \ldots b_1 u_1 v_s \ldots v_1 \) is equivalent to \( b_q \ldots b_1 v_s \ldots v_1 u_q \ldots u_1 \).

Proof. With the equivalence relations from Remark 44 (4), (5), (8)-(9) and Remark 45 (2), we can make the following equivalence moves:

\[ b_q u_q \ldots b_1 u_1 v_s \ldots v_1 \sim b_q u_q \ldots b_1 v_s u_1 \ldots v_1 \sim b_q u_q \ldots b_1 v_s \ldots v_1 u_1 \ldots u_1. \]
Lemma 51. We have \( b_quq \ldots b_1u_1xv_{s-1} \ldots v_1 \) is equivalent to \( b_q \ldots b_1xv_{s-1} \ldots v_1u_q \ldots u_1 \).

Proof. With the equivalence relations from Remark 44 (1), (3), (5) and (10) we have the following equivalence moves:

\[
\begin{align*}
\ & b_quq \ldots b_1u_1xv_{s-1} \ldots v_1 \sim b_quq \ldots b_1xu_1v_{s-1} \ldots v_1 \sim \\
\ & b_quq \ldots b_1xv_{s-1} \ldots v_1u_1 \sim b_q \ldots b_1xv_{s-1} \ldots v_1u_q \ldots u_1. 
\end{align*}
\]

Proposition 52. Suppose \( h \) is a fully-commutative decreasing factorization such that \( f^*_k(h) \neq 0 \) (resp. \( e^*_k(h) \neq 0 \)). Then \( f^*_k(h)^{rev} \sim h^{rev} \) (resp. \( e^*_k(h)^{rev} \sim h^{rev} \)) for the equivalence relation \( \sim \) of Definition 36.

Proof. We prove the statement for \( f^*_k \). Since \( e^*_k \) is a partial inverse of \( f^*_k \), the result follows.

Let \( h = h^m \ldots h^1 \in \mathcal{H}^{m,*} \) and define \( \tilde{h} = f^*_k(h) = h^m \ldots \tilde{h}^{k+1} \tilde{h}^{k-1} \ldots h^1 \). Specifically, \( \tilde{h}^{k+1} = (w_1 \ldots w_p u_1 \ldots u_q) \) and \( \tilde{h}^k = (v_1 \ldots v_s x v_1 \ldots v_t) \), where \( x \) is the largest unpaired letter in \( h^k \). Then by Lemmas 46 and 47, we have the following sequence of equivalence moves:

\[
\begin{align*}
\ & (b_q \ldots b_1xv_s \ldots v_1u_q \ldots u_1)w_p \ldots w_{p-s+1} \sim (b_q u_q \ldots b_1u_1xv_s \ldots v_1)w_p \ldots w_{p-s+1} \\
\ & b_qu_q \ldots b_1u_1x(v_s \ldots v_1w_p \ldots w_{p-s+1}) \sim b_qu_q \ldots b_1u_1(xw_p \ldots w_{p-s+1}). 
\end{align*}
\]

Case (1): When \( v_s \neq x + 1 \), \( \tilde{h}^{k+1} = (w_1 \ldots w_p xu_1 \ldots u_q) \), \( \tilde{h}^k = (v_1 \ldots v_s b_1 \ldots b_t) \). By Lemmas 49 and 50, we have

\[
\begin{align*}
\ & b_qu_q \ldots b_1u_1xv_s(w_p \ldots u_{s+1}) \sim b_qu_q \ldots b_1u_1(xv_s \ldots u_{s+1}) \\
\ & (b_qu_q \ldots b_1u_1v_1(xw_p \ldots w_s) \sim (b_qu_q \ldots b_1u_1v_1(xw_p \ldots w_{s-1}). 
\end{align*}
\]

Thus, we have that

\[
\begin{align*}
\ & b_t \ldots b_1xv_s \ldots v_1u_q \ldots u_1 \sim b_t \ldots b_1u_1xv_s \ldots v_1u_q \ldots u_1xw_p \ldots w_1. 
\end{align*}
\]

Case (2): When \( v_s = w_p = x+1 \), \( \tilde{h}^{k+1} = (w_1 \ldots w_p xu_1 \ldots u_q) \), \( \tilde{h}^k = (v_1 \ldots v_{s-1}xb_1 \ldots b_t) \). Then by Lemmas 48 and 51, we have

\[
\begin{align*}
\ & b_qu_q \ldots b_1u_1v_1xw_p \ldots w_{s-1}w_p \ldots v_1w \ldots w_{p-s+1} \sim b_qu_q \ldots b_1u_1v_1xw_p \ldots v_1w \ldots w_{p-s+1} \\
\ & b_qu_q \ldots b_1u_1x(w_p, \ldots v_1w \ldots w_{p-s+1}) \sim b_qu_q \ldots b_1u_1x(w_{s-1}, \ldots v_1w \ldots w_{p-s+1}) \\
\ & (b_qu_q \ldots b_1u_1xv_s-1 \ldots v_1xw_p \ldots w_{p-s+1} \sim (b_qu_q \ldots b_1u_1v_s-1 \ldots v_1v_s \ldots u_1xw_p \ldots w_{p-s+1}.
\end{align*}
\]

Thus, we have that

\[
\begin{align*}
\ & b_t \ldots b_1xv_s \ldots v_1u_q \ldots u_1w \ldots w_1 \sim b_t \ldots b_1xv_s \ldots v_1u_q \ldots u_1w \ldots w_1. 
\end{align*}
\]

Therefore, we have shown that in both cases, \( f^*_k(h)^{rev} \sim h^{rev} \).

Proposition 53. For \( h \in \mathcal{H}^{m,*} \) such that \( f^*_k(h) \neq 0 \) for some \( 1 \leq k < m \), the \( * \)-insertion tableau for \( h \) equals the \( * \)-insertion tableau for \( f^*_k(h) \).
Proof. By Proposition 52, the reverse words for $h$ and $f_k^*(h)$ are $\sim$-equivalent. By Proposition 40, the corresponding insertion tableaux are equal. \qed

**Proposition 54.** Let $h \in H^{m,*}$ be a lowest weight element under Definition 4 of weight $\lambda$. Then there exists $r \geq 1$ where $\lambda_i = 0$ for $i < r$ and $\lambda_{i+1} \geq \lambda_i$ for $1 \leq i \leq m$. Suppose $h = h^m \cdots h^r = (h_{\lambda_m}^m \cdots h_1^m)(h_{\lambda_m-1}^r \cdots h_1^r)(h_{\lambda_{r+1}} \cdots h_1^r)$, then the $i$-th row of the $\star$-insertion tableau equals $h_1^{r-i+1}, h_2^{r-i+1}, \ldots, h_{\lambda_{r+1}}^{r-i+1}$, that is, $P^*(h) = \begin{array}{cccc} h_1^r & \ldots & h_{\lambda_r}^r \\ \ldots & \ldots & \ldots & \ldots \\ h_1^{r-1} & h_2^{r-1} & \ldots & h_{\lambda_{r-1}}^{r-1} \\ h_1^{m-1} & h_2^{m-1} & \ldots & h_{\lambda_{m-1}}^{m-1} \\ h_1^m & h_2^m & \ldots & h_{\lambda_m}^m \end{array}. \quad (6)

Proof. Without loss of generality, we may assume that $r = 1$. We prove the statement by induction on $m$. The case $m = 1$ is trivial.

Let $m \geq 1$ be arbitrary and suppose that the statement holds for this $m$. We prove the statement for $m + 1$. We need to insert $P^*(h) \leftarrow h_1^{m+1} \leftarrow h_2^{m+1} \leftarrow \cdots \leftarrow h_{\lambda_{m+1}}^{m+1}$ where $P^*(h)$ is as in (6) with $r = 1$. Note that $h_i^{m+1} \leq h_i^m$ for $1 \leq i \leq \lambda_m$. Specifically, $h_1^{m+1} \leq h_1^m$, so its insertion path is vertical along the first column and we obtain $P^*(h) \leftarrow h_1^{m+1} = \begin{array}{cccc} h_1^1 \\ h_2^1 & \ldots & h_{\lambda_r}^r \\ \ldots & \ldots & \ldots & \ldots \\ h_1^{m-1} & h_2^{m-1} & \ldots & h_{\lambda_{m-1}}^{m-1} \\ h_1^m & h_2^m & \ldots & h_{\lambda_m}^m \\ h_1^{m+1} & h_2^m & \ldots & h_{\lambda_m}^m \end{array}.

Since $h_1^{m+1} < h_2^{m+1} \leq h_2^m$, the insertion path of $h_2^{m+1}$ is strictly to the right of the insertion path of $h_1^{m+1}$ and weakly left of the second column by Lemma 32, so it is vertical along the second column. Similar arguments show that the insertion path for $h_i^{m+1}$ is just vertical along the $i$-th column. Thus, the result holds for $m + 1$. \qed

**Remark 55.** For a lowest weight element $h \in H^{m,*}$ of weight $\mathbf{a}$, the corresponding insertion tableau must have shape $\mu = \text{sort}(\mathbf{a})$, which is the partition obtained by reordering $\mathbf{a}$.

**Proposition 56.** Let $T \in \text{SSYT}(\lambda)$ and $(P,Q) = \star \circ \text{res}(T)$. Then $Q = T$. 
Lemma 29. In other words, the following diagram commutes:

Note that we do not encounter Case (1) in the proof of Theorem 24. All other arguments still hold since for every insertion the letter is not contained in the row it is inserted into, that is, the insertion always bumps the smallest letter that is greater than itself. Thus, we omit the detail of the proof.

4.2 The $\star$-insertion and crystal operators

In this section, we prove that the $\star$-insertion and the crystal operators on fully-commutative decreasing factorizations and semistandard Young tableaux intertwine.

Theorem 57. Let $\mathbf{h} \in \mathcal{H}^{m,*}$. Let $(P^*(\mathbf{h}), Q^*(\mathbf{h})) = * (\mathbf{h})$ be the insertion and recording tableaux under the $\star$-insertion of Definition 27. Then

1. $f_i^*(\mathbf{h})$ is defined if and only if $f_i(Q^*(\mathbf{h}))$ is defined.
2. If $f_i^*(\mathbf{h})$ is defined, then $Q^*(f_i^*(\mathbf{h})) = f_i Q^*(\mathbf{h})$.

In other words, the following diagram commutes:

Proof. The crystal operator $f_i^*$ acts only on factors $h^{i+1}$ and $h^i$. Hence it suffices to prove the statement for $\mathbf{h} = h^{i+1} h^i \ldots h^1$ with $i + 1$ factors.

Suppose $f_i^*(\mathbf{h}) \neq 0$. By Proposition 53, $P^*(\mathbf{h}) = P^*(f_i^*(\mathbf{h}))$. Furthermore, by Lemma 29 $P^*(\mathbf{h})$ and $Q^*(\mathbf{h})$ have the same shape. Hence in particular, $Q^*(\mathbf{h})$ and $Q^*(f_i^*(\mathbf{h}))$ have the same shape and therefore the letters $i$ and $i+1$ in $Q^*(\mathbf{h})$ and $Q^*(f_i^*(\mathbf{h}))$ occupy the same skew shape.

Recall from Definition 4 that $f_i^*$ removes precisely one letter from factor $h^i = (h^i_k, h^i_{k-1} \ldots h^i_1)$, say $h^i_k$. By Lemma 32, the insertion paths of $h^i_1, \ldots, h^i_k$ into $P^*(h^{i-1} \ldots h^1)$ move strictly to the right and the newly added cells form a horizontal strip. In addition, the letters $h^i_1, \ldots, h^i_k$ appear in the first row of $P^*(h^i \ldots h^1)$. Now compare this to the insertion paths for $h^i_1, \ldots, h^i_k, \ldots, h^i_1$ into $P^*(h^{i-1} \ldots h^1)$, where $h^i_k$ is missing. Up to the insertion of $h^i_{k-1}$, everything agrees. Suppose that $h^i_k$ bumps the letter $x$ in the first row and $h^i_{k+1}$ bumps the letter $y > x$ in the first row by Lemma 32. Then when $h^i_{k+1}$ gets
inserted without prior insertion of $h_k$, the letter $h_{k+1}$ either still bumps $y$ or $h_{k+1}$ bumps $x$ (in which case $x$ and $y$ are adjacent in the first row in $P^*(h_{k-1} \cdots h_1)$). There are no other choices, since if there are letters between $x$ and $y$ in the first row and $h_{k+1}$ bumps one of these, it would have already bumped a letter to the left of $y$ in $P^*(h_{k+1} \cdots h_1)$. If $h_{k+1}$ bumps $x$ without prior insertion of $h_k$, then its insertion path is the same as the insertion path of $h_k$ previously. If $h_{k+1}$ bumps $y$, then the letter inserted into the second row by similar arguments either bumps the same letter as in the previous insertion path of $h_k$ or $h_{k+1}$. Repeating these arguments, exactly one cell containing $i$ in $Q^*(h_1 \cdots h_{i+1})$ is missing in $Q^*((h'_1 \cdots h'_k h_i h_{i+1}) h_i h_{i+1} \cdots h_1)$ and all other cells containing $i$ are the same. Hence, $Q^*(f_{i+1}^*(h))$ is obtained from $Q^*(h)$ by changing exactly one letter $i$ to $i + 1$.

It remains to prove that $f_i^*(h) \neq 0$ if and only if $f_i(Q^*(h)) \neq 0$ and, if $f_i^*(h) \neq 0$, then the letter $i$ that is changed to $i + 1$ from $Q^*(h)$ to $Q^*(f_{i+1}^*(h))$ is the rightmost unbracketed $i$ in $Q^*(h)$. First assume that under the bracketing rule for $f_i^*$, all letters in the factor $h^i$ are bracketed, so that $f_i^*(h) = 0$. This means that each letter in $h^i$ is paired with a weakly smaller letter in $h^{i+1}$. Then by similar arguments as in Lemma 32 (2), for each insertion path for the letters in $h^i$, there is an insertion path for the letters in $h^{i+1}$ that is weakly to the left and the resulting new cell is weakly to the left and strictly above of the corresponding new cell for the letter in $h^i$. This means that each $i$ in $Q^*(h)$ is paired with an $i+1$ and hence $f_i(Q^*(h)) = 0$.

Now assume that $f_i^*(h) \neq 0$. Let us use the same notation as in Remark 42 (with $k$ replaced by $i$). Since all letters $u_q, \ldots, u_1 < x$ are paired with some letters $b_j < x$, their insertion paths (again by similar arguments as in Lemma 32) lie strictly to the left of the insertion path for $x$. First assume that $v_s \neq x + 1$. Recall that by Proposition 53, $P^*(h) = P^*(f_i^*(h))$. Also, by the above arguments, moving letter $x$ to factor $h^{i+1}$ under $f_i^*$, changes one $i$ to $i + 1$ (precisely the $i$ that is missing when removing $x$ from $h^i$). Now the letters $w_p, \ldots, w_1 > x$ are inserted after the letter $x$ in the $(i+1)$-th factor in $f_i^*(h)$ and by Lemma 32 their insertion paths are strictly to the right of the insertion path of $x$ in $f_i^*(h)$. But this means that the corresponding $i + 1$ in $Q^*(h)$ cannot bracket with the $i$ that changes to $i + 1$ under $f_i^*$. This proves that $f_i(Q^*(h)) \neq 0$. Furthermore, each $v_p, \ldots, v_1$ is paired with some $w_j$ and hence the insertion path of this $w_j$ is weakly to the left of the left insertion path of the corresponding $v_h$. Hence all $i$ to the right of the $i$ that changes to an $i + 1$ under $f_i^*$ are bracketed. This proves that this $i$ is the rightmost unbracketed $i$, proving the claim. The case $v_s = x + 1$ is similar.

**Remark 58.** Proposition 54 and Theorem 57 provide another proof via $\ast$-insertion, in the case where $w$ is fully-commutative, of the Schur positivity of $\mathfrak{B}_w$ of Fomin and Greene [FG98]

$$\mathfrak{B}_w = \sum_{\mu} \beta_{|\mu| - \ell(w)}^w g^\mu_s \mu,$$

where $g^\mu_w = |\{T \in SSYT^n(\mu') \mid w_C(T) \equiv w\}|$. 

THE ELECTRONIC JOURNAL OF COMBINATORICS 27(2) (2020), #P2.29
4.3 Uncrowding set-valued skew tableaux

Buch [Buc02] introduced a bijection from a set-valued tableau of straight shape to a pair \((P, Q)\), where \(P\) is a semistandard tableau and \(Q\) is a flagged increasing tableau. The map involves the use of a dilation operation [BM12, RTY18] which can be defined equally to act on set-valued skew tableaux. Chan and Pflueger [CP19] recently studied the operation in this more general context. We review here the results needed for our purposes.

Let \(\lambda, \mu\) be partitions such that \(\lambda \subseteq \mu\) and \(\lambda_1 = \mu_1\). A flagged increasing tableau (introduced in [Len00] and called elegant fillings by various authors [Len00, LP07, BM12, Pat16]) is a row and column strict filling of the skew shape \(\mu/\lambda\) such that the positive integers entries in the \(i\)-th row of the tableau are at most \(i - 1\) for all \(1 \leq i \leq \ell(\mu)\). In particular, the bottom row is empty. Denote the set of all flagged increasing tableaux of shape \(\mu/\lambda\) by \(\mathcal{F}_{\mu/\lambda}\).

We use multicell to refer to a cell in a set-valued tableau with more than one letter.

**Definition 59.** For a skew shape \(\lambda/\mu\), the uncrowding operation is defined on \(T \in \text{SVT}(\lambda/\mu)\) as follows: identify the topmost row \(r\) in \(T\) containing a multicell. Let \(x\) be the largest letter in row \(r\) which lies in a multicell; delete this \(x\) and perform RSK row bumping with \(x\) into the rows above. The resulting tableau is the output of this operation. Note that its shape differs from \(\lambda/\mu\) by the addition of one cell.

The uncrowding map, denoted \(\text{uncrowd}\), is defined as follows. Let \(T \in \text{SVT}(\lambda/\mu)\) with \(\text{ex}(T) = \ell\).

- Start with \(\tilde{P}_0 = T\) and \(\tilde{Q}_0 = F\), where \(F\) is the unique flagged increasing tableau of shape \(\lambda/\lambda\).
- For each \(1 \leq i \leq \ell\), \(\tilde{P}_i\) is obtained from \(\tilde{P}_{i-1}\) by successively applying the uncrowding operation until no multicells remain. Each operation involves the addition of cell \(C\) to form \(\tilde{P}_i\) by first deleting an entry in cell \(B\) of \(\tilde{P}_{i-1}\); this is recorded by adding a cell with entry \(k\) to \(\tilde{Q}_{i-1}\) at the same position as \(C\), where \(k\) is the difference in the row indices of cells \(B\) and \(C\).
- Terminate and return \((\tilde{P}, \tilde{Q}) = (\tilde{P}_\ell, \tilde{Q}_\ell)\).

**Example 60.** Let \(T\) be the semistandard set-valued tableau

\[
T = \begin{array}{cccc}
5 \\
4 & 4 & 5 \\
2 & 2 & 3 \\
1 & 1 & 1 & 12 & 234 & 5 \\
\end{array}
\]
Perform an uncrowding operation to obtain

\[
T' = \begin{array}{cccc}
5 & 4 & 3 & 4 \\
2 & 2 & 3 & \\
1 & 1 & 1 & 12 \end{array} 234 5.
\]

Proceeding with uncrowding the remaining multicells and recording the changes, we have \(\text{uncrowd}(T) = (\widehat{P}, \widehat{Q})\), where

\[
\begin{array}{cccc}
5 & 5 & & \\
4 & 4 & & \\
3 & 3 & 4 & \\
2 & 2 & 2 & 3 \\
1 & 1 & 1 & 1 & 2 & 5 \\
\end{array}
\quad \text{and} \quad
\begin{array}{cccc}
3 & 4 & & \\
& & & \\
& & & \\
& & & \\
& & & \\
\end{array}
\]

Lemma 61. For skew shape \(\lambda/\mu\), the crystal operators on \(\text{SVT}^m(\lambda/\mu)\) intertwine with those on \(\text{SSYT}^m(\nu/\mu)\), for \(\lambda \subseteq \nu\), under \(\text{uncrowd}\).

Proof. Chan and Pflueger [CP19] proved that the image of \(T \in \text{SVT}(\lambda/\mu)\) under the uncrowding map is a pair \((P, Q)\), where \(P\) is a semistandard tableau of shape \(\nu/\mu\) and \(Q\) is a flagged increasing tableau of shape \(\nu/\lambda\). Monical, Pechenik and Scrimshaw in [MPS18, Theorem 3.12] proved that the crystal operators on \(\text{SVT}^m(\lambda)\) intertwine with those on \(\text{SSYT}^m(\nu)\) under \(\text{uncrowd}\). Since \(\text{uncrowd}\) is defined equally on skew shapes, the result follows. \(\square\)

4.4 Compatibility of \(\star\)-insertion with uncrowding

For a partition \(\mu\), let \(T_\mu\) be the unique tableau of shape \(\mu\) with \(\mu_i\) letters \(i\) in each row \(i\). Note that \(\text{uncrowd}(T_\mu) = (T_\mu, \emptyset)\) since \(\text{ex}(T_\mu) = 0\).

Lemma 62. For \(T \in \text{SVT}^m(\lambda/\mu)\), if \((P, Q) = \star(\text{hh}')\) where \(h = \text{res}(T)\) and \(h' = \text{res}(T_\mu)\), then \(T_\mu\) is contained in \(Q\).

Proof. For \(T \in \text{SVT}^m(\lambda/\mu)\), let \(T*\) be the set-valued tableau of shape \(\lambda\) obtained from \(T\) by adding \(\ell(\mu)\) to each entry and filling in the cells of \(\mu\) with \(T_\mu\). By Proposition 56, we have

\[
\star \circ \text{res}(T_\mu) = (P_\mu, T_\mu),
\]

(7) where \(P_\mu\) is the semistandard tableau specified in the proof of Proposition 56. The claim follows by noting that \(\text{res}(T*) = \text{res}(T)\text{res}(T_\mu)\). \(\square\)
Definition 63. A modification of $\star$-insertion is defined on $\mathcal{H}^{\star,m}$ as follows: for $h \in \mathcal{H}^{\star,m}$, let $\lambda/\mu$ be the shape of $\text{res}^{-1}(h)$ (which is well-defined up to a shift by Proposition 14). For $h' = \text{res}(T_{\mu})$, let $(P^\star,Q^\star) = \star(h'h)$. Define $\hat{\star}(h) = (P,Q)$ where $P$ is obtained from $P^\star$ by deleting all entries in cells of $\mu$ and $Q$ is defined from $Q^\star$ by deleting $T_{\mu}$ from it and decreasing all other letters by $\ell(\mu)$.

Note that this is well-defined by Lemma 62 and the fact that each $h \in \mathcal{H}^{\star,m}$ can be associated to a skew shape $\lambda/\mu$ which is the shape of $\text{res}^{-1}(h)$ by Proposition 14. Also note that $\hat{\star}(h) = \star(h)$ if $\mu = \emptyset$.

Theorem 64. Let $T \in \text{SVT}^m(\lambda/\mu)$, $(\hat{P},\hat{Q}) = \text{uncrowd}(T)$, and $(P,Q) = \hat{\star} \circ \text{res}(T)$. Then $Q = P$.

Proof. We start by addressing the straight-shape case; for $T^\star \in \text{SVT}^m(\lambda)$, consider the following compositions of maps:

\begin{equation*}
(\hat{P},\hat{Q}) \xleftarrow{\text{uncrowd}} T^\star \xrightarrow{\text{res}} h \xrightarrow{\star} (P,Q)
\end{equation*}

By Lemma 61, the left square commutes. By Theorem 17 the center square commutes. By Proposition 53 and Theorem 57 the right square commutes. Hence it suffices to prove that $Q = \hat{P}$ when $T^\star$ is a lowest weight element in the crystal.

Suppose $T^\star \in \text{SVT}^m(\lambda)$ is of lowest weight with $\text{wt}(T^\star) = a$ and $\text{ex}(T^\star) = \ell$. Then the decreasing factorization $h \in \mathcal{H}^{m,*}$ is lowest weight by Theorem 17. By Remark 55, $P$ and hence $Q$ has to be of shape $\nu = \text{sort}(a)$. By Theorem 57, $Q$ is the unique lowest weight element in $\text{SSYT}^m$ of shape $\nu$.

Consider the uncrowding operator on $T^\star$ and record each tableau during the process of uncrowding as in Definition 59 by a sequence of set-valued tableaux $T^\star = \hat{P}_0 \to \hat{P}_1 \to \cdots \to \hat{P}_\ell = \hat{P}$. Since $T^\star$ is of lowest weight, so are all the $\hat{P}_i$. Furthermore, all $\hat{P}_i$ have the same weight $a$. Let $(P_i,Q_i) = \star \circ \text{res}(\hat{P}_i)$ for all $0 \leq i \leq \ell$, $Q_i$ is the unique lowest weight element in $\text{SSYT}^m$ of shape $\nu$. Hence in particular $Q_i = Q$ for all $0 \leq i \leq \ell$. By Proposition 56, $Q = Q_\ell = \hat{P}$, proving the claim for straight shapes.

Now take $T \in \text{SVT}^m(\lambda/\mu)$ and construct $T^\star$ from $T$ by adding $\ell(\mu)$ to each entry and filling in the cells of $\mu$ with $T_{\mu}$. Note that $T^\star$ is a set-valued tableaux of shape $\lambda$. Let $(P,Q) = \star \circ \text{res}(T^\star)$ and $(P^\star,Q^\star) = \hat{\star} \circ \text{res}(T)$. Since $\text{res}(T^\star) = \text{res}(T)\text{res}(T_{\mu})$, Lemma 62 implies that $Q^\star = Q/T_{\mu}$. On the other hand, since $T^\star$ has straight shape, the preceding paragraph gives that $\text{uncrowd}(T^\star) = (Q,\hat{Q})$ for some $\hat{Q}$. We then note that $\text{uncrowd}(T)$ and $\text{uncrowd}(T^\star)$ are identical on cells of $\lambda/\mu$ up to a shift of the entries by $\ell(\mu)$; in particular, applying $\text{uncrowd}$ to $T^\star$ does not involve any cell of $\mu$ since none of these are multicells and their entries are the smallest $\ell(\mu)$ letters. \qed
5 Results on the non-fully-commutative case

In this section, we discuss some aspects when we generalize to the non-fully-commutative case. In Section 5.1, we describe a local crystal on $\mathcal{H}^m(3)$. In Section 5.2, we show that under very mild assumptions it is not possible to expect a local crystal for $n > 3$.

5.1 The case $n = 3$

We provide a description of a type $A_{m-1}$ crystal structure on $\mathcal{H}^m(3)$.

**Definition 65.** Let $h = h^m h^{m-1} \ldots h^2 h^1 \in \mathcal{H}^m(3)$. Fix $1 \leq k < m$. Define the *pairing process* of $h$ and the number of pairs in $h^{k-1} \ldots h^j h^j$, denoted $p([j, k-1])$, recursively as follows:

1. The empty factorization, denoted $\emptyset$, has no pairs and $p(\emptyset) = 0$.
2. If $p([1, j-1])$ is defined for all $1 \leq j \leq k$, then we have $p([j, k-1]) = p([1, k-1]) - p([1, j-1])$.
3. If $h^k = ( )$, then set $p([1, k]) = p([1, k-1])$.
4. Otherwise, if $h^k = (21)$, pair the 2 with the 1 in $h^k$ and set $p([1, k]) = p([1, k-1]) + 1$.
5. Otherwise, if $h^k = (2)$ and $p([1, k-1])$ is even, ignoring all previously paired letters, locate the leftmost unpaired letter in $h^{k-1} \ldots h^2 h^1$.
   (a) If this letter is in $h^j = (1)$ and $p([j+1, k-1])$ is even, then pair the 2 in $h^k$ with the 1 in $h^j$ and set $p([1, k]) = p([1, k-1]) + 1$.
   (b) If this letter is in $h^j = (2)$ and $p([j+1, k-1])$ is odd, then pair the 2 in $h^k$ with the 2 in $h^j$ and set $p([1, k]) = p([1, k-1]) + 1$.
   (c) Else, set $p([1, k]) = p([1, k-1])$.
6. Otherwise, if $h^k = (1)$ and $p([1, k-1])$ is odd, ignoring all previously paired letters, locate the leftmost unpaired letter in $h^{k-1} \ldots h^2 h^1$.
   (a) If this letter is in $h^j = (2)$ and $p([j+1, k-1])$ is even, then pair the 1 in $h^k$ with the 2 in $h^j$ and set $p([1, k]) = p([1, k-1]) + 1$.
   (b) If this letter is in $h^j = (1)$ and $p([j+1, k-1])$ is odd, then pair the 1 in $h^k$ with the 1 in $h^j$ and set $p([1, k]) = p([1, k-1]) + 1$.
   (c) Else, set $p([1, k]) = p([1, k-1])$.
7. Else, set $p([1, k]) = p([1, k-1])$.

**Example 66.** Let $m = 8$ and consider $h = ( ) (2) ( ) (21) (1) (1) (2) (21) \in \mathcal{H}^8(3)$. The pairing process results in $h = ( ) (2) ( ) (21) (1) (1) (2) (21)$, where the paired letters are indicated...
with braces. Hence, we have the following values of $p([1,k])$ for $1 \leq k \leq 8$: 0, 1, 1, 2, 2, 3, 3, 3. Note that the letters in the fourth and seventh factors are left unpaired.

Similarly, if we take $h = ( ) (2)(21)(2)(1)(21)(21) \in \mathcal{H}^8(3)$, we obtain


Thus, we have the following values of $p([1,k])$ for $1 \leq k \leq 8$: 0, 1, 2, 2, 2, 3, 4, 5. In this case all the letters in $h$ are paired.

**Definition 67.** Let $h = h^m \ldots h^2 h^1 \in \mathcal{H}^m(3)$. The crystal operator $f_i$ for $1 \leq i < m$ on $h$ is defined as follows. The operator $f_i$ only depends on $h^{i+1} h^i$ and the parity of $p([1, i - 1])$ of Definition 65. In the following cases, we indicate only the changes in $h^{i+1} h^i$ under $f_i$ as the remainder of $h$ remains invariant:

1. $(21)(x) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2), (21) \}$,
2. $(x)( ) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2), (21) \}$,
3. $(x)(x) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2) \}$,
4. $(1)(21) \xrightarrow{i} (21)(2)$,
5. $(2)(21) \xrightarrow{i} (21)(1)$,
6. $( ) (x) \xrightarrow{i} (x)( )$, where $(x) \in \{ (1), (2) \}$,
7. $( ) (21) \xrightarrow{i} (2)(1) \xrightarrow{i} (21)( )$, if $p([1, i - 1])$ is even,
8. $( ) (21) \xrightarrow{i} (1)(2) \xrightarrow{i} (21)( )$, if $p([1, i - 1])$ is odd.

The operator $e_i$ is defined similarly. One reverses the changes introduced in cases (4) to (8) and annihilates $h$ when the following occurs at $h^{i+1} h^i$:

1. $^\prime (x)(21) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2), (21) \}$,
2. $^\prime (x)(x) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2), (21) \}$,
3. $^\prime (x)(x) \xrightarrow{i} 0$, where $(x) \in \{ ( ), (1), (2) \}$.

Similar to Definition 4, the weight map is defined as

$$\text{wt}(h) = (\text{len}(h^1), \text{len}(h^2), \ldots, \text{len}(h^m)).$$

Meanwhile, $\varphi_i(h)$ (resp. $\varepsilon_i(h)$) is defined to be the largest nonnegative integer $k$ such that $f^k_i(h) \neq 0$ (resp. $e^k_i(h) \neq 0$).
Figure 3: The crystal graph for $\mathcal{H}^3(3)$ restricted to decreasing factorizations with four letters.

It is not difficult to check that the operators $f_i$ and $e_i$ defined above preserve the relation $\equiv h_0$ on $\mathcal{H}^m(3)$ whenever they do not annihilate the decreasing factorizations. Furthermore, the structure above defines an abstract, seminormal $A_{m-1}$ crystal on $\mathcal{H}^m(3)$.

We note that one may also verify that the crystal is a Stembridge crystal by checking that the axioms formulated in [Ste03] are satisfied. Figure 3 displays the crystal graph on $\mathcal{H}^3(3)$ restricted to decreasing factorizations that use exactly 4 letters.

5.2 Nonlocality

In this subsection, we show that it is impossible to construct a crystal on $\mathcal{H}^m$ with the following properties for $f_i$:

1. $f_i$ only changes the $i$-th and $(i+1)$-th decreasing factors;
2. $f_i$ is determined by the first $(i+1)$ factors;
3. $f_i(h) \equiv h_0$ and $\text{ex}[f_i(h)] = \text{ex}(h)$, for all $h \in \mathcal{H}^m$ with $f_i(h) \neq 0$.

Let $h_1 = h_1^m \ldots h_1^2 h_1^1 \in \mathcal{H}^m$ and suppose that $f_i(h_1) \neq 0$. If we write $f_i(h_1) = h_1^2 \ldots h_2^2 h_1^1$, then the above assumptions imply that $h_1^2 \ldots h_1^1 \equiv h_0 h_2^2 \ldots h_1^1$. Obviously the crystal on $\mathcal{H}^m(3)$ defined in Section 5.1 satisfies these assumptions.

Suppose that a crystal structure with the above assumptions exists on $\mathcal{H}^4(4)$. Consider the Schur expansion of the stable Grothendieck polynomial in 4 variables for $w = 12132$:

$$G_{12132}(x_1, x_2, x_3, x_4; \beta) = s_{221} + \beta (2s_{222} + 3s_{2211}) + \beta^2 (6s_{2221} + 6s_{22111}) + \cdots.$$
Figure 4: Partial filling of the connected component of $\mathcal{H}^4(3)$ containing highest weight element $(21)(32)(32)$.

(Note that $s_{22111}$ is zero in four variables and hence could be omitted). The linear term in $\beta$ implies that there are two connected components with highest weight $(2,2,2,0)$ (lowest weight $(0,2,2,2)$) for the crystal $\mathcal{H}^4(4)$ with excess 1. All decreasing factorizations mentioned below are those of $w=12132$ with 4 factors and excess 1.

There are two decreasing factorizations of weight $(2,2,2,0)$: $(21)(21)(32)$ and $(21)(32)(32)$. Focus on the connected component with highest weight $(21)(32)(32)$ and try to complete the crystal graph from top to bottom. Since the only decreasing factorization of weight $(2,2,1,1)$ with the first and second factors both being $(32)$ is $(21)(32)(32)$, we can compute the action of $f_3$ on this highest weight element. By some similar arguments we can fill in part of the crystal graph as indicated in Figure 4 with the above assumptions. The dashed spaces are undetermined.

Yet note that the red $f_2$ highlighted in the graph changed the first factor from $(3)$ to $(2)$. Hence, Condition (1) is violated, providing a counterexample that crystals with the above conditions always exist on $\mathcal{H}^m(n)$ for $n > 3$.

Acknowledgments

We are grateful to Travis Scrimshaw and Zach Hamaker for discussions. Jennifer Morse and Anne Schilling would like to thank MRSI for hospitality during their stay in July 2019, where part of this research was carried out.

This material is based upon work supported by the National Security Agency under Grant No. H98230-19-1-0119, The Lyda Hill Foundation, The McGovern Foundation, and
Microsoft Research, while JM and AS were in residence at the Mathematical Sciences Research Institute in Berkeley, California, during the summer of 2019. The first author was partially supported by NSF grant DMS–1833333. The last author was partially supported by NSF grants DMS–1760329 and DMS–1764153.

References


