Characterization of $\mathcal{B}(\infty)$ using marginally large tableaux and rigged configurations in the $A_n$ case via integer sequences

Roger Tian

December 16, 2016

Abstract

Rigged configurations are combinatorial objects prominent in the study of solvable lattice models. Marginally large tableaux are semi-standard Young tableaux of special form that give a certain realization of the crystals $\mathcal{B}(\infty)$. We introduce cascading sequences to characterize marginally large tableaux. Then we use cascading sequences and a known non-explicit crystal isomorphism between marginally large tableaux and rigged configurations to give a characterization of the latter set, and to give an explicit bijection between the two sets.

Contents

1 Introduction 1
2 Marginally Large Tableaux and Cascading Sequences 3
  2.1 Marginally Large Tableaux . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3
  2.2 Cascading Sequences and a Bijection . . . . . . . . . . . . . . . . . . . . . . . . . 4
  2.3 Lanes of Cascading Sequences . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5
3 Cascading Sequences and Rigged Configurations 8
  3.1 Rigged Configurations . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8
  3.2 Kashiwara Operators Acting in a Cascading Sequence Arrangement . . . . . . 9
  3.3 Obtaining the Rigged Configuration from the Cascading Sequence . . . . . . . 12
  3.4 Rough Idea of the Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16
  3.5 Plateaus as Base for Construction . . . . . . . . . . . . . . . . . . . . . . . . . . 17
  3.6 Adding Boxes to a Stretch . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
  3.7 Proof of the Procedures for Adding Boxes . . . . . . . . . . . . . . . . . . . . . . 20
  3.8 Growth Algorithm . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
  3.9 Determining the Cascading Sequence of a Rigged Configuration . . . . . . . . 32
4 Further Discussions 36

1 Introduction

Kashiwara introduced the crystal $\mathcal{B}(\infty)$, which is the crystal base of the negative part $U_q^{-}(\mathfrak{g})$ of a quantum group, in [5] and used it to study the Demazure crystals that were conjectured by
Littelmann [6]. As $B(\infty)$ reveals much about the structure of the quantum group $U_q(g)$ itself, it is an active topic of research. By the work of Hong and Lee [7], $B(\infty)$ can be realized as crystals consisting of combinatorial objects called marginally large tableaux, which are a special class of semi-standard Young tableaux.

Schilling [12] gave an explicit $U_q(g)$-crystal structure to combinatorial objects called rigged configurations, which naturally serve as indexes for the eigenvalues and eigenvectors of the Hamiltonian in the Bethe Ansatz. A crystal model for $B(\infty)$ in terms of rigged configurations was given [10] by Salisbury and Scrimshaw for affine simply-laced types, who also established [11] an isomorphism between rigged configurations and marginally large tableaux as crystals. However, this isomorphism is not explicit, and the $B(\infty)$ rigged configurations have not yet been explicitly characterized at the writing of [4].

The purpose of this paper is to characterize the rigged configurations of the $A_n$ type in $B(\infty)$ and to give an explicit bijection between marginally large tableaux and $B(\infty)$ rigged configurations of the $A_n$ type. We will achieve this by introducing special integer sequences that will be called cascading sequences. Any element of a highest weight crystal is obtained by acting on the highest weight vector via a sequence of Kashiwara operators, though this sequence is not necessarily unique. A cascading sequence can be viewed as the “canonical” sequence of Kashiwara operators leading to any crystal from the highest weight crystal. The desired bijection will be obtained by first establishing a bijection between the marginally large tableaux and the cascading sequences, and then establishing a bijection between the cascading sequences and the rigged configurations.

This paper is organized as follows. In Subsection 2.1, we recall the definition of marginally large tableaux. In Subsection 2.2, we introduce cascading sequences and use them to characterize marginally large tableaux. In Subsection 2.3, we introduce an aspect of cascading sequences called lanes that will later be used in the characterization of rigged configurations. In Subsection 3.1, we recall the definition of $B(\infty)$ rigged configurations in the $A_n$ case. In Subsection 3.2, we show in Lemma 3.2.4 that Kashiwara operators for rigged configurations act nicely when arranged in a cascading sequence, which allows us to obtain an interesting structural property Theorem 3.2.7 of rigged configurations. In Subsection 3.3, we show that lanes of a cascading sequence correspond to columns of rigged partitions in the corresponding rigged configuration, and we obtain the first half of the characterization of rigged configurations Theorem 3.3.11. In Subsection 3.4, we give the rough idea of our growth algorithm for characterizing rigged configurations. In Subsection 3.5, we introduce special cascading sequences called $p$-plateaus that will be used in the growth algorithm. In Subsection 3.6, we show how to modify a cascading sequence to achieve the effect of adding boxes to a rigged partition in the corresponding rigged configuration. In Subsection 3.8, we give our growth algorithm Theorem 3.8.3 for characterizing rigged configurations. In Subsection 3.9, we give in Theorem 3.9.1 an algorithm for obtaining the cascading sequence of any rigged configuration.

Acknowledgments

The author would like to thank Anne Schilling for helpful discussions.

The author would like to thank Jin Hong, Hyenomi Lee, and Travis Scrimshaw for reviewing parts of this paper, which began as part of a joint project with Hyenomi Lee. However, Jin Hong and Hyenomi Lee posted a paper on the same topic to the arXiv in April, without giving due credit to RT and to strong protest from RT, even though HL had received considerable documented mathematical input from RT. This resulted in a joint paper [4] (version 2) between JH, HL and RT; some form of the current writing was part of this joint paper. However, JH and HL decided unilaterally to terminate this joint project recently. Lastly, JH and HL left comments
on their arXiv posting suggesting that they terminated the joint project due to serious errors and logical gaps that they uncovered in RT’s paper, even though they were unable to present an example of such a serious issue when confronted on this matter by RT. This compels RT to include this description of the dispute.

The author used Sage ([8] and [9]) extensively to do computations with marginally large tableaux and rigged configurations that would have been forbidding by hand.

The author was partially supported by NSF grants DMS-1001256, OCI-1147247, DMS-1500050.

2 Marginally Large Tableaux and Cascading Sequences

We give a bijection between the set of \( A_n \) marginally large tableaux and a special set of integer sequences that we call cascading \( n \)-sequences.

2.1 Marginally Large Tableaux

In this subsection, we recall the definition of marginally large tableaux for type \( A_n \), as given in [3].

Definition 2.1.1. We call a semi-standard tableau \( T \) large if it has exactly \( n \) nonempty row, and the \( i \)th row has strictly more \( i \)-boxes than the total number of boxes in the \((i + 1)\)st row, for each \( 1 \leq i < n \).

Definition 2.1.2. By a marginally large tableau in the \( A_n \) case we will mean a Young tableau with exactly \( n \) rows whose entries come from the alphabet \( J = \{1 < 2 < \cdots < n < n + 1\} \) that satisfies the following conditions:

1. The \( i \)th row of the leftmost column is a single \( i \)-box, for each \( 1 \leq i \leq n \).
2. Entries increase weakly as we go from left to right along each row.
3. The number of \( i \)-boxes in the \( i \)th row exceeds by exactly one the total number of boxes in the \((i + 1)\)st row, for each \( 1 \leq i < n \).

Let \( T(\infty) \) denote the set of \( A_n \) marginally large tableaux. As shown in [3], \( T(\infty) \) has a crystal structure, given as follows.

Procedure 2.1.3. We describe how to apply the Kashiwara operator \( f_i \) to any marginally large tableau:

1. Apply \( f_i \) to this tableau in the usual way, by writing the tableau as a tensor product, applying the tensor product rule, and assembling the result back into tableau form.
2. We are done if the result we obtain is a large tableau, as it will be marginally large automatically.
3. If the result we obtain is not a large tableau, then \( f_i \) must have acted on the rightmost \( i \)-box of the \( i \)th row. Insert a single column of height \( i \) to the left of this box that \( f_i \) acted upon. For \( 1 \leq k \leq i \), the \( k \)th row of the added column must be a \( k \)-box.

Example 2.1.4. Given the marginally large tableau

\[
S = \begin{array}{c}
1 & 1 & 1 & 1 \\
2 & 2 & 2 \\
3 & 4 \\
\end{array}
\]
we have
\[
f_2(S) = \begin{bmatrix}
1 & 1 & 1 & 1 \\
2 & 2 & 2 & 3 \\
3 & 4
\end{bmatrix}.
\]

Procedure 2.1.5. We describe how to apply the Kashiwara operator \(e_i\) to any marginally large tableau:

1. Apply \(e_i\) to this tableau in the usual way.

2. We are done if the result we obtain is zero or a marginally large tableau.

3. Otherwise, the result is a large tableau that is not marginally large. \(e_i\) must have acted on the box to the right of the rightmost \(i\)-box of the \(i\)th row. Remove the column that contains this changed box. This column will have height \(i\), and its \(k\)th row consists of a single \(k\)-box, for \(1 \leq k \leq i\).

Example 2.1.6. Given the marginally large tableau
\[
S = \begin{bmatrix}
1 & 1 & 1 & 1 \\
2 & 2 & 2 \\
3 & 4
\end{bmatrix},
\]
we have
\[
e_3(S) = \begin{bmatrix}
1 & 1 & 1 \\
2 & 2 \\
3
\end{bmatrix}.
\]

2.2 Cascading Sequences and a Bijection

For any \(m \in [n] = \{1, 2, \ldots, n\}\), we will call any subinterval \([a, m] = \{a, a + 1, \ldots, m\}\) of \([n]\) an \(m\)-lower subinterval. For example, \([3, 5]\) is a 5-lower subinterval of \([6]\). By an \(m\)-component, we will mean a sequence of finitely many (allowed to be zero) \(m\)-lower subintervals of \([n]\) ordered by nonincreasing length.

Definition 2.2.1. By a cascading \(n\)-sequence we will mean an integer sequence formed by concatenating an \(n\)-component, an \((n - 1)\)-component, an \((n - 2)\)-component, \ldots, in that order. Let \(\tilde{A}_n\) denote the set of cascading \(n\)-sequences.

Example 2.2.2. \((1, 2, 3, 4, 5, 3, 4, 5, 3, 4, 5, 3, 2, 2, 2, 1)\) is an element of \(\tilde{A}_5\) where the lower subintervals (written as tuples) are \((1, 2, 3, 4, 5), (3, 4, 5), (3, 4, 5), (5), (2, 3, 4), (3, 4), (2, 3), (3), (2), (2), (1)\).

We will follow the English notation for the Young tableau, with weakly increasing row length as we move up the tableau. Let \(M^A_n\) denote the set of marginally large tableaux (MLT) in the \(A_n\) case. We now define a map \(\Phi : M^A_n \rightarrow \tilde{A}_n\) which will be shown to be a bijection. Given a marginally large tableau \(T\), we will give an \(f\)-string (sequence of Kashiwara operators \(f_1, f_2, \ldots, f_n\); also called Lusztig data [7]) with nice properties that gives rise to \(T\) upon acting on the highest weight MLT. We will write this \(f\)-string as its corresponding sequence of indices, and we will see that this sequence is an element of \(\tilde{A}_n\). Let \(T(i)\) denote the portion of the \(i\)th row of \(T\) without the \(i\) boxes.

Define \(\Phi(T)\) as follows. The \(f\)-string that we will give will add the \((n + 1)\)-boxes, the \(n\)-boxes, the \((n - 1)\)-boxes, and so on up that order. Let \(x_{i,j}\) denote the number of \((j + 1)\)-boxes in the \(i\)th row of \(T\). The \(n\)-component of \(\Phi(T)\) consists of \(x_{i,n}\) copies of \((i, i + 1, \ldots, n)\) for \(i = 1, 2, \ldots\). In general, the \(m\)-component of \(\Phi(T)\) consists of \(x_{i,m}\) copies of \((i, i + 1, \ldots, m)\) for \(i = 1, 2, \ldots\); each copy of \((i, i + 1, \ldots, m)\) adds an \((m + 1)\)-box to the \(i\)th row.
Remark 2.2.3. The cascading sequences can actually be rewritten as Berenstein-Zelevinsky-Lusztig data (in [1], [2]) for the reduced word 

\[(s_1)(s_2s_1)\ldots(s_n\ldots s_2s_1)\].

However, to the best of our knowledge, the use of such sequences for the purpose of characterization given in this paper is new.

Example 2.2.4. Given 

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

in \(M^{A_5}\), we have 

\[\Phi(T) = (1, 2, 3, 4, 5, 3, 4, 5, 3, 4, 5, 3, 4, 5, 3, 4, 5, 3, 1, 2, 1, 2, 1, 2, 1, 1, 1)\]

where the lower subintervals \(1, 2, 3, 4, 5\), \(3, 4, 5\), \(3, 4, 5\), \(5\), \(5\), \(5\) add all the 6-boxes of \(T\), the lower subintervals \(3, 4\), \(3, 4\), \(4\) add all the 5-boxes of \(T\), the lower subintervals \(3\), \(3\) add all the 4-boxes of \(T\), the lower subintervals \(1, 2\), \(1, 2\), \(1, 2\) add all the 3-boxes of \(T\), and the lower subintervals \(1\), \(1\), \(1\) add all the 2-boxes of \(T\).

Proposition 2.2.5. The map \(\Phi : M^{A_n} \to \bar{A}_n\) defined above is a bijection.

Remark 2.2.6. Thus, we can take the cascading \(n\)-sequences to be “canonical” \(f\)-strings for \(M^{A_n}\).

Proof. The inverse map \(\Phi^{-1}\) can be described as follows. Given an \(f\)-string \(\alpha \in \bar{A}_n\), we can read off all its lower subintervals in left-right order. Each such lower subinterval \([i, m]\) gives an \((m + 1)\)-box in the \(i\)th row of the MLT resulting from \(\alpha\) acting on the highest weight element. Thus, each such lower subinterval \([i, m]\) specifies that there must be an \((m + 1)\)-box in the \(i\)th row of \(\Phi^{-1}(\alpha)\). In this way, the MLT \(\Phi^{-1}(\alpha)\) is completely determined, since \(\Phi^{-1}(\alpha)(i)\) is completely determined for each row \(i\).

Remark 2.2.7. Notice that the elements \(\alpha \in \bar{A}_n\) are particularly convenient as \(f\)-strings for MLT’s, as we can obtain the corresponding MLT \(\Phi^{-1}(\alpha)\) (which is the same MLT obtained by having \(\alpha\) act on the highest weight element) by simply reading off the lower subintervals of \(\alpha\), without having to apply the Kashiwara operators on the highest weight element. For instance, we see in Example 2.2.4 that we can immediately obtain \(T\) from the \(f\)-string by noting that \(T\) has exactly one 6-box in the first row specified by the lower subinterval \(1, 2, 3, 4, 5\), exactly two 5-boxes in the third row specified by the lower subintervals \(3, 4\), \(3, 4\), and so on.

Finally, we mention that the cascading sequence characterization in this section can also be applied to regular Young tableaux, with slight modification.

2.3 Lanes of Cascading Sequences

As already shown in [10], the marginally large tableaux are isomorphic to the rigged configurations as crystals, so we can use cascading sequences to characterize the latter objects (which are in bijection with cascading sequences), which have not yet been characterized explicitly.
Given two tuples $u = (u_1, \ldots, u_i), v = (v_1, \ldots, v_j)$ we define

$$u \oplus v = (u_1, \ldots, u_i, v_1, \ldots, v_j).$$

If $u, v$ are lower subintervals, we define their intersection $u \cap v$ in the natural way. For example, we have $(3, 5, 2) \cap (5) = (3, 5, 2, 5)$ and we have $(7, 8, 9) \cap (6, 7, 8) = (7, 8).

We first introduce the aspects of cascading $n$-sequences that will be useful in describing $A_n$ rigged configurations. Let $\alpha \in \mathcal{A}_n$ be a cascading $n$-sequence.

For the remainder of this subsection, we partition $\alpha$ into subsequences that we will call lanes. As subsequences, lanes will be written as tuples. For any tuple, the first entry will be called the head of the tuple and the last entry will be called the tail of the tuple. Also, for any lane $L$ of $\alpha$, let $|L|$ denote the length of $L$. Formation of these lanes will reflect the way Kashiwara operators act on rigged configurations in Lemma 3.2.4. Furthermore, we will show that the $i$th partition in the corresponding rigged configuration.

Label the lower subintervals of $\alpha$ as $I_1, I_2, \ldots, I_k$ from left to right. Denote by $I_i(j)$ the $j$th entry of $I_i$, and by $\mathcal{T}_i(j)$ the integer value (in $[n]$) of $I_i(j)$; in Example 2.2.2, $I_5 = (2, 3, 4)$ and $\mathcal{T}_5(3) = 4$. Lanes will be formed, via the following iterative procedure, for each integer in $\alpha$; i.e. for $m \in [n]$ there will be lanes $L_1(m), L_2(m), \ldots$ at the end of the procedure. The lane forming procedure builds the lanes in stages, as follows:

At the outset, we form lanes using entries of $I_1$, by setting $L_1(\mathcal{T}_1(j)) := (I_1(j))$ for each $j$.

In general, suppose a collection of lanes $M^1, M^2, \ldots, M^a$ has been formed from the lower subintervals $I_1, I_2, \ldots, I_{b-1}$. Set $L_p(q) := \emptyset$ for any $L_p(q) \notin \{M^1, M^2, \ldots, M^a\}$. We will form new lanes using entries of $I_b$. First, pick the maximal $d_1$ such that $L_{d_1}(\mathcal{T}_b(1)) \in \{M^1, M^2, \ldots, M^a\}$, and set $L_{d_1+1}(\mathcal{T}_b(1)) := (I_b(1))$; if no such $d_1$ exists, take $d_1 = 0$. In general, for any entry $I_b(k)$ with $k > 1$, pick the maximal $d_k \leq d_{k-1}$ such that $|L_{d_k}(\mathcal{T}_b(k))| > |L_{d_k+1}(\mathcal{T}_b(k))|$, and set $L_{d_k+1}(\mathcal{T}_b(k)) := L_{d_k+1}(\mathcal{T}_b(k)) \oplus (I_b(k))$; take $d_k = 0$ if no such $d_k$ exists. Finally, we fix all other preexisting lanes. At the end of this iterative procedure, we obtain the lanes partitioning $\alpha$.

Example 2.3.1. Consider the cascading 10-sequence

$$(8, 9, 10, 8, 9, 10, 7, 8, 9, 7, 8, 9, 7, 8, 9, 8, 9, 6, 7, 8, 7, 8),$$

whose lower subintervals are $I_1 = (8, 9, 10), I_2 = (8, 9, 10), I_3 = (7, 8, 9), I_4 = (7, 8, 9), I_5 = (7, 8, 9), I_6 = (8, 9), I_7 = (6, 7, 8), I_8 = (7, 8)$. The lanes are formed in the following processes (with exactly one entry added to the lane at each stage):

1. $L_1(8) : (I_1(1)) \rightarrow (I_1(1), I_3(2)) \rightarrow (I_1(1), I_3(2), I_7(3))$

2. $L_1(9) : (I_1(2)) \rightarrow (I_1(2), I_3(3))$

3. $L_1(10) : (I_1(3))$

4. $L_2(8) : (I_2(1)) \rightarrow (I_2(1), I_4(2))$

5. $L_2(9) : (I_2(2)) \rightarrow (I_2(2), I_4(3))$

6. $L_2(10) : (I_2(3))$

7. $L_1(7) : (I_3(1)) \rightarrow (I_3(1), I_7(2))$

8. $L_2(7) : (I_4(1))$

9. $L_3(7) : (I_5(1))$

10. $L_3(8) : (I_5(2))$
11. \( L_3(9) : (I_5(3)) \)
12. \( L_4(8) : (I_6(1)) \rightarrow (I_6(1), I_8(2)) \)
13. \( L_4(9) : (I_6(2)) \)
14. \( L_1(6) : (I_7(1)) \)
15. \( L_4(7) : (I_8(1)) \)

Written another way, the lower subintervals and lanes are \( I_1 = (8^1, 9^1, 10^1), I_2 = (8^2, 9^2, 10^2), I_3 = (7^1, 8^1, 9^1), I_4 = (7^2, 8^2, 9^2), I_5 = (7^3, 8^3, 9^3), I_6 = (8^4, 9^4), I_7 = (6^1, 7^1, 8^1), I_8 = (7^4, 8^4) \), where lane \( i \) has been marked with a superscript \( i \).

Example 2.3.2. Let us now look at a more complex example. The cascading 10-sequence

\[
(6, 7, 8, 9, 10, 7, 8, 9, 10, 6, 7, 8, 9, 5, 6, 7, 8, 9, 7, 8, 9, 5, 6, 7, 8, 9, 6, 7, 8, 5, 6, 7, 8, 10, 8, 9, 10, 6, 7, 8, 9, 5, 6, 7, 8, 9)
\]

has lower subintervals and lanes \( I_1 = (6^1, 7^1, 8^1, 9^1, 10^1), I_2 = (7^2, 8^2, 9^2, 10^2), I_3 = (7^3, 8^3, 9^3, 10^3), I_4 = (8^4, 9^4, 10^4), I_5 = (6^2, 7^2, 8^2, 9^1), I_6 = (6^3, 7^3, 8^2, 9^2), I_7 = (7^4, 8^3, 9^3), I_8 = (5^1, 6^1, 7^1, 8^1), I_9 = (5^2, 6^2, 7^2, 8^2), I_{10} = (5^3, 6^3, 7^3, 8^3), I_{11} = (6^4, 7^4, 8^4), \) where lane \( i \) has been marked with a superscript \( i \).

We now show with more detail the formation of lanes at the stage where \( I_6 \) is acting.

The lanes formed before \( I_6 \) are:

\[
L_1(6) = (I_1(1))
\]
\[
L_1(7) = (I_1(2), I_5(2))
\]
\[
L_1(8) = (I_1(3), I_5(3))
\]
\[
L_1(9) = (I_1(4), I_5(4))
\]
\[
L_1(10) = (I_1(5))
\]
\[
L_2(6) = (I_5(1))
\]
\[
L_2(7) = (I_2(1))
\]
\[
L_2(8) = (I_2(2))
\]
\[
L_2(9) = (I_2(3))
\]
\[
L_2(10) = (I_2(4))
\]
\[
L_3(7) = (I_3(1))
\]
\[
L_3(8) = (I_3(2))
\]
\[
L_3(9) = (I_3(3))
\]
\[
L_3(10) = (I_3(4))
\]
\[
L_4(8) = (I_4(2))
\]
\[
L_4(9) = (I_4(3))
\]
\[
L_4(10) = (I_4(4))
\]

Lastly, we have \( L_3(6) = L_3(7) = L_3(8) = L_3(9) = L_4(7) = L_4(5) = L_2(5) = L_3(5) = L_4(6) = () \).

Assignment of entries of \( I_6 \):

We have \( L_3(6) := () \oplus (I_6(1)) = (I_6(1)) \), since \( d_1 = 2 \) for integer value 6.

We have \( L_2(7) := (I_2(1)) \oplus (I_6(2)) = (I_2(1), I_6(2)) \), since \( d_2 = 1 \) for integer value 7.

We have \( L_2(8) := (I_2(2)) \oplus (I_6(3)) = (I_2(2), I_6(3)) \), since \( d_3 = 1 \) for integer value 8.

We have \( L_2(9) := (I_2(3)) \oplus (I_6(4)) = (I_2(3), I_6(4)) \), since \( d_4 = 1 \) for integer value 9.
3 Cascading Sequences and Rigged Configurations

We use cascading sequences to give an explicit characterization (with a growth algorithm) of \( \mathcal{B}(\infty) \) rigged configurations in the \( A_n \) case, and we give an explicit bijection between these rigged configurations and cascading sequences. This results in an explicit bijection between the marginally large tableaux and \( A_n \) rigged configurations.

3.1 Rigged Configurations

The definition of \( \mathcal{B}(\infty) \) rigged configurations in the \( A_n \) case is given in [10], based on work done in [12]. We now recall this definition. Let \( \mathfrak{g} \) be a simply-laced Kac-Moody algebra with index set \( I \), and let \( \mathcal{H} := I \times \mathbb{Z}_{>0} \). Fix a multiplicity array

\[
L = (L_i^{(a)} \in \mathbb{Z}_{>0} : (a, i) \in \mathcal{H}).
\]

We typically define a partition to be a multiset of integers sorted in decreasing order. Define a \textbf{rigged partition} to be a multiset of integer pairs \((i, x)\) with \( i > 0 \), with these pairs sorted in decreasing lexicographic order. We will call each \((i, x)\) a \textbf{string}, with \( i \) the \textbf{size} or \textbf{length} of the string and \( x \) the \textbf{quantum number}, \textbf{label}, or \textbf{rigging} of the string. By a \textbf{rigged configuration} we will mean a pair \((\nu, J)\) where \( \nu = \{ \nu^{(a)} : a \in I \} \) is a sequence of rigged partitions and \( J = (J_i^{(a)})(a, i) \in \mathcal{H} \) where each \( J_i^{(a)} \) is the weakly increasing sequence of riggings of strings in \( \nu^{(a)} \) whose length is \( i \). The \textbf{vacancy number} of \( \nu \) is defined as

\[
p_i^{(a)} = p_i^{(a)}(\nu) = -\sum_{(b,j) \in \mathcal{H}} A_{ab} \min(i,j) m_j^{(b)},
\]

where \( m_j^{(b)} \) is the number of parts in the partition \( \nu^{(b)} \) with length \( j \). The \textbf{coquantum number} or \textbf{colabel} of a string \((i, x)\) is defined to be \( p_i^{(a)}(\nu) - x \). The \( a \)th part of \((\nu, J)\) is often denoted by \((\nu, J)^{(a)}\) for brevity.

To give the definition of \( \mathcal{B}(\infty) \) rigged configurations, denoted \( RC(\infty) \), let \( \nu_0 \) be the multipartition with all parts empty; that is, set \( \nu_0 = (\nu^{(1)}, \ldots, \nu^{(n)}) \) where \( \nu^{(a)} = \emptyset \) for all \((a, i) \in \mathcal{H} \). Therefore the rigging \( J_0 \) of \( \nu_0 \) must be \( J_i^{(a)} = \emptyset \) for all \((a, i) \in \mathcal{H} \).

**Definition 3.1.1.** The Kashiwara operators \( e_a \) and \( f_a \) act on elements \((\nu, J) \in RC(\infty)\) as follows: Fix \( a \in I \), and let \( x \) denote the smallest label of \((\nu, J)^{(a)}\), assuming \((\nu, J)^{(a)} \neq \emptyset \).

1. Set \( e_a(\nu, J) = 0 \) if \( x \geq 0 \). Otherwise, let \( l \) denote the smallest length of all strings which have label \( x \) in \((\nu, J)\). We obtain the rigged configuration \( e_a(\nu, J) \) by replacing the string \((l, x)\) with \((l - 1, x + 1)\) and then changing all the other labels to ensure that all colabels are preserved.

2. Add the string \((1, -1)\) to \((\nu, J)^{(a)}\) if \( x > 0 \). Otherwise, let \( l \) denote the greatest length of all strings which have label \( x \) in \((\nu, J)^{(a)}\). Replace the string \((l, x)\) by \((l + 1, x - 1)\), then change all the other labels to ensure that all colabels are preserved. The result is \( f_a(\nu, J) \).

If \((\nu, J)^{(a)} \) is empty, then \( f_a \) adds the string \((1, -1)\) to \((\nu, J)^{(a)}\).

RC(\infty) is the graph generated by \((\nu_0, J_0)\) using \( e_a \) and \( f_a \), for \( a \in I \).

We now give the remaining part of the crystal structure:

\[
\epsilon_a(\nu, J) = \max\{k \in \mathbb{Z}_{\geq 0} : e_a^k(\nu, J) \neq 0\},
\]
\[
\phi_a(\nu, J) = \epsilon_a(\nu, J) + \langle h_\alpha, \text{wt}(\nu, J) \rangle,
\]
where \(\{\alpha_a\}_{a \in \mathbb{I}}\) denotes the simple roots.

### 3.2 Kashiwara Operators Acting in a Cascading Sequence Arrangement

We show in this section that \(RC(\infty)\) Kashiwara operators act in a nice way when arranged in a cascading sequence. Let \(R = (\nu_1, \nu_2, \ldots, \nu_n)\) be a \(B(\infty)\) rigged configuration of \(A_n\) type where \(\nu_i\) is the \(i\)th rigged partition whose \(j\)th row has rigging \(\text{rig}^j_{\nu_i}\).

**Notation 3.2.1.** Whenever we write a rigged configuration in the form

\(R = (\nu_1, \nu_2, \ldots, \nu_n)\),

it is understood that each \(\nu_i\) is a rigged partition carrying the riggings information \(\text{rig}^j_{\nu_i}\).

Fix rigged partition \(\nu_m\). If \(\nu_m = \emptyset\), then we regard \(\nu_m\) as a single empty row \(r_1\) whose length is zero and whose rigging is zero. Generally, if we label as \(r_1, r_2, \ldots, r_k\) the rows of \(\nu_m\) from top to bottom, then we regard \(\nu_m\) as having an “empty row” \(r_{k+1}\) beneath \(r_k\), where \(r_{k+1}\) is understood to have zero length and a rigging of zero.

Let \(\alpha\) denote the cascading sequence of \(R\). Recall how the vacancy number changes when a Kashiwara operator acts on \(R\):

If the Kashiwara operator \(f_\alpha\) adds a box to a row of length \(l\) in \(\nu_a\), then the vacancy numbers of \(R\) are changed using the formula

\[
p_i^{(b)} = \begin{cases} 
  p_i^{(b)} & \text{if } i \leq l \\
  p_i^{(b)} - A_{ab} & \text{if } i > l 
\end{cases}
\]

where \(p_i^{(b)}\) denotes the vacancy number of a row of length \(i\) in \(\nu_b\), and

\[
A_{ab} = \begin{cases}
  -1 & \text{if } b = a \pm 1 \\
  2 & \text{if } b = a \\
  0 & \text{otherwise}
\end{cases}
\]

For each partition \(\lambda\), we will denote by \(\lambda^b\) the \(b\)th part (row) of \(\lambda\) and by \(\hat{\lambda}^b\) the portion of \(\lambda^b\) that has no boxes beneath it; we call \(\hat{\lambda}^b\) the **stretch** of \(\lambda^b\). For instance, \(T^4 = [1311]\). If \(\lambda\) is a rigged partition, by the **rigging of the stretch** \(\hat{\lambda}^b\) we will always mean the rigging of the row \(\hat{\lambda}^b\). Also, letting \(\lambda^t\) denote the transpose of \(\lambda\), \(l(\lambda) := \max(\lambda^t)\) is then the number of rows \(\lambda\) has.

By an integer sequence \(\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_p)\) **acting** on a rigged configuration \(R'\) we will always mean the corresponding sequence of Kashiwara operators \(\{f_i\}_{i \in \gamma}\) acting on \(R'\). More precisely, \(\gamma\) acts on \(R'\) by \(f_{\gamma}R' = f_{\gamma_p}f_{\gamma_{p-1}}\ldots f_{\gamma_1}R'\).

When working with cascading sequences, we can rely on the following useful lemmas:

Let \(I = (a, a + 1, \ldots, m)\) be an \(m\)-lower subinterval of the cascading sequence \(\alpha\). Denote by \(\alpha_I\) the subsequence of \(\alpha\) before \(I\). Let \(R_I = (\mu_1, \mu_2, \ldots, \mu_n)\) denote the preexisting rigged configuration (corresponding to \(\alpha_I\)). Whenever \(I\) acts on \(R_I\), it adds one box to each of the partitions \(\mu_a, \mu_{a+1}, \ldots, \mu_{m-1}, \mu_m\). As will be proven below, the box added to any \(\mu_j\) is of two forms: contributing and noncontributing. A **contributing box** is a single box

\[
\square
\]
which contributes $-1$ to the rigging of the row to which this box is added, contributes $-1$ to the rigging of any row of $\mu_j$ longer than the row to which it is added, and contributes $+1$ to the rigging of any row of $\mu_{j+1}$ longer than the row to which it is added, but does not change the riggings of $\mu_b$ for $b \neq j, j + 1$. A noncontributing box is a single box

\[ \square \]

with rigging $0$, which does not change the riggings of any rigged partition.

Remark 3.2.2. The contributing and noncontributing boxes describe the cumulative effect of the action of $f_I$, demonstrated in Lemma 3.2.4.

Let us analyze in more detail how $I$ acts on the preexisting rigged configuration $R_I$ corresponding to $\alpha_I$. For any partition $\lambda$ let $\bar{\lambda}$ denote the portion of $\lambda$ beneath the top row.

Lemma 3.2.3. Let $\lambda_1, \lambda_2$ be two partitions satisfying $\lambda_2 \subset \lambda_1 \subset \lambda_2$. Fix a positive integer $p$. Let $u_1$ be the uppermost row of $\lambda_1$ with length $p$ and $u_2$ be the uppermost row of $\lambda_2$ with $|u_2| \leq p$. Then every row of $\lambda_1$ below $u_1$ is no longer than $u_2$.

Proof. Suppose $u_1 = \lambda_1^1$ and $u_2 = \lambda_2^2$. Since $\lambda_1 \subset \lambda_2$, we must have $c \geq b$. Since $\lambda_2 \subset \lambda_1$, we must have $c \leq b + 1$. If $c = b$, then we have $|u_1| = |u_2|$, and the claim follows immediately. Suppose $c = b + 1$. Then we have $|\lambda_1^1| \leq |\lambda_2^{b+1}| = |u_2|$ for any $d \geq b + 1$, since $\lambda_1 \subset \lambda_2$.

Lemma 3.2.4 (Main Lemma). $R_I = (\mu_1, \mu_2, \ldots, \mu_m)$ for $I = (a, a + 1, \ldots, m)$ satisfies the following properties. Let $r_a$ be the top row of $\mu_a$, and $r_i$ be the uppermost row of $\mu_i$ with $|r_i| \leq |r_{i-1}|$.

1. The partitions $\mu_1, \mu_2, \ldots, \mu_{m-1}$ have all riggings equal to zero. For $l \in [m - 1]$ we have $\mu_{l+1} \subset \mu_l \subset \mu_{l+1}$. By Lemma 3.2.3 it follows in particular that:
   Fix a positive integer $p$. For $l \in [m - 1]$ let $u_l$ be the uppermost row of $\mu_l$ with length $p$, and let $u_{l+1}$ be the uppermost row of $\mu_{l+1}$ such that $|u_{l+1}| \leq p$. Then every row of $\mu_l$ below $u_l$ must be no longer than $u_{l+1}$.

2. The rows of the rigged partition $\mu_m$ above $r_m$ have the same rigging as $r_m$, and this rigging is non-positive and minimal in $\mu_m$.

3. $I$ acts on $R_I$ by adding a noncontributing box to the rows $r_a, r_{a+1}, \ldots, r_{m-1}$ and a contributing box to the row $r_m$.

Proof. Induction. $R_I$ clearly satisfies these properties if $I$ is the first or second lower subinterval of $\alpha$. Now consider the general case, assuming that $R_I$ satisfies these properties.

Let $R'_I = (\mu'_1, \mu'_2, \ldots, \mu'_m)$ denote the rigged configuration corresponding to $\alpha'_I := \alpha_I \oplus I$. We apply $I$ to $R'_I$ to obtain $R''_I$ and prove that it satisfies these properties as well.

We first check Property 3 and the first statement of Property 1 for $R''_I$. By Property 1 for $R_I$, Kashiwara operator $f_a$ adds a box to the first row $r_a$ of $\mu_a$, adding $-1$ to its rigging, adding $+1$ to the vacancy number as well as the rigging of rows of $\mu_{a+1}$ longer than $r_a$, and not changing the riggings of $\mu_1, \ldots, \mu_{a-1}$. $f_{a+1}$ then adds a box to the uppermost row $r_{a+1}$ of $\mu_{a+1}$ with $|r_{a+1}| \leq |r_a|$, adding $-1$ to its rigging, adding $-2$ to the vacancy number as well as the rigging of rows of $\mu_{a+1}$ longer than $r_{a+1}$ (so these rows end up with a rigging of $1 - 2 = -1$), adding $+1$ to the vacancy number and the rigging of rows of the ath partition longer than $r_{a+1}$ (which by Property 1 gives $\mu'_a$ with zero riggings), and adding $+1$ to the vacancy number as well as the rigging of rows of $\mu_{a+2}$ longer than $r_{a+1}$. $f_{a+2}$ then adds a box to the uppermost row $r_{a+2}$ of $\mu_{a+2}$ with $|r_{a+2}| \leq |r_{a+1}|$, adding $-1$ to its rigging, adding $-2$ to the vacancy number and the rigging of rows of $\mu_{a+2}$ longer than $r_{a+2}$ (so these rows end up with a rigging of $-1 - 2 = -3$).
12−2 = −1), adding +1 to the vacancy number and the rigging of rows of the (a+1)st partition longer than $r_{a+2}$ (which by Property \[1\] gives $\mu_{a+1}'$ with zero riggings), and adding +1 to the vacancy number as well as the rigging of rows of $\mu_{a+3}$ longer than $r_{a+2}$. Iterating this process, for $j = 0, 1, \ldots, m - a - 2$ we obtain $\mu_{a+j}'$ by adding a noncontributing box to row $r_{a+j}$ of $\mu_{a+j}$ so $\mu_{a+j}'$ has zero riggings.

Now, after $f_{m-1}$ added a box to row $r_{m-1}$ of $\mu_{m-1}$, all rows of the resulting $(m-1)$st partition with length at least $|r_{m-1}| + 1$ have rigging −1. By Property \[2\] for $R_l$, this action of $f_{m-1}$ must have contributed +1 to the vacancy number and the rigging of all rows of $\mu_m$ longer than $r_{m-1}$, and consequently these rows of $\mu_m$ now have greater rigging than $r_m$ does, so $r_m$ is now the longest row of $\mu_m$ with the smallest rigging. Finally, $f_m$ adds a box to $r_m$, adding −1 to its rigging, adding +1 to the vacancy number and the rigging of rows of the $(m-1)$st partition longer than $|r_m|$, (which by Property \[1\] gives $\mu_{m}'$ with zero riggings), adding −2 to the vacancy number and the rigging of rows of the $m$th partition longer than $r_m$ (so these rows now have the same rigging as rows of length $|r_m| + 1$), and adding +1 to the vacancy number as well as the rigging of rows of $\mu_{m+1}$ longer than $|r_m|$. This shows that $\mu_{m-1}'$ is obtained from $\mu_{m-1}$ by adding a noncontributing box to $r_{m-1}$, and that $\mu_{m}'$ is obtained from $\mu_m$ by adding a contributing box to $r_m$.

Now we verify Property \[2\] for $R_1 = (\mu_1', \mu_2', \ldots, \mu_n')$. By above, we conclude that rows of $\mu_m'$ with length at least $|r_m| + 1$ have identical rigging, and this rigging is minimal and non-negative. Let $a' \geq a$. Let $r_{a'}'$ be the top row of $\mu_{r_{a'}}'$, and let $r_{a'}^k$ be the uppermost row of $\mu_{r_{a'}}'$ with $|r_{a'}^k| \leq |r_{a'}^k - 1|$, for $k = a' + 1, a' + 2, \ldots, m$. Notice that we have $|r_{a'}^k| = |r_a| + 1$ if $a' = a$ and we have $|r_{a'}^k| \geq |r_a| + 1$ if $a' > a$. Since $\mu_{r_{a'}}'$ contains a row with length $|r_k| + 1$, we have $|r_{a'}^k| \geq |r_k| + 1$, for $k = a', a'+ 1, \ldots, m$. It follows that the rows of $\mu_m'$ above $r_{a'}'$ have the same rigging as $r_{a'}'$, and this rigging is non-positive and minimal in $\mu_m'$.

Finally, we verify the second statement of Property \[1\] for $R_1 = (\mu_1', \mu_2', \ldots, \mu_n')$. If $l, l + 1 < a$, then the claim follows by hypothesis. If $l = a - 1$, then the claim follows since $\mu_{a-1}' = \mu_a'$ is obtained from $\mu_a$ by adding a single box to the first row. Now suppose $l \in [a, m - 1]$. $\mu_1', \mu_{l}'$ are obtained from $\mu_1, \mu_{l}$ respectively by adding a box via Property \[3\]. Let $r_l = \mu_l', r_{l+1} = \mu_{l+1}'$ be the rows of $\mu_1, \mu_{l+1}$ respectively to which the box was added. Then $r_{l+1}$ is the uppermost row of $\mu_{l+1}$ no longer than $r_l$. Since $\mu_1 \subset \mu_l$, we must have $d \leq c$. Since $\mu_{l+1} \subset \mu_{l}$, we must have $d \leq c + 1$. Thus, either $d = c$ or $d = c + 1$. Suppose $d = c$. Then $|r_l| = |r_{l+1}|$, and it follows immediately that $\mu_l' \subset \mu_{l+1}'$. If $r_{l+1}$ is the first row, then $\mu_{l+1}' \subset \mu_l'$ by hypothesis. If $r_{l+1}$ is not the first row, we still have $\mu_{l+1}' \subset \mu_l'$ since $|\mu_l'|_{c-1} = |\mu_l'|_{c-1} \geq |r_{l+1}| + 1 = |\mu_l'|_{c-1}$, and we have $\mu_{l+1}' \subset \mu_{l+1}'$. Since $|r_{l+1}| = |\mu_{l+1}'| \leq |\mu_l'| = |r_l|$, we have $|\mu_{l+1}'| = |r_{l+1}| + 1 \leq |\mu_l'| + 1 = |\mu_l'|_{c}$, and thus we have $\mu_{l+1}' \subset \mu_l'$. This completes the induction. □

Remark 3.2.5. It is easy to see that the first containment $\mu_{l+1} \subset \mu_l$ in fact holds for all $l \in [n-1]$, since no more boxes will be added to the $(l+1)$st partition once all the $(l+1)$-lower subintervals have acted.

It follows immediately from Lemma 3.2.4 that

**Lemma 3.2.6.** The following are true.

1. If $\alpha$ ends in a $p$-lower subinterval, then $\nu_q$ of $R$ has zero riggings for all $q \leq p - 1$.

2. All contributing boxes (and hence negative riggings) to the $\nu_m$ of $R$ are added by $m$-lower subintervals of $\alpha$.

3. All positive contributions to the riggings of $\nu_m$ are added by $(m-1)$-lower subintervals of $\alpha$, which add no boxes to $\nu_m$. 

11
4. Suppose $I_1, I_2$ are $m$-lower subintervals of $\alpha$ with $I_1$ preceding $I_2$. If $I_1$ adds a contributing box to column $i_1$ and $I_2$ adds a contributing box to column $i_2$ of the $m$th partition, then we have $i_1 < i_2$.

Proof. The first three items follow immediately from the lemma. For the fourth item, note that $I_1 = (a_1, \ldots, m)$ and $I_2 = (a_2, \ldots, m)$, where $a_1 \leq a_2$. Let $r_{a_1}$ be the top row of $\mu_{a_1}$, and $r_i$ be the uppermost row of $\mu_i$ with $|r_i| \leq |r_{i-1}|$. Let $r'_{a_2}$ be the top row of $\mu_{a_2}$, and $r'_i$ be the uppermost row of $\mu_i$ with $|r'_i| \leq |r'_{i-1}|$. Notice that $|r'_k| > |r_k|$ for $k = a_2, a_2 + 1, \ldots, m$. Applying Property 3 of Lemma 3.2.4, we deduce that the contributing box added by $I_2$ must be strictly to the right of the contributing box added by $I_1$.

We thus obtain the following interesting result.

**Theorem 3.2.7.** Identical rows of $\nu_m$ of $R$ have equal riggings, for any $m \in [n]$.

Proof. Before any $m$-lower subinterval has acted, the $m$th partition has zero riggings. After the first $m$-lower subinterval adds a contributing box to row $r$, every row with length at least $|r| + 1$ has rigging $-1$, while the rigging of every row with length at most $|r|$ remains unchanged. In general, assume that the $j$th $m$-lower subinterval has added a box to row $r'$ of the $m$th partition, so that rows with length at least $|r'| + 1$ have equal rigging, and that identical rows with length at most $|r'|$ have equal rigging. By the fourth item of Lemma 3.2.6 the $(j + 1)$st $m$-lower subinterval adds a contributing box to row $r''$ with $|r''| \geq |r'| + 1$. In the resulting $m$th partition, rows of length at most $|r''|$ have unchanged rigging, which is the same for identical rows, while the new row with length $|r''| + 1$ and other rows with length at least $|r''| + 1$ receive a $-1$ contribution to their identical riggings. This shows that the riggings of identical rows remain equal after all the $m$-lower subintervals have acted.

Similarly, each time an $(m - 1)$-lower subinterval acts, all the rows of $\nu_m$ no longer than a certain length $k$ experience no change in rigging, while all the rows of $\nu_m$ longer than $k$ receive $+1$ contribution to the rigging. Therefore, the riggings of identical rows remain equal after all the $(m - 1)$-lower subintervals have acted.

### 3.3 Obtaining the Rigged Configuration from the Cascading Sequence

Now we relate the concepts in Subsection 3.3 to the $B(\infty)$ rigged configurations in the $A_n$ case. Let $R = (\nu_1, \nu_2, \ldots, \nu_n)$ be an $A_n$ rigged configuration. Let $\alpha$ denote the cascading sequence of $R$. We can obtain any partition in the corresponding rigged configuration without doing explicit calculation via the Kasuhrara operators involved. This is done by partitioning $\alpha$ into lanes and then analyzing the relevant lanes.

Each column of $\nu_1$ ends in exactly one of the stretches of $\nu_1$. We denote by $\text{col}(\nu_1^b)$ the set of columns of $\nu_1$ ending in the stretch $\nu_1^b$, and let $W_1^b := \{\text{lanes } L_i(l) | L_i(l) = b\}$.

By Lemma 3.2.4, the $l$-lanes correspond precisely to the columns of $\nu_1$, and we have the following useful facts:

**Lemma 3.3.1.** The set $\text{col}(\nu_1^b)$ corresponds to the set $W_1^b$. In fact, under the definition of lanes, $L_i(l)$ corresponds exactly to the $i$th column of $\nu_1$, with the height of the column given by $|L_i(l)|$.

Proof. Adding a box to the longest row $r$ with $|r| \leq c$ is the same as adding a box to the $d$th column for maximal $d \leq c + 1$ whose height is strictly less than that of the $(d - 1)$st column.

**Remark 3.3.2.** In particular, the number of columns of height $b$ in $\nu_1$ is given by the number of $l$-lanes $L_i(l)$ with $|L_i(l)| = b$ in the corresponding cascading $n$-sequence.
Roughly speaking, the riggings of $v_l$ are determined by the number of $l$-lanes that contain the right endpoint of some lower subinterval and by the number of $(l-1)$-lanes that contain the right endpoint of some lower subinterval. In Example 2.3.2 if we fix $l = 9$, then the 9-lane $L_2(9)$ is an example of the former because it contains the right endpoint of $I_5$, and the 8-lane $L_3(8)$ is an example of the latter because it contains the right endpoint of $I_9$.

**Lemma 3.3.3.** Suppose $r$ is a row of $v_l$. Let $V^l_r$ denote the set of $l$-lanes $L_i(l)$ ending at a right endpoint, where $i \leq |r|$. Let $V^{l-1}_r$ denote the set of $(l-1)$-lanes $L_i(l-1)$ ending at a right endpoint, where $i \leq |r|$. Then the rigging of $r$ is given by $-|V^l_r| + |V^{l-1}_r|$.

**Proof.** Lemma 3.2.0 gives us that at most one contributing box can be added to a column. The rigging of $r$ is determined by the number of contributing boxes (negative contribution) that were added to the columns of the $l$th partition occupied by $r$ and the number of contributing boxes (positive contribution) that will have been added to the same columns of the $(l-1)$st partition; the former number corresponds to the term $-|V^l_r|$, while the latter number corresponds to the term $|V^{l-1}_r|$.

For any entry $\chi$ of $\alpha$ with value $|\chi|$, if $\chi$ is the $r$th entry of the lane $L_n(|\chi|)$, we say that $\chi$ has lane depth $v$, $\chi$ has lane number $u$, and we also refer to $u$ as the lane number of $L_n(|\chi|)$. In Example 2.3.2 the entry 8 of $I_9$ has lane number 2 and depth 3.

**Lemma 3.3.4.** $v_l$ has at most $\max_l := \min(n-l,l-1) + 1 = \min(n-l+1,l)$ rows.

**Proof.** It suffices to consider $L_1(l)$, which corresponds to the first column of $v_l$. Each entry of $L_1(l)$ is an entry of some lower subinterval $I$ with $\min I \leq |l|$ and $\max I \in [l,n]$. Suppose that $L_1(l)(j), L_1(l)(j+1)$ are contained in an $m_1$-lower subinterval, an $m_2$-lower subinterval respectively. By Lemma 3.2.0 Property 4 we must have $m_1 \neq m_2$. Since the $m_1$-lower subintervals must precede the $m_2$-lower subintervals, it follows that $m_2 < m_1$. Hence we have $|L_1(l)| \leq |[l,n]| = n-l+1$. On the other hand, notice that $v_1$ can only be a single row, by Lemma 3.2.1. By Lemma 3.2.1 for any lower subinterval $I'$, the lane depth of entry $b+1$ of $I'$ exceeds the lane depth of entry $b$ of $I'$ by at most one. Thus, inductively we have $|L_1(l)| \leq |[l]| = l$ as well.

**Lemma 3.3.5.** If $l > \frac{n+1}{2}$, then $\max_l = n-l+1$. If $l \leq \frac{n+1}{2}$, then $\max_l = l$.

**Proof.** If $l > \frac{n+1}{2}$, then $n-l+1 < n - \frac{n+1}{2} + 1 = \frac{2n+1}{2} < l$. If $l \leq \frac{n+1}{2}$, then $n-l+1 \geq n - \frac{n+1}{2} + 1 = \frac{n+1}{2} \geq l$. The claims then follow by Lemma 3.3.4.

**Lemma 3.3.6.** We have $\max_{l-1} \leq \max_l$ if and only if $l-1 > n-l$ (equivalently $l > \frac{n+1}{2}$).

**Example 3.3.7.** Consider the element

$$(7, 8, 9, 10, 7, 8, 9, 10, 8, 9, 10, 6, 7, 8, 9, 6, 7, 8, 9, 7, 8, 9, 5, 6, 7, 8, 7, 8, 7, 8, 7, 8)$$

of $A_{10}$, whose lower subintervals are $(7^1, 8^1, 9^1, 10^1), (7^2, 8^2, 9^2, 10^2), (8^3, 9^3, 10^3), (6^1, 7^1, 8^1, 9^1), (6^2, 7^2, 8^2, 9^2), (7^3, 8^3, 9^3), (5^1, 6^1, 7^1, 8^1), (5^2, 6^2, 7^2, 8^2), (7^4, 8^4)$, where the lanes have been marked with superscripts. From this information, we can tell, for example, that the 9th partition of this rigged configuration has exactly three columns of height 2, the 8th partition has exactly one column of height 2, and the 7th partition has height 1 for both its third and fourth columns.

We can apply Lemmas 3.3.1 and 3.3.3 to obtain the $l$th partition in the rigged configuration as well as its riggings, given the corresponding cascading $n$-sequence. We illustrate this in the following
Example 3.3.8. In the corresponding rigged configuration in Example 2.3.1 the 10th partition is

\[
\begin{array}{cccc}
\hline \\
\end{array}
\]

with rigging $-2+2=0$, since $L_1(10), L_2(10)$ end at right endpoints (contributing $-1-1$ to the rigging) and since $L_1(9), L_2(9)$ also end at right endpoints (contributing $+1+1$ to the rigging). The 9th partition is

\[
\begin{array}{cccc}
\hline \\
\end{array}
\]

with rigging $-1$ for the second row and rigging $-2$ for the first row, since $L_1(9), L_2(9), L_3(9), L_4(9)$ all end at right endpoints (contributing $-1-1$ to the rigging of the second row and $-1-1-1-1$ to the rigging of the first row) with $|L_1(9)| = |L_2(9)| = 2$ and $L_3(9) = L_4(9) = 1$, and since $L_1(8), L_4(8)$ end at right endpoints (contributing $+1$ to the rigging of the second row and $+1+1$ to the rigging of the first row).

Similarly, the 8th partition is

\[
\begin{array}{cccc}
\hline \\
\end{array}
\]

with riggings $-1$ for the third row, $-2$ for the second row, and $-2$ for the first row.

Example 3.3.9. In Example 2.3.2 the cascading 10-sequence

\[
(6, 7, 8, 9, 10, 7, 8, 9, 10, 8, 9, 10, 6, 7, 8, 9, 7, 8, 9, 5, 6, 7, 8, 5, 6, 7, 8, 5, 6, 7, 8)
\]

has lower subintervals and lanes $I_1 = (6^1, 7^1, 8^1, 9^1, 10^1), I_2 = (7^2, 8^2, 9^2, 10^2), I_3 = (7^3, 8^3, 9^3, 10^3), I_4 = (8^4, 9^4, 10^4), I_5 = (6^2, 7^1, 9^1), I_6 = (6^3, 7^2, 8^2, 9^2), I_7 = (7^4, 8^3, 9^3), I_8 = (5^1, 6^1, 7^1, 10^3), I_9 = (5^2, 6^2, 7^2, 8^3), I_{10} = (5^3, 6^3, 7^3, 8^3), I_{11} = (6^4, 7^4, 8^4)$, where lane $i$ has been marked with a superscript $i$.

Looking at these lanes, we can tell that

1. $\nu_{10}$ has four columns of length one, with $\text{rig}^{1}_{10} = -4+3 = -1$, since $L_1(10), \ldots, L_4(10)$ end at right endpoints, and since $L_1(9), L_2(9), L_3(9)$ end at right endpoints
2. $\nu_9$ has three columns of length two, and one column of length one, with $\text{rig}^{2}_{9} = -3+3 = 0$ and $\text{rig}^{1}_{9} = -3+3+1 = 1$, since $L_1(9), L_2(9), L_3(9)$ end at right endpoints but $L_4(9)$ does not, and since $L_1(8), \ldots, L_4(8)$ end at right endpoints
3. $\nu_8$ has three columns of length three and one column of length two, with $\text{rig}^{3}_{8} = -3$ and $\text{rig}^{2}_{8} = \text{rig}^{1}_{8} = -3-1 = -4$, since $L_1(8), \ldots, L_4(8)$ end at right endpoints

The rigged configuration (with $\nu_i$ in top-bottom order) in its entirety is
Lastly, the following theorem imposing constraints (in a recursive manner, starting from the last partition) on the range of possible legitimate $B(\infty)$ rigged configurations of type $A$ also follows from Lemma 3.2.4 (which states that at most one noncontributing box can be added to each column), Lemma 3.2.6 Property 4 (which states that at most one contributing box can be added to each column), and Lemma 3.3.4. This result is the first half of our classification of rigged configurations. For convenience of description, we will regard $\nu_m$ as having $|\nu_1^m| - |\nu_2^m|$ columns of height zero to the right of its first row (i.e. $\nu_m$ has an empty row of length $|\nu_1^m| - |\nu_2^m|$ to the right of its first row).

**Theorem 3.3.10.** $\nu_{m-1}$ is obtained from $\nu_m$ in three stages:

1. Add at most one noncontributing box to each column of $\nu_m$, resulting in a partition $\nu_{m-1}$.  
2. Add at most one contributing box to each column of $\nu_{m-1}$, resulting in a partition $\nu_{m-1}'$.  
3. Finally add a number of contributing boxes to the first row of $\nu_{m-1}'$.  

In this process, any column of $\nu_m$ with height $\max((\nu_m)^t)$ must receive at most $\min(\max r_{m-1} - \max((\nu_m)^t), 2)$ boxes.

We will use the following restatement extensively.

**Theorem 3.3.11** (Restatement of Theorem 3.3.10). $\nu_{m-1}$ is obtained from $\nu_m$ by adding at most two boxes to each column of $\nu_m$, the first box being noncontributing and the second box being contributing. Moreover, we have the following constraints:

1. At most $\min(\max r_{m-1} - \max((\nu_m)^t), 2)$ boxes can be added to any column of height $\max((\nu_m)^t)$.
2. At most one box can be added to the $d$th column for $d > |\nu_m^1|$, and this box must be contributing.

3. In any row of $\nu_{m-1}$, no contributing box precedes a noncontributing box.

Remark 3.3.12. Item 2 simply states that the resulting $(m - 1)$st partition cannot have more rows than $\max r_{m-1}$. Any box added to a column of $\nu_m$ with height zero will be in the first row of the resulting partition. Needless to say, there cannot be any gaps between the boxes added to any row, as the result would not be a valid partition.

Even though we have not specified the rigging of $\nu_m$ here, this theorem gives us the “at most two boxes to each column” constraint. Precisely how the rigging of $\nu_m$ constrains $\nu_{m-1}$ will be handled in later sections.

3.4 Rough Idea of the Algorithm

Given an $m$-lower subinterval $I = (a, a + 1, \ldots, m)$, we say that the lower subinterval $I_+ = (a - 1, a, a + 1, \ldots, m)$ is the lengthening of $I$, and we say that we lengthen $I$ to obtain $I_+$. Our characterization for the $A_n$ rigged configurations will be an algorithm for growing rigged configurations starting from the last ($n$th) rigged partition; this growth algorithm can determine whether any given $n$-tuple of rigged partitions is a legitimate $A_n$ rigged configuration. In other words, given the last partition (which consists of a row with any number of boxes), we can give the range of all possible $(n - 1)$st partitions and its riggings. In general, given the $n$th, $(n - 1)$st, ..., $(n - i)$th partitions, we can give the range of all possible $(n - i - 1)$st partitions and its riggings.

Growing the rigged configuration in our algorithm corresponds to growing its corresponding cascading $n$-sequence. Note that any cascading $n$-sequence can be constructed by first adding copies of $n$ to the (initially empty) string, then copies of $n - 1$ to the string, then copies of $n - 2$ to the string, and so on, such that we have a cascading $n$-sequence at each stage. It follows a fortiori that any cascading $n$-sequence can be constructed by first adding copies of $i \leq n$ to the (initially empty) string, then copies of $i \leq n - 1$ to the string, then copies of $i \leq n - 2$ to the string, and so on, such that we have a cascading $n$-sequence at each stage. Hence any $A_n$ rigged configuration can be constructed (by applying the Kashiwara operators in the order of the cascading $n$-sequence at each stage) via this type of iterative process, which constructs the $n$th partition, $(n - 1)$st partition, $(n - 2)$nd partition, and so on, in that order. What we need to do is to fine tune this process so that the already constructed $n$th, $(n - 1)$st, ..., $(n - i)$th partitions and their riggings do not change when we construct the $(n - i - 1)$st partition. More precisely, at the $i$th stage, we will add all the copies of $n - i + 1$ along with minimal copies of $j < n - i + 1$ necessary to preserve the previously constructed rigged partitions; we will elaborate on this in the next few subsections.

Remark 3.4.1. We mention that, by Lemma 3.3.1 if $\nu$ is the $l$th partition in the rigged configuration, then the stretch $\tilde{\nu}^b$ corresponds to the set $\{L_i(l) || L_i(l) = b\}$. In other words, the stretch $\tilde{\nu}^b$ corresponds to the set of $l$-lanes of length $b$, or equivalently the set of columns of $\nu$ with height $b$.

Let $R = (\nu_1, \nu_2, \ldots, \nu_n)$ be a rigged configuration we want to construct by our growth algorithm. To construct the compatible rigged partition $\nu_{l-1}$ given that we have already constructed $\nu_1, \nu_{i+1}, \ldots, \nu_n$, where the riggings of $\nu_1, \nu_{i+1}, \ldots, \nu_n$ are fixed, we will add noncontributing boxes and contributing boxes beneath the stretches of $\tilde{\nu}^b$ (which has zero riggings by default). Roughly speaking, at most two rows of boxes will be added beneath each stretch of $\tilde{\nu}^b$ with the first row consisting of noncontributing boxes and the second row consisting of contributing boxes. This is justified by Theorem 3.3.11. Of course, the contributing boxes added beneath each stretch of
must account for the riggings of $\nu_1$, by Lemma 3.3.3. Before describing how to add boxes to $\overline{\nu}$, we need the notion of *plateaus* to delineate the stretches of a rigged partition to which boxes can be added.

### 3.5 Plateaus as Base for Construction

**Definition 3.5.1.** We say that a cascading sequence $\beta$ (as well as its corresponding rigged configuration) is a $(p, q, r)$-plateau if it satisfies the following property:

1. For every $i$ for which $L_i(p)$ exists, we have $|L_i(p-1)| = |L_i(p)| - 1$ whenever $|L_i(p-1)| < q$.
2. For every $(p-1)$-lane $L_i(p-1)$ with $|L_i(p-1)| \leq r$, $L_i(p-1)$ does not end at a right endpoint.
3. For any $k < p-1$ no $k$-lane ends at a right endpoint.

If the above property holds for $q = \infty$ and $r = \infty$, then we call $\beta$ a $p$-plateau. If $\beta$ is an $m$-plateau for every $m \in [p]$, then we call $\beta$ a $p^*$-plateau; here (and in what follows) we will exclude the case $m = 1$ from consideration, as the letter 0 does not occur in $\beta$.

**Lemma 3.5.2.** If $\beta$ is a $p^*$-plateau corresponding to the rigged configuration $(\mu_1, \mu_2, \ldots, \mu_n)$, then $\mu_{i-1} = \overline{\mu}$ with zero riggings for $l \in [p]$.

**Proof.** First we show that a right endpoint can only exist at the end of a lane. Suppose a right endpoint occurs in an $l$-lane $L_l(l)$, and let $I$ denote the lower subinterval containing this right endpoint. By the definition of cascading sequences, the only lower subintervals after $I$ containing $l$ as an entry must have $l$ as a right endpoint. However, any lower subinterval after $I$ that contains $l$ as a right endpoint must add its right endpoint to a lane $L_k(l)$ where $k > j$, by Lemma 3.2.6.

Since right endpoints can only occur at the end of a lane, we have $\mu_{i-1} = \overline{\mu}$ by the definition of $p^*$-plateau, and we have that $\mu_l$ has zero riggings by Lemma 3.3.3 for $l \in [p]$.

**Example 3.5.3.** The cascading sequence consisting of lower subintervals $(7^1, 8^1, 9^1), (8^2, 9^2), (8^3, 9^3), (9^4), (6^1, 7^1, 8^1), (7^2, 8^2), (8^4)$ is a $7^*$-plateau and a $(8, 2, \infty)$-plateau. The cascading sequence consisting of the lower subintervals $(7^1, 8^1, 9^1, 10^1), (7^2, 8^2, 9^2, 10^2), (8^3, 9^3, 10^3), (8^4, 9^4, 10^4), (8^5, 9^5, 10^5), (6^1, 7^1, 8^1, 9^1), (7^2, 8^2, 9^2), (5^1, 6^1, 7^1, 8^1), (6^2, 7^2, 8^2), (6^3, 7^3, 8^3), (5^2, 6^2, 7^2)$ is a $7^*$-plateau and an $(8, 2, 2)$-plateau.

**Remark 3.5.4.** As a start, notice that the cascading sequence consisting of the (singleton) lower subintervals $(n), (n), \ldots, (n)$ is an $n^*$-plateau.

We now present procedures for adding at most two boxes to each column of the $(p-1)$st partition of a rigged configuration that is both a $(p, q, r)$-plateau and a $(p-1)^*$-plateau, generating all possible $(p-1)$st partitions compatible with the predetermined $p$th, $(p+1)$st, \ldots, $n$th partitions. Rigged configurations that are both a $(p, q, r)$-plateau and a $(p-1)^*$-plateau will serve as the “skeletons” upon which boxes are added in our growth algorithm.

### 3.6 Adding Boxes to a Stretch

Since any stretch $s$ of a partition $\lambda$ corresponds to all columns of some fixed height $\text{ht}(s)$, we will also refer to $\text{ht}(s)$ as the height of the stretch $s$. 

---

17
Convention 3.6.1. For any $A_n$ rigged configuration $R' = (\nu_1, \nu_2, \ldots, \nu_n)$, by Remark 3.2.5 we already know that $\nu_{i-1} \supseteq \nu_i$. If we label the stretches of $\nu_i$ from bottom to top by $g_1, g_2, \ldots, g_k$, and the stretches of $\nu_1$ from bottom to top by $g'_1, g'_2, \ldots, g'_l$, then clearly $g'_i$ is identical to $g_i$ for $i \in [k - 1]$. Here $g'_k$ is identical to $g_k$ if $|\nu'_1| = |\nu_1^2|$, and is empty if $|\nu'_1| > |\nu_1^2|$. In the case $|\nu'_1| > |\nu_1^2|$, we will refer to $g'_k$ as an “invisible” stretch above the first row of $\nu_1$, with $|g'_k| = |g_k|$. Thus, in either case, we will regard $\nu_1$ as having identical copies of all the stretches of $\nu_i$; this will be convenient for when we talk of adding boxes to $\nu_1$ to form $\nu_{i-1}$, where a box added beneath $g'_k$ will be in the first row of the resulting partition in the case $|\nu'_1| > |\nu_1^2|$.

Now, fix cascading sequence $\beta$ that is both a $(p - 1)^*\text{-plateau}$ and a $(p, q, r)\text{-plateau}$, with corresponding rigged configuration $R = (\mu_1, \mu_2, \ldots, \mu_n)$. We give two procedures for adding respectively noncontributing boxes and contributing boxes beneath the stretches $z_1, z_2, \ldots, z_a$ of $\mu_{p-1}$ (ordered from bottom to top, following Convention 3.6.1), which fixes $\mu_{a}$ for all $x > p$ and also fixes the shape of $\mu_{x}$ (though not necessarily the rigging, which will depend on the resulting $(p - 1)\text{-st}$ partition). Assume that $ht(z_1) < \max x_{p-1}$; otherwise no boxes can be added beneath $z_1$. Both procedures output both the desired cascading sequence and its corresponding rigged configuration, and can be applied repeatedly to add boxes to multiple stretches sequentially.

Procedure 3.6.2 (Adding Noncontributing Boxes to a Given Stretch). Suppose $ht(z_i) < \min (q, r)$. The following algorithm adds $n_i$ noncontributing boxes beneath $z_i$, where $0 \leq n_i \leq |z_i|:

We will add $n_i$ copies of $p - 1$, $n_i$ copies of $p - 2$, $\ldots$, $n_i$ copies of $p - (ht(z_i) + 1)$ to $\beta$ as follows. Let $B_v$ denote the set of lower subintervals with head $v$. We will delete a number of elements from each $B_v$ before lengthening the first $n_i$ of the remaining elements.

Label from left to right by $I^p_{ht(z_i)-1}, I^p_{ht(z_i)-2}, \ldots, I^p_{ht(z_i)}$ the elements of $B_{p-ht(z_i)+1}$. Then delete the element $I^p_{ht(z_i)+1} \in B_{p-ht(z_i)+1}$ nearest $I^p_{ht(z_i)+1} \in B_{p-ht(z_i)+1}$ left of $I^p_{ht(z_i)-1}$, delete the element $I^p_{1} \in B_{p-ht(z_i)}$ nearest $I^p_{2} \in B_{p-ht(z_i)}$ left of $I^p_{1}$, $\ldots$, delete the element $I^p_{w} \in B_{p-ht(z_i)+1}$ nearest $I^p_{w} \in B_{p-ht(z_i)+1}$ left of $I^p_{w}$. Let $\overline{B}_{p-ht(z_i)}$ denote the subset obtained from $B_{p-ht(z_i)}$ after performing this sequence of deletions. Next, delete the element $I^p_{1} \in B_{p-ht(z_i)+1}$ nearest $I^p_{1} \in B_{p-ht(z_i)+1}$ left of $I^p_{1}$, delete the element $I^p_{2} \in B_{p-ht(z_i)+1}$ nearest $I^p_{2} \in B_{p-ht(z_i)+1}$ left of $I^p_{2}$, $\ldots$, delete the element $I^p_{w} \in B_{p-ht(z_i)+1}$ nearest $I^p_{w} \in B_{p-ht(z_i)+1}$ left of $I^p_{w}$. Let $\overline{B}_{p-ht(z_i)+1}$ denote the subset obtained from $B_{p-ht(z_i)+1}$ after performing this sequence of deletions. In general, delete the element $I^p_{1} \in B_{p-ht(z_i)+c}$ nearest $I^p_{1} \in B_{p-ht(z_i)+c}$ left of $I^p_{1}$, delete the element $I^p_{2} \in B_{p-ht(z_i)+c}$ nearest $I^p_{2} \in B_{p-ht(z_i)+c}$ left of $I^p_{2}$, $\ldots$, delete the element $I^p_{w} \in B_{p-ht(z_i)+c}$ nearest $I^p_{w} \in B_{p-ht(z_i)+c}$ left of $I^p_{w}$ after performing this sequence of deletions.

Finally, lengthen the first $n_i$ elements (in left-right order as usual) of $\overline{B}_{p-ht(z_i)+d}$ in $\beta$, for $d = 0, 1, \ldots, ht(z_i)$.

Notation 3.6.3. Let $\mathcal{C}_{ht(z_i)}$ denote the set of lower subintervals $I^p_{u}$ of $\beta$ deliberately fixed (i.e. not lengthened) in Procedure 3.6.2. We will call $\mathcal{C}_{ht(z_i)}$ the set of deleted elements of $\beta$, or the set of fixed elements of $\beta$.

Remark 3.6.4. In Procedure 3.6.2 we say that the element $I^p_{u}$ is paired with the element $I^p_{v}$, for each $u \in [w]$. This pairing process used to obtain $\mathcal{C}_{ht(z_i)}$ is in fact the same pairing/bracketing process in the definition of the Kashiwara operator.

A rough illustration of the pairing/bracketing in Procedure 3.6.2. If we let $a$ denote a lower subinterval with head $j$ and $b$ denote a lower subinterval with head $j - 1$ (a and b are used as shorthand here; the a’s (resp. b’s) are not necessarily identical), and if we let $aaabbaababaa$
be the cascading subsequence whose lower subintervals have only \( j \) or \( j - 1 \) as head, then the pairing and deletion process in Procedure 3.6.2 works as follows.

\[ aaabbaabababa \rightarrow aa(ab)ba(ab)ba(ab)aa \rightarrow (ab)(ab)aa \rightarrow aaa \] (this means that only the remaining lower subintervals \( aaa \) can be lengthened)

That Procedure 3.6.2 works as described will be proven in Section 3.7. Meanwhile, let us look at some examples of how this procedure works.

**Example 3.6.5.** Consider the cascading sequence \( \alpha \) (with lanes marked by superscripts as usual) consisting of the lower subintervals \((8^1, 9^1, 10^1, 11^1), (8^2, 9^2, 10^2, 11^2), (8^3, 9^3, 10^3, 11^3), (7^1, 8^1, 9^1, 10^1), (7^2, 8^2, 9^2, 10^2), (8^4, 9^4, 10^4), (6^1, 7^1, 8^1, 9^1), (7^3, 8^3, 9^3), \) which is an 8*-plateau. We have

\[
\mu_7 = \begin{pmatrix}
0 & 0 \\
0 & 0
\end{pmatrix}
\]

To add two noncontributing boxes to the second stretch of \( \mu_7 \), we add two copies of 7 and two copies of 6 to \( \alpha \): \( C_1 \) in this case consists of the lower subintervals \((6^1, 7^1, 8^1, 9^1), (7^2, 8^2, 9^2, 10^2), (8^3, 9^3, 10^3, 11^3)\). The resulting cascading sequence \( \alpha' \) (where the added copies are in bold) consisting of the lower subintervals \((7^1, 8^1, 9^1, 10^1, 11^1), (7^2, 8^2, 9^2, 10^2, 11^2), (8^3, 9^3, 10^3, 11^3), (6^1, 7^1, 8^1, 9^1, 10^1), (7^3, 8^3, 9^2, 10^2), (8^4, 9^4, 10^4), (6^2, 7^2, 8^1, 9^1), (6^3, 7^3, 8^3, 9^3)\) corresponds to the resulting rigged configuration whose seventh partition is

\[
\begin{pmatrix}
-2 & 0 \\
0 & 0
\end{pmatrix}
\]

**Example 3.6.6.** Consider the cascading sequence \( \alpha \) (with lanes marked by superscripts as usual) consisting of the lower subintervals \((7^1, 8^1, 9^1, 10^1, 11^1), (8^2, 9^2, 10^2, 11^2), (8^3, 9^3, 10^3, 11^3), (6^1, 7^1, 8^1, 9^1, 10^1), (7^3, 8^3, 9^2, 10^2), (8^4, 9^4, 10^4), (6^2, 7^2, 8^1, 9^1), (7^3, 8^3, 9^3), \) which is a 7*-plateau and an \((8, 2, 3)\)-plateau. We have

\[
\mu_7 = \begin{pmatrix}
-1 & 0 \\
0 & 0
\end{pmatrix}
\]

and

\[
\mu_8 = \begin{pmatrix}
-2 & 0 \\
-1 & 0
\end{pmatrix}
\]

To add one noncontributing box to the third stretch (which has length 1, since \( |\mu_4^1| - |\mu_4^2| = 1 \)) of \( \mu_7 \), we add one copy of 7 to \( \alpha \): \( C_0 \) in this case consists of the lower subintervals \((7^2, 8^2, 9^2, 10^2), (8^3, 9^3, 10^3, 11^3), (8^4, 9^4, 10^4)\). The resulting cascading sequence \( \alpha' \) (where the added copies are in bold) consisting of the lower subintervals \((7^1, 8^1, 9^1, 10^1, 11^1), (7^2, 8^2, 9^2, 10^2, 11^2), (8^3, 9^3, 10^3, 11^3), (6^1, 7^1, 8^1, 9^1, 10^1), (7^3, 8^3, 9^2, 10^2), (8^4, 9^4, 10^4), (6^2, 7^2, 8^1, 9^1), (7^3, 8^3, 9^3)\) corresponds to the resulting rigged configuration whose seventh partition is

\[
\begin{pmatrix}
-2 & 0 \\
-1 & 0
\end{pmatrix}
\]

**Example 3.6.7.** Consider the cascading 11-sequence consisting of the lower subintervals \((8^1, 9^1, 10^1, 11^1), (8^2, 9^2, 10^2, 11^2), (8^3, 9^3, 10^3, 11^3), (8^4, 9^4, 10^4, 11^4), (7^1, 8^1, 9^1, 10^1), (7^2, 8^2, 9^2, 10^2), (7^3, 8^3, 9^3, 10^3), (5^1, 6^1, 7^1, 8^1), (6^2, 7^2, 8^2, 9^2), (6^3, 7^3, 8^3, 9^3), (7^4, 8^4, 9^4), (7^5, 8^5, 9^5)\). To add boxes to the stretch of height two, notice that \( C_2 \) consists of the lower subintervals \((5^1, 6^1, 7^1, 8^1), (6^2, 7^2, 8^2, 9^2), (7^3, 8^3, 9^3, 10^3), (8^4, 9^4, 10^4, 11^4)\). If we add three boxes to the stretch of height two, we obtain \((7^1, 8^1, 9^1, 10^1, 11^1), (7^2, 8^2, 9^2, 10^2, 11^2), (7^3, 8^3, 9^3, 10^3, 11^3), (8^4, 9^4, 10^4, 11^4), (6^1, 7^1, 8^1, 9^1, 10^1), (6^2, 7^2, 8^2, 9^2, 10^2), (7^4, 8^4, 9^4, 10^4), (8^5, 9^5, 10^5), (5^1, 6^1, 7^1, 8^1)\).
To add boxes to the stretch of height one, notice that $\mathcal{C}_1$ consists of the lower subintervals $(6^1, 7^1, 8^1, 9^1), (6^1, 7^1, 8^1, 9^1), (6^1, 7^1, 8^1, 9^1, 10^1), (6^1, 7^1, 8^1, 9^1), (6^1, 7^1, 8^1, 9^1, 10^1)$. If we add one box to the stretch of height one, we obtain $(7^1, 8^1, 9^1), (7^1, 8^1, 9^1, 10^1), (7^1, 8^1, 9^1, 10^1), (7^1, 8^1, 9^1, 10^1)$. We now show that the two procedures for adding boxes to a stretch works as described.

Procedure 3.6.8 (Adding Contributing Boxes to a Given Stretch). Suppose $ht(z_i) < r$.

1. To add $n_i$ contributing boxes beneath $z_i$, where $0 \leq n_i \leq |z_i|$: Add $n_i$ copies of the lower subinterval $(p - ht(z_i) - 1, p - ht(z_i)), \ldots, (p - 1)$ to the right of $\beta$.

2. Exceptional Case: We can add any number $N \in \mathbb{Z}_{\geq 0}$ of contributing boxes to the top row of $\mu_{p-1}$ by adding $N$ singleton lower subintervals $(p - 1)$ to the right of $\beta$.

That Procedure 3.6.8 works as described will be proven in Section 3.7.

3.7 Proof of the Procedures for Adding Boxes

We now show that the two procedures for adding boxes to a stretch works as described.

Lemma 3.7.1. Suppose $\alpha$ is a $(p, q, r)$-plateau and a $(p-1)^*$-plateau, with corresponding rigged configurations $\mathcal{R}' = (\nu_1, \ldots, \nu_{p-1}, \nu_p, \ldots, \nu_n)$. Then $\mathcal{R}'$ (and hence $\alpha$) is completely determined once we know the rigged partitions $\nu_{p-1}, \nu_p, \ldots, \nu_n$.

Proof. Follows from Lemma 3.5.2.

Remark 3.7.2. Hence, if we keep $\nu_p, \ldots, \nu_n$ fixed, the range of all possible such $\alpha$ is completely determined by the range of all possible $\nu_{p-1}$ (obtained by adding boxes to the allowed stretches of $\mathcal{R}_p$).

For any cascading sequence $\alpha$ with corresponding rigged configuration $\mathcal{R}$, let $\alpha[i]$ denote the sub-cascading sequence formed by the first $i$ lower subintervals of $\alpha$, and let $\mathcal{R}[i]$ denote the rigged configuration corresponding to $\alpha[i]$; we call $\alpha[i]$ the initial $i$-segment of $\alpha$. For any lower subinterval $I$ of $\alpha$, let $\alpha[I]$ denote the portion of $\alpha$ preceding $I$, and let $R[I]$ denote the rigged configuration corresponding to $\alpha[I]$. The following lemmas show that, if $\alpha$ is an $l$-plateau, then it has certain nice properties, which we will use in the proof of the main lemma of this section.

Lemma 3.7.3. Suppose that $\alpha$ is a cascading sequence with corresponding rigged configuration $\mathcal{R}$ such that $\mathcal{R}[i] = (\mu_1, \mu_2, \ldots, \mu_n)$ has the property that the $j$th column of $\mathcal{R}_I$ is shorter than the $j$th column of $\mu_{k-1}$ for all $j \geq b$ for some $b$. Then $\mathcal{R}[i + 1] = (\mu'_1, \mu'_2, \ldots, \mu'_n)$ has the property that the $j$th column of $\mu'_k$ is shorter than the $j$th column of $\mu'_{k-1}$ for all $j \geq c$ for some $c \geq b$.

Remark 3.7.4. This lemma is used to prove Lemma 3.7.5.

Proof. Let $I$ denote the lower subinterval after $\alpha[i]$. We may assume that $I$ has head at most $k$. If $I$ has head $k$, then it adds a box to the first row of $\mu_k$, and the conclusion is still true for $c = b$. Suppose $I$ has head less than $k$. Suppose $I$ adds a box to the $p$th column of $\mu_{k-1}$. Then it must add a box to the $q$th column of $\mu_k$, where $p \geq q$. If $q < b$, then the conclusion is still true for $c = b$. Suppose that $q \geq b$. Then the conclusion is true for $c = p$.

Lemma 3.7.5. Suppose that $\alpha$ is an $l$-plateau with corresponding rigged configuration $\mathcal{R} = (\lambda_1, \lambda_2, \ldots, \lambda_n)$. Then no lower subinterval of $\alpha$ with head less than $l$ can contain the head of some $l$-lane.

20
Proof. Suppose for a contradiction that \( \alpha \) has a lower subinterval \( J \) with head less than \( l \) containing the head of some \( l \)-lane. Without loss of generality assume that \( J \) is the leftmost such lower subinterval. Let \( \alpha[i] \) denote the portion of \( \alpha \) preceding \( J \), and let \( R[i] = (\mu_1, \mu_2, \ldots, \mu_n) \) denote the corresponding rigged configuration. By definition, \( J \) adds a box to the top row of \( \mu_1 \). By Lemma \([18]\), we have \( \mu_{l-1} \subset \mu_1 \), so \( J \) must add a box to the top row of \( \mu_{l-1} \) as well; in fact, the top rows of \( \mu_{l-1} \) and \( \mu_1 \) must be identical. Then, again by Lemma \([18]\), \( R[i+1] = (\nu_1, \nu_2, \ldots, \nu_n) \) has the property that the \( j \)-th column of \( \mathcal{R} \) is shorter than the \( j \)-th column of \( \nu_{l-1} \) for all \( j \geq m \), where \( m \) is the length of the top row of \( \nu_{l-1} \). Repeatedly applying Lemma \([3, 18] \) we conclude that the \( d \)-th column of \( \mathcal{R} \) is shorter than the \( d \)-th column of \( \lambda_{l-1} \), for some \( d \). This contradicts the \( l \)-plateau assumption on \( \alpha \).

Lemma 3.7.6. Suppose that \( \alpha \) is an \( l \)-plateau. Then the following hold:

1. \( \alpha[i] \) is also an \( l \)-plateau for any \( i \).
2. For any lower subinterval \( I \) of \( \alpha[i] \), the entries \( l-1 \) and \( l \) of \( I \) have the same lane number.

Proof. Notice that, by definition, \( I \) must contain \( l \) whenever it contains an entry less than \( l \). The two items are vacuously true if every lower subinterval of \( \alpha[i] \) has head greater than \( l \). In the base case that \( \alpha[i] \) ends in \( J \), where \( J \) is the first lower subinterval of \( \alpha \) with head at most \( l \), \( J \) must contain the head of some \( l \)-lane, so \( J \) must have head \( l \) by Lemma \([3, 7.5] \) and hence the two items hold for \( \alpha[i] \). Now we suppose that the two items hold for \( \alpha[i] \), and prove them for \( \alpha[i+1] \). Let \( I \) denote the lower subinterval following \( \alpha[i] \). If \( I \) has head at least \( l \), clearly the two items hold true (vacuously true for the second item). Suppose \( I \) has head less than \( l \). By Lemma \([3, 7.6] \) \( I \) must add a box beneath the top row of the \( l \)-th partition. Since the first item holds for \( \alpha[i] \), the \( (l-1) \)-th row is immediately the portion of the \( l \)-th partition beneath the first row, in the corresponding rigged configuration. If \( I \) adds a box to row \( r_l \) of the \( (l-1) \)-th partition (where \( r_l \) is the uppermost row of its length), then it must add a box to row \( r_l \) of the \( l \)-th partition, where \( r_l \) is the uppermost row of the \( l \)-th partition with length \( |r_l-1| \). This shows that \( \alpha[i+1] \) is still an \( l \)-plateau, and that the entries \( l-1 \) and \( l \) are the same lane number, completing the induction.

Lemma 3.7.7. Suppose that \( \alpha \) is an \( l \)-plateau. For any \( i \) and any \( j < l \), \( \alpha[i] \) has no fewer lower subintervals with head \( l \) than lower subintervals with head \( j \). We say that \( \alpha \) satisfies the Lyndon property for letter \( l \).

Remark 3.7.8. In particular, \( \alpha \) can be considered as a left Lyndon word in the letters \( l \) and \( j \).

Proof. Fix \( i \) and \( j < l \). By Lemma \([3, 7.6] \) \( \alpha[i] \) is also an \( l \)-plateau. Let \( R[i] = (\nu_1, \nu_2, \ldots, \nu_n) \) denote the rigged configuration corresponding to \( \alpha[i] \). By Lemma \([3, 2.4] \) and by the definition of an \( l \)-plateau, we have \( \nu_j \subset \nu_{j+1} \subset \ldots \subset \nu_l \). By Lemma \([3, 7.5] \) any lower subinterval that adds a box to the first row of the \( l \)-th partition must have head \( l \). Since any lower subinterval of \( \alpha[i] \) with head \( j \) adds a box to the top row of the \( j \)-th partition and any lower subinterval of \( \alpha[i] \) with head \( l \) adds a box to the top row of the \( l \)-th partition, it follows that \( \alpha[i] \) has no fewer lower subintervals with head \( l \) than lower subintervals with head \( j \).

Lemma 3.7.9. Suppose that \( \alpha \) satisfies the Lyndon property for all letters at most \( l \). If \( I \) is a lower subinterval of \( \alpha \) with head \( j \leq l \), then all entries of \( I \) at most \( l \) have the same lane number, and the depth of entry \( l' \) is \( l' - j + 1 \) for any \( j \leq l' \leq l \).

Proof. Claim is obvious for the base case of the initial segment \( \alpha[i] \) ending in the first lower subinterval with head \( l \). Now suppose that \( I \) is a lower subinterval of \( \alpha \) with head \( j \leq l \), and suppose that the claim holds for \( \alpha_i \). We show that \( I \) satisfies the desired properties. Let \( I' \)
denote the lower subinterval of $\alpha_I$ with head $j$ nearest $I$; if $I'$ does not exist, then $\alpha_I$ has no lower subinterval with head smaller than $j + 1$, so all entries of $I$ at most $l$ have lane number one and the claim follows immediately (by the inductive hypothesis, the depth of entry $c \leq l$ of $I$ is one more than the depth of entry $c$ of a lower subinterval with head $j + 1$). By the inductive hypothesis, all entries of $I'$ at most $l$ have the same lane number $k$, and the depth of entry $l'$ is $l' - j + 1$ for any $j \leq l' \leq l$. Fix $j \leq l' \leq l$. We show that entry $l'$ of $I$ has lane number $k + 1$ and depth $l' - j + 1$. Let $I''$ denote any lower subinterval of $\alpha_I$ after $I'$. By definition, $I''$ has head $j'' \neq j$. Let $j \leq d \leq l'$. If $j'' < j$, then entry $d$ of $I''$ has depth greater than that of entry $d$ of $I'$ by inductive hypothesis. If $j'' > j$, then entry $d$ of $I''$ has depth less than that of entry $d$ of $I'$ by inductive hypothesis. It follows that the number of $d$-lanes of length at least $d - j + 1$ in $\alpha_I$ is $k$; equivalently, the $d$th partition of the rigged configuration corresponding to $\alpha_I$ has exactly $k$ columns of height at least $d - j + 1$. Recall that $d - j + 1$ is the depth of entry $d$ of $I'$. Moreover, since $\alpha_I$ has more lower subintervals with head $j + 1$ than those with head $j$ by the Lyndon property, $\alpha_I$ has some $d$-lanes of length $d - (j + 1) + 1 = d - j$. Since the head of $I$ clearly has lane number $k + 1 = l'$, we deduce that the entry $d$ of $I$ must also have lane number $k + 1$ and depth $d - j + 1$, for $d = j, j + 1, \ldots, l'$ in that order, by Lemma 3.2.4. This completes the induction.

We now complete the proof of the two procedures for adding boxes. If a cascading sequence $\gamma$ has entry $g$ and lanes $L, L'$ such that $L' = L \oplus (g)$, we say that $L'$ is the lengthening of $L$, and that we lengthen $L$ to obtain $L'$.

**Lemma 3.7.10** (Main Lemma). Let $\beta$ be both a $(p - 1)^{*}$-plateau and a $(p, q, r)$-plateau, with corresponding rigged configuration $R = (\mu_1, \mu_2, \ldots, \mu_n)$. Label the stretches of $\mu_{p-1}$ from bottom to top as $z_1, z_2, \ldots, z_a$. Then the following is true.

1. Suppose $ht(z_i) < \min(q, r)$. Let $\beta^\oplus$ denote the cascading sequence obtained from $\beta$ via Procedure 3.6.8. $\beta^\oplus$ thus obtained is a $(p - 1)^{*}$-plateau and a $(p, \min(ht(z_i), r))$-plateau, and $\beta^\oplus$ corresponds to the rigged configuration obtained after adding $n_i$ noncontributing boxes beneath $z_i$, where $0 \leq n_i \leq |z_i|$, and fixing $\mu_x$ for all $x \geq p$.

2. Suppose $ht(z_i) < \min(q, r)$, and let $0 < a' \leq \min(ht(z_i))$ be an integer.

(a) There exists $j_1$ such that $\beta[j_1]$ contains all the lower subintervals with head $p - 1$ containing the head of some $p$-lane, and that $\beta[j_1]$ contains no more lower subintervals with head $p - 1$ than lower subintervals with head $p - 2$. Let such $j_1$ be minimal. Then there exists $j_2 \geq j_1$ such that $\beta[j_2]$ contains no more lower subintervals with head $p - 2$ than lower subintervals with head $p - 3$. Let such $j_2$ be minimal. Then there exists $j_3 \geq j_2$ such that $\beta[j_3]$ contains no more lower subintervals with head $p - 3$ than lower subintervals with head $p - 4$. This continues until, for minimal $j_{a' - 1}$ there exists $j_{a'} \geq j_{a' - 1}$ such that $\beta[j_{a'}]$ contains no more lower subintervals with head $p - a'$ than lower subintervals with head $p - a' - 1$.

(b) If $I$ is an element of $B_{p-h}$ outside $\beta[j_{p-(p-h)}] = \beta[j_h]$ where $1 \leq h \leq \min(ht(z_i))$, then $I$ has entry $p$ of depth $p - (p-h) + 1 = h + 1$, entry $p - 1$ of depth $p - 1 - (p-h) + 1 = h$, and one common lane number for all entries not exceeding $p$.

(c) If $\tilde{I}$ is an element of $B_{p-1}$ containing the head of some $p$-lane, then all elements of $B_p$ in $\beta[\tilde{I}]$ are elements of $C_{ht(z_i)}$.

3. Suppose $ht(z_i) < r$.

(a) Suppose $\beta^\dagger$ is the cascading sequence obtained from $\beta$ via the Procedure 3.6.8. $\beta^\dagger$ thus obtained is a $(p - 1)^{*}$-plateau and a $(p, q, \min(ht(z_i)))$-plateau, and $\beta^\dagger$ corresponds
to the rigged configuration obtained after adding \( n_i \) contributing boxes beneath \( z_i \) in \( \mu_{p-1} \), where \( 0 \leq n_i \leq |z_i| \), and fixing \( \mu_x \) for all \( x > p \) as well as fixing the shape of \( \mu_p \).

(b) Suppose \( \beta^* \) is the cascading sequence obtained from \( \beta \) via the Procedure 3.6.8. Then \( \beta^* \) thus obtained is a \( (p-1)^\ast \)-plateau and a \( (p,q,0) \)-plateau, and corresponds to the rigged configuration obtained after adding any number \( N \in \mathbb{Z}_{\geq 0} \) of contributing boxes to the top row of \( \mu_{p-1} \), and fixing \( \mu_x \) for all \( x > p \) as well as fixing the shape of \( \mu_p \).

Remark 3.7.11. Keep the following in mind for the proof that follows.

1. Item 2 is a technical fact about \( \beta \) that will be used in the proof of Procedure 3.6.2 (i.e. Item 1) in an induction argument.

2. Although adding boxes beneath \( z_i \) changes the partition, it does not change the stretches \( z_x \) for any \( x > i \), so we will continue referring to the stretches \( z_x \) even though they may well belong to a partition different from \( \mu_{p-1} \).

3. Convention: Let \( \alpha, \alpha' \) be cascading sequences with corresponding rigged configurations \( S, S' \) respectively. We say that \( \alpha[i] \) and \( \alpha'[i] \) have the same \( l \)-lanes or have identical \( l \)-lanes if the following holds: \( l \) appears as an entry in \( \alpha[i] \) the same number of times as \( l \) appears as an entry in \( \alpha'[i] \), and the \( j \)th occurrence of \( l \) in \( \alpha[i] \) has the same lane number as the \( j \)th occurrence of \( l \) in \( \alpha'[i] \). In particular, this implies that the \( l \)th partitions of \( S[i] \) and \( S'[i] \) are identical.

Proof. We prove these items by induction on \( i \) (i.e. one stretch at a time by decreasing height); observe that conditions on \( \beta \) become less restrictive with smaller \( q \) and \( r \), while the number of stretches beneath which boxes can be added decreases. In the base case where \( \beta \) is a \( p^\ast \)-plateau, Item 2 holds vacuously with \( j_1 = 0 \), by Lemma 3.7.9.

In the general case, suppose \( \beta \) is both a \( (p-1)^\ast \)-plateau and a \( (p,q,r) \)-plateau. We will prove Item 3 executability of Procedure 3.6.2 Item 1 and finally Item 2 in that order.

Proof of Item 3
We first prove Item 3 that Procedure 3.6.8 works as stated. Suppose \( \text{ht}(z_i) < r \). By Lemma 3.7.9 if \( I \) is a lower subinterval whose entry \( p-1 \) has depth \( \text{ht}(z_{i-1}) \), then \( I \in B_{p-\text{ht}(z_{i-1})} \). Since \( p-\text{ht}(z_{i-1}) < p-\text{ht}(z_i) - 1 \), and since \( \beta \) is a \( (p-1)^\ast \)-plateau, the entry \( p-1 \) of any of the added \( (p-\text{ht}(z_i) - 1, p-\text{ht}(z_i), \ldots, p-1) \) has depth not exceeding \( p-1 - (p-\text{ht}(z_{i-1})) + 1 = \text{ht}(z_{i-1}) \). Notice that \( \beta \) has \( |B_{p-\text{ht}(z_i)}| = |B_{p-\text{ht}(z_{i-1})}| + |z_i| \) and \( |B_{p-\text{ht}(z_i) - 1}| = |B_{p-\text{ht}(z_{i-1})}| \) by definition. Since \( |B_{p-\text{ht}(z_i)}| = |B_{p-\text{ht}(z_{i-1})}| + |z_i| \), and since \( \beta \) is a \( (p-1)^\ast \)-plateau, the entry \( p-1 \) of any of the added \( (p-\text{ht}(z_i) - 1, p-\text{ht}(z_i), \ldots, p-1) \) has depth exceeding \( p-1 - (p-\text{ht}(z_i)) + 1 = \text{ht}(z_i) \). By Lemma 3.2.6, the entries \( p-1 \) of the \( n_i \) added copies of \( (p-\text{ht}(z_i) - 1, p-\text{ht}(z_i), \ldots, p-1) \) must occupy \( n_i \) distinct columns, so it follows that these entries \( p-1 \) must have depth \( \text{ht}(z_i) + 1 \), as desired. Clearly, \( \beta^* \) is a \( (p-1)^\ast \)-plateau as well.

That Part 2 of this procedure works for adding boxes to the top row is obvious from Lemma 3.2.4. Finally, the property of Item 2 is clearly preserved by Procedure 3.6.8. This concludes our proof of Item 3.

Now assume that \( \text{ht}(z_i) < \text{min}(q,r) \), for which Item 2 holds for \( \beta \). We show that Items 1 and 2 hold for \( \beta^* \). Let \( B_{p-1} \) denote the set of elements of \( B_{p-1} \) containing the head of some \( p \)-lane. Let \( B^0_{p-1} \) denote the number of lower subintervals of \( \beta^* \) with head \( v \). Let \( B^0_{p-1} \) denote the set of elements of \( B^0_{p-1} \) containing the head of some \( p \)-lane.
Proof of Executability

We first prove that Procedure $3.6.2$ is executable. By Lemma $3.7.9$, it immediately follows that the deletions (recall that a deleted lower subinterval is ultimately fixed by the procedure) and lengthening specified in Procedure $3.6.2$ are executable for all pairs $B_p - j, B_{p - j - 1}$ for all $j \geq 1$. More precisely, by Lemma $3.7.9$, after deleting all the elements of $C_{ht(z_1)}$ from $B_{p - ht(z_1)}$, there will be exactly $|z_1|$ elements of $B_{p - ht(z_1)}$ remaining; after deleting all the elements of $C_{ht(z_1)}$ from $B_{p - ht(z_1) + 1}$, there will be at least $|z_1|$ elements of $B_{p - ht(z_1) + 1}$ remaining; after deleting all the elements of $C_{ht(z_1)}$ from $B_{p - ht(z_1) + 2}$, there will be at least $|z_1|$ elements of $B_{p - ht(z_1) + 2}$ remaining; and so on.

We now verify that the deletion and lengthening are executable for the pair $B_p, B_{p - 1}$. By Item $2$ and Lemma $3.7.7$, all elements of $\hat{B}_p$ are elements of $C_{ht(z_1)}$. By Lemma $3.7.9$, no element of $B$ contains the head of some $p$-lane or the head of some $(p - 1)$-lane, for any $l < p - 1$. Hence only the elements of $\hat{B}_p$ and $B_p$ contain the head of some $p$-lane. It follows that any initial segment of $\beta$ contains no fewer elements of $B_p$ than elements of $B_{p - 1} - \hat{B}_p - 1$; if an initial segment contains fewer elements of $B_p$ than elements of $B_{p - 1} - \hat{B}_p - 1$, then some of the latter elements would have to belong to $\hat{B}_p$, a contradiction.

Let $\hat{I} \in \hat{B}_p$. By Lemma $3.2.3$, $\beta_\hat{I}$ has the same number of $p$-lanes and $(p - 1)$-lanes. By Item $2$ every $I_1 \in B_p$ in $\beta_\hat{I}$ belongs to $C_{ht(z_1)}$. If $\hat{I}$ is the leftmost element of $\hat{B}_p$, then clearly every $I_1 \in B_p$ in $\beta_\hat{I}$ must be paired with an $I_1' \in B_p - 1 - \hat{B}_p - 1$ in $\beta_\hat{I}$. In general, suppose that $\hat{I} \in \hat{B}_p - 1$ precedes $\hat{I}$ in $\beta$, where every $I_1 \in B_p$ in $\beta_\hat{I}$ is paired with an $I_1' \in B_p - 1 - \hat{B}_p - 1$ in $\beta_\hat{I}$. Since $\beta_\hat{I}$ has the same number of $p$-lanes and $(p - 1)$-lanes, and since no elements of $\hat{B}_p - 1$ exist after $\hat{I}$ in $\beta_\hat{I}$, $B_p$ has the same number of elements after $\hat{I}$ in $\beta_\hat{I}$ as those of $B_p - 1 - \hat{I}$ after $\hat{I}$ in $\beta_\hat{I}$. Hence every $I_1 \in B_p$ after $\hat{I}$ in $\beta_\hat{I}$ must be paired with an $I_1' \in B_p - 1 - \hat{B}_p - 1$ after $\hat{I}$ in $\beta_\hat{I}$. Therefore, inductively we conclude that every element of $B_p \cap C_{ht(z_1)}$ must be paired with exactly one element of $B_{p - 1} - \hat{B}_p - 1$. This shows that, after deleting all of the elements of $C_{ht(z_1)}$ from $B_p$, there are at least $|z_1|$ elements remaining in $B_p$ that can be lengthened, because $B_{p - 1} - \hat{B}_p \supset B_{p - 1} - C_{ht(z_1)}$. This shows that Procedure $3.6.2$ is executable.

$\beta_\hat{I}$ satisfies the Lyndon property

We first show that $\beta_\hat{I}$ satisfies the Lyndon property for all letters $l \leq p - 1$, so that we can apply Lemma $3.7.9$ to $\beta_\hat{I}$ later on. Since $\beta_\hat{I}$ clearly satisfies the Lyndon property for all letters $l \leq p - ht(z_1) - 1$, we only need to consider $l > p - ht(z_1) - 1$. Let $m$ be any positive integer. We analyze $\beta_\hat{I}[m]$ using $\beta[m]$. Suppose $p - ht(z_1) - 1 < l' \leq p - 1$. By Lemma $3.7.7$, $\beta[m]$ has no fewer elements of $B_{l'}$ than elements of $B_{l' - 1}$.

Claim 3.7.12. $\beta[m]$ has no fewer elements of $B_{l'} \cap C_{ht(z_1)}$ than elements of $B_{l' - 1} - C_{ht(z_1)}$.

Proof. Suppose not. Then we must have $|B_{l'} \cap C_{ht(z_1)}| > |B_{l' - 1} \cap C_{ht(z_1)}|$ in $\beta[m]$. Pick the minimal $m'$ such that $|B_{l'} \cap C_{ht(z_1)}| = |B_{l' - 1} \cap C_{ht(z_1)}|$ in $\beta[m + m']$. By definition, any element of $B_{l'}$ in $\beta[m + m']$ outside $\beta[m]$ must be an element of $C_{ht(z_1)}$ and must be paired with an element of $B_{l' - 1} \cap C_{ht(z_1)}$ to its right in $\beta[m + m']$, and $\beta[m + m']$ must end in an element of $B_{l' - 1} \cap C_{ht(z_1)}$. Notice that $|B_{l'} \cap C_{ht(z_1)}| = |B_{l' - 1} \cap C_{ht(z_1)}|$ in $\beta[m + m']$, but $|B_{l'} - C_{ht(z_1)}| < |B_{l' - 1} - C_{ht(z_1)}|$ in $\beta[m + m']$ due to the assumption that $|B_{l'} - C_{ht(z_1)}| < |B_{l' - 1} - C_{ht(z_1)}|$ in $\beta[m]$. This implies that $\beta[m + m']$ has fewer elements of $B_{l'}$ than elements of $B_{l' - 1}$, which contradicts the Lyndon property for $\beta$, and the claim is proved.

Suppose $l' = p - 1$. By Item $2$ we have $\hat{B}_p \subseteq C_{ht(z_1)}$, so by definition only elements of $B_{p - 1} - C_{ht(z_1)} \subseteq B_{p - 1} - \hat{B}_p - 1$ can be lengthened by Procedure $3.6.2$. By Lemma $3.2.3$ and Lemma $3.7.9$ any lower subinterval of $\beta$ whose entry $p$ has depth one must have either $p$ or
\( p - 1 \) as head. It follows that \( \beta[j] \) has no fewer elements of \( B_p \) than elements of \( B_{p-1} - \widehat{B}_{p-1} \) for any \( j \).

Claim 3.7.13. \( \beta[m] \) has no fewer elements of \( B_p - C_{ht(z)} \) than elements of \( B_{p-1} - C_{ht(z)} \).

Proof. Suppose not. Then we must have \( |B_p \cap C_{ht(z)}| > |(B_{p-1} - \widehat{B}_{p-1}) \cap C_{ht(z)}| \) in \( \beta[m] \). Pick the minimal \( b' \) such that \( |B_p \cap C_{ht(z)}| = |(B_{p-1} - \widehat{B}_{p-1}) \cap C_{ht(z)}| \) in \( \beta[m + b'] \). By Item 2 all of \( \widehat{B}_{p-1} \) is contained inside \( \beta[j_1] \), and \( \beta[j_1] \) contains no elements of \( B_{p-1} - C_{ht(z)} \). Thus, we must have \( j_1 < m \), and there are no elements of \( \widehat{B}_{p-1} \) outside \( \beta[m] \). By definition, any element of \( B_p \) in \( \beta[m + b'] \) outside \( \beta[m] \) must be an element of \( C_{ht(z)} \) and must be paired with an element of \( B_{p-1} \cap C_{ht(z)} \) to its right in \( \beta[m + b'] \). Notice that \( |B_p \cap C_{ht(z)}| = |B_{p-1} - \widehat{B}_{p-1}) \cap C_{ht(z)}| \) in \( \beta[m + b'] \) but \( |B_p - C_{ht(z)}| < |B_{p-1} - \widehat{B}_{p-1}) - C_{ht(z)}| \) in \( \beta[m + b'] \) due to the assumption that \( |B_p - C_{ht(z)}| < |B_{p-1} - C_{ht(z)}| \) in \( \beta[m] \). This implies that \( |B_p| < |B_{p-1} - \widehat{B}_{p-1})| \) in \( \beta[m + b'] \), which is a contradiction, and the claim is proved.

Now let \( p - h(t(z))_1 - l'' \leq p - 1 \). By Lemma 3.7.7 and definition of pairing, \( |B_{p'} \cap C_{ht(z)}| \geq |B_{p'} \cap C_{ht(z)}| \) in \( \beta[m] \). By Claim 3.7.12 for \( l'' < p - 1 \) or Claim 3.7.13 for \( l'' = p - 1 \), we have \( |B_{p' + 1} \cap C_{ht(z)}| \geq |B_{p' + 1} - C_{ht(z)}| \) and \( |B_{p'} - C_{ht(z)}| \geq |B_{p' - 1} - C_{ht(z)}| \) in \( \beta[m] \). After running Procedure 3.6.2 we compare \( B_{p'} \) in \( \beta[m] \) with \( B_{p'}^\alpha \) in \( \beta[m] \) and \( B_{p' - 1} \) in \( \beta[m] \) with \( B_{p'}^\alpha \) in \( \beta[m] \). If \( |B_{p' - 1} - C_{ht(z)}| \geq n_i \) in \( \beta[m] \), then Procedure 3.6.2 forms \( \beta^\alpha[m] \) by lengthening \( n_i \) elements of \( B_{p' + 1} \cap C_{ht(z)} \), \( n_i \) elements of \( B_{p'} - C_{ht(z)} \), and \( n_i \) elements of \( B_{p' - 1} - C_{ht(z)} \) in \( \beta[m] \), and thus \( B_{p'} \) in \( \beta[m] \), \( B_{p'}^\alpha \) in \( \beta^\alpha[m] \) are equinumerous and \( B_{p' - 1} \) in \( \beta[m] \), \( B_{p'}^\alpha \) in \( \beta^\alpha[m] \) are equinumerous. If \( |B_{p' - 1} - C_{ht(z)}| < n_i \) in \( \beta[m] \), then Procedure 3.6.2 forms \( \beta^\alpha[m] \) by lengthening \( a_1 \) elements of \( B_{p' + 1} - C_{ht(z)} \), \( a_2 \) elements of \( B_{p'} - C_{ht(z)} \), and all elements of \( B_{p' - 1} - C_{ht(z)} \) in \( \beta[m] \), where \( a_2 \leq a_1 \). Therefore, in all cases we have \( |B_{p'}^\alpha| \geq |B_{p'}^\alpha| \) in \( \beta^\alpha[m] \). Since \( m \) was arbitrary, this completes the proof that \( \beta^\alpha \) satisfies the Lyndon property for all letters \( p - p - 1 \).

Proof of Item 7

We show inductively that \( \beta[\beta] \) and \( \beta[\beta] \) are identical for all \( \beta[p] \), by comparing \( \beta \) and \( \beta[\beta] \) one lower subinterval at a time, from left to right. In this case, given a lower subinterval or a portion of \( \beta \), it will be obvious what we mean by the corresponding lower subinterval or corresponding portion of \( \beta[\beta] \), and vice versa.

Let \( G_1 \) denote the first lower subinterval of \( \beta \) to be lengthened, and let \( g_1 \) be the lane number of the head \( \beta[p] \) of \( G_1 \). By definition, the lanes of \( \beta[G_1] \) are identical to those of \( \beta[G_1] \). To determine the number of \( (p - 1) \)-lanes in \( \beta[G_1] \), we need only determine the number of elements of \( B_{p - 1} \) in \( \beta[G_1] \), since \( \beta[G_1] \) and \( \beta[G_1] \) are identical, and since \( \beta[G_1] \) is a \( (p - 1) \)-plateau. By Lemma 3.2.4 any lower subinterval of \( \beta[G_1] \) whose entry \( p \) has depth one must have either \( p \) or \( p - 1 \) as head, since \( \beta[G_1] \) is a \( (p - 1) \)-plateau. Let \( J \) denote the rightmost element of \( \widehat{B}_{p - 1} \) in \( \beta[G_1] \). The entries \( p - 1, p \) of \( J \) have the same lane number, by Lemma 3.2.4. All elements of \( B_p \) in \( \beta[G_1] \) right of \( J \) must be elements of \( C_{ht(z)} \) by definition, so they must be paired with the same number of elements of \( B_{p - 1} \cap C_{ht(z)} \) in \( \beta[G_1] \) (which by definition do not contain the head of any \( p \)-lane). Since \( \beta[G_1] \) has the same number \( g_1 - 1 \) of elements of \( B_{p - 1} \) as those of \( B_p \), it follows that \( (G_1)_1 \) has entry \( p - 1 \) with lane number \( g_1 \), so its entry \( p \) has lane number \( g_1 \) as well.

Let \( G_d \) be a lower subinterval of \( \beta[\beta] \) and let \( G_d' \) denote the lower subinterval of \( \beta \) corresponding to \( G_d \). For our inductive hypothesis suppose that the \( l \)-lanes of \( \beta[G_d] \) are identical to those of \( \beta[G_d'] \) for all \( \beta \leq p \). We show that the action of \( G_d \) preserves this property; in fact, it suffices to show that entry \( p \) has the same lane number in \( G_d \) and \( G_d' \). We treat separately the case that
G'_d, G_d are identical and the case that G_d = (G'_d)_+. Recall that both β and β^α contain a copy of C_{ht(z_i)}, the context will make it clear which copy we refer to.

Consider the case that G_d = (G'_d)_+ and min G'_d = p. Let g'_d be the lane number of the head p of G'_d. Since β^α satisfies the Lyndon property, β^α_{G'_d} is a (p - 1)^*-plateau by Lemma 3.7.13, so the number of (p - 1)-lanes in β^α_{G'_d} is the number of elements of B_{p-1} in β^α_{G'_d}. By Lemma 3.2.4 any lower subinterval of β^α_{G_d}, whose entry p has depth one must have either p or p - 1 as head, since β^α_{G_d} is a (p - 1)^*-plateau. Let J_d denote the rightmost element of Ć_{p-1} in β^α_{G_d}. The entries p - 1, p of J_d have the same lane number, by Lemma 3.2.4. All entries of B^α in β^α_{G_d} right of J_d must be elements of C_{ht(z_i)} by definition (since they were fixed by Procedure 3.6.2), so they must be paired with the same number of elements of B_{p-1} \cap C_{ht(z_i)} in β^α_{G_d} (which by definition do not contain the head of any p-lane). Since β^α_{G_d} has the same number g'_d - 1 of (p - 1)-lanes as p-lanes, it follows that G_d has entry p - 1 with lane number g'_d, so its entry p has lane number g_d as well by the inductive hypothesis.

Consider the case that G_d = (G'_d)_+ and min G'_d ≤ p - 1. Let g_d be the lane number of the head of G_d. Since β^α satisfies the Lyndon property, β^α_{G_d} is a (p - 1)^*-plateau by Lemma 3.7.13, so the number of (min G'_d - 1)-lanes in β^α_{G_d} is the number of elements of B_{min G'_d} in β^α_{G_d}. Similarly, the number of (min G'_d) -lanes in β_{G_d'} is the number of elements of B_{min G'_d} in β_{G_d'}.

The elements of B_{min G'_d} \cap C_{ht(z_i)} in β_{G_d'} are lengthened by Procedure 3.6.2, while the elements of B_{min G'_d} \cap C_{ht(z_i)} in β^α_{G_d} are paired with the same number of elements of B_{min G'_d} \cap C_{ht(z_i)} in β^α_{G_d} by definition. In addition, all elements of B_{min G'_d-1} \cap C_{ht(z_i)} in β_{G_d'} are lengthened by Procedure 3.6.2 by Claim 3.6.2, since |B_{min G'_d} \cap C_{ht(z_i)}| < n_i in β_{G_d'}. Since β_{G_d'} has the same number g_d - 1 of elements of B^α_{min G'_d-1} as elements of B_{min G'_d} in β_{G_d'}, it follows that G_d has entry min G_d = min G'_d - 1 with lane number g_d, so its entry min G'_d has lane number g_d as well by the inductive hypothesis.

Consider the case that G'_d, G_d are identical and min G_d < p - ht(z_i) - 1. Since Procedure 3.6.2 lengthens no lower subintervals with head smaller than p - ht(z_i), the (min G_d)-lanes are identical in β^α_{G'_d} and β_{G_d'}. Thus, the entry min G_d has the same lane number h_d in G_d and G'_d. By Lemma 3.7.13 all entries of G_d not exceeding p - 1 have lane number h_d, and all entries of G'_d not exceeding p - 1 have lane number h_d. Since the p-lanes are identical in β^α_{G'_d} and β_{G_d'} by the inductive hypothesis, entry p has the same lane number in G_d and G'_d.

Consider the case that G'_d, G_d are identical and min G_d ≥ p - ht(z_i) - 1. By the inductive hypothesis, the l-lanes are identical in β^α_{G'_d} and β_{G_d'} for all l ≥ p, so we only need to consider the case min G_d ≤ p - 1. We now apply Item 2 with α = ht(z_i).

If min G'_d = p - 1, then entry p of G'_d must have depth one or two, by Lemma 3.2.4. Suppose that min G'_d = p - 1 and entry p has depth one. Then G_d must lie inside β[p-min G_d], and by Item 2 no element of B_p \cup B_{p-1} in β_{G_d'} can be lengthened by Procedure 3.6.2, so the entry p - 1 has the same lane number in G_d and G'_d, and hence the entry p also has the same lane number in G_d and G'_d by the inductive hypothesis.

Suppose that min G'_d = p - 1 and entry p has depth two. Then β_{G'_d} must have fewer (p - 1)-lanes than p-lanes. By Claim 3.7.13 |B_p - C_{ht(z_i)}| ≥ |B_{p-1} - Ć_{ht(z_i)}| in β_{G_d'}. If |B_{p-1} - C_{ht(z_i)}| > n_i in β_{G_d'}, then Procedure 3.6.2 lengthens n_i elements of B_p and n_i elements of B_{p-1} in β_{G_d'}, so the number of (p - 1)-lanes in β^α_{G_d} equals that of (p - 1)-lanes in β_{G_d'} and remains less than that of p-lanes in β^α_{G_d}, and hence entry p of G_d has depth two. In the case |B_{p-1} - C_{ht(z_i)}| < n_i in β_{G_d'}, we must have G'_d \in C_{ht(z_i)}. If G'_d \in Ć_{p-1}, then no element of B_p in β_{G_d'} is lengthened, by Item 2 so the number of (p - 1)-lanes does not increase. Suppose G'_d \in (B_{p-1} \cap C_{ht(z_i)}) - Ć_{p-1}. Since each I \in Ć_{p-1} in β_{G_d'} contributes a new (p - 1)-lane and a new p-lane, we can exclude Ć_{p-1} from consideration. Procedure 3.6.2 fixes the elements of
$C_{ht(z)}$, and lengthens all elements of $B_{p-1} - C_{ht(z)}$ in $\beta_{C_d}$ and some elements of $B_p - C_{ht(z)}$ in $\beta_{G_d}$, and we have $|B_{p-1} \cap C_{ht(z)}| - B_{p-1} < |B_p \cap C_{ht(z)}|$ in $\beta_{G_d}$ since $G_d \subset C_{ht(z)}$ lies outside $\beta_{G_d}$. It follows that the number of $(p-1)$-lanes in $\beta_{G_d}$ is again less than that of $p$-lanes in $\beta_{G_d}$, and hence entry $p$ of $G_d$ has depth two. Thus, in both cases entry $p$ of $G_d$ has depth two, so it has the same lane number in $G_d$ and $G_d'$ by the inductive hypothesis.

Suppose $\min G_d' < p - 1$ and all elements of $B_{\min G_d' + 1}$ in $\beta_{G_d'}$ belong to $C_{ht(z)}$. Then all elements of $B_{\min G_d'}$ in $\beta_{G_d'}$ must belong to $C_{ht(z)}$ as well, by Claim 3.7.12. It follows that no element of $B_{\min G_d' + 1}$ in $\beta_{G_d'}$ can be lengthened by Procedure 3.6.2 and no element of $B_{\min G_d'}$ in $\beta_{G_d'}$ can be lengthened by Procedure 3.6.2, so the number of elements of $B_{\min G_d'}$ is the same in $\beta_{G_d}$ and $\beta_{G_d'}$. Thus, the entries $\min G_d' \leq l \leq p - 1$ have the same lane number $k_l$ in $G_d$ and $G_d'$ by Lemma 3.7.9, so the entry $p$ also has the same lane number in $G_d$ and $G_d'$ by the inductive hypothesis.

Suppose $\min G_d' < p - 1$ and $\beta_{G_d'}$ contains some element $H \subset B_{\min G_d' + 1} - C_{ht(z)}$. Let $H^*$ be the element of $B_{\min G_d' + 1}$ left of $G_d'$. By Item 2 $H$ must lie outside $\beta_{[p_0 - \min G_d' + 1]} = \beta_{[p_0 - \min G_d' - 1]}$, so $H^*$ must lie outside $\beta_{[p_0 - \min G_d' - 1]}$ as well. Applying the second part of Item 2 we know that all entries of $H$ not exceeding $p$ have the same lane number $d^*$, $H^*$ has entry $p$ of depth $p - (\min G_d' + 1) = p - \min G_d'$, and $H^*$ has entry $p - 1$ of depth $p - \min G_d' - 1$. By Lemma 3.7.9, all entries of $G_d'$ not exceeding $p - 1$ have a common lane number $l_1$, and all entries of $G_d$ not exceeding $p - 1$ have a common lane number $l_2$. By Lemma 3.7.10 we have $l_1 \leq d^*$, so $l_1 < d^* + 1$. By Lemma 3.7.10 in $G_d'$ the depth of entry $p$ exceeds that of entry $p - 1$ by at most one, meaning that entry $p$ of $G_d'$ has depth at most $(p - \min G_d') + 1$ (since $G_d'$ has entry $p - 1$ of depth $(p - \min G_d' - 1) + 1 = p - \min G_d'$ by Lemma 3.7.9). By Claim 3.7.12 Procedure 3.6.2 lengthens no fewer elements of $B_{\min G_d' + 1}$ than elements of $B_{\min G_d'}$ in $\beta_{G_d'}$. Looking at $\beta_{G_d'}$, this means that $l_2 \geq l_1$, and hence the entry $p$ of $G_d$ has depth not exceeding that of the entry $p$ of $G_d'$, by the inductive hypothesis. On the other hand, we now show that entry $p$ of $G_d$ has depth at least $p - \min G_d' + 1$. If $|B_{\min G_d'} - C_{ht(z)}| \geq n_l$ in $\beta_{G_d'}$, then Procedure 3.6.2 lengthens $n_l$ elements of $B_{\min G_d' + 1}$ and $n_l$ elements of $B_{\min G_d'}$ in $\beta_{G_d'}$, so we must have $l_2 = l_1 \leq d^*$. If $|B_{\min G_d'} - C_{ht(z)}| < n_l$ in $\beta_{G_d'}$, then $G_d' \subset C_{ht(z)}$, $|B_{\min G_d'} \cap C_{ht(z)}| < |B_{\min G_d'} + 1 \cap C_{ht(z)}|$ in $\beta_{G_d'}$ since $G_d'$ lies outside $\beta_{G_d'}$, and Procedure 3.6.2 lengthens all elements of $B_{\min G_d' - C_{ht(z)}}$, in $\beta_{G_d'}$ and some elements of $B_{\min G_d' - C_{ht(z)}}$ in $\beta_{G_d'}$; so we must have $l_2 - 1 < d^*$ as well. Thus, in both cases the entry $p$ of $G_d$ has depth exceeding $p - \min G_d'$. This shows that entry $p$ has depth exactly $p - \min G_d' + 1$ in both $G_d$ and $G_d'$, so its lane number is the same in both $G_d'$ and $G_d$ by the inductive hypothesis. This completes the proof that the $l$-lanes are identical in $\beta$ and $\beta^\alpha$ for all $l \geq p$.

Since $\beta^\alpha$ satisfies the Lyndon property, $\beta^\alpha$ is a $(p - 1)^*$-plateau. We now show that $\beta^\alpha$ is indeed obtained from $\beta$ by lengthening $n_1 \times (p - 1)$-lanes of length $ht(z)$. Any initial segment $\beta^\alpha[j]$ contains no fewer lower subintervals with head $l$ than those with head $l - 1$, for all $l \leq p - 1$. Notice that, compared to $\beta$, $\beta^\alpha$ has the same number of elements of $B_m$ for all $m > p - ht(z)$ and $m < p - ht(z) - 1$, but has $n_l$ more elements of $B_{p - ht(z)} - 1$ and $n_l$ fewer elements of $B_{p - ht(z)}$. By Lemma 3.7.9 this shows that $\beta^\alpha$ corresponds to the rigged configuration obtained after adding $n_l$ noncontributing boxes beneath $z$. It follows that in particular $\beta^\alpha$ is a $(p, ht(z), r)$-plateau.

Proof of Item 2
Finally, we prove that Item 2 holds for $\beta^\alpha$ for any integer $0 < s' < \min(ht(z), r)$. Since $\beta$ is a $(p - 1)^*$-plateau, any lower subinterval containing the head of some $p$-lane must have head $p$ or $p - 1$. By Claim 3.7.13 if $j^\alpha_1 \geq j_1$ is minimal such that $\beta^\alpha[j^\alpha_1]$ contains all $n_l$ added copies of $p - 2$, then $\beta^\alpha[j^\alpha_1]$ contains all the lower subintervals with head $p - 1$ containing the head of some $p$-lane, and $\beta^\alpha[j^\alpha_1]$ contains no more lower subintervals with head $p - 1$ than lower subintervals with head $p - 2$; by definition, any $I_1 \in B_{p-1} \cap C_{ht(z)}$ inside $\beta[j^\alpha_1]$ must be paired
with an $I_2 \in B_p \cap C_{ht(z)}$ after $I_1$ in $\beta[j_1^\hat{a}]$, so that $\beta^\hat{a}[j_1^\hat{a}]$ contains both $I_1, I_2$. By Claim \ref{claim:7.12} for all $1 < d \leq s'$, if $j_d^\hat{a} \geq \max(j_a, j_{d-1})$ is minimal such that $\beta^\hat{a}[j_d^\hat{a}]$ contains all $n_i$ added copies of $p - d - 1$, then $\beta^\hat{a}[j_d^\hat{a}]$ contains no more lower subintervals with head $p - d$ than lower subintervals with head $p - d - 1$; by definition, any $J_1 \in B_p \cap C_{ht(z)}$ inside $\beta[j_d^\hat{a}]$ must be paired with a $J_2 \in B_p \cap C_{ht(z)}$ after $J_1$ in $\beta[j_d^\hat{a}]$, so that $\beta^\hat{a}[j_d^\hat{a}]$ contains both $J_1, J_2$.

Let $I'$ be an element of $B_p \cap \beta[j_h^\hat{a}]$, where $1 \leq h \leq s'$. Denote by $I'^*$ the lower subinterval of $\beta$ corresponding to $I'$. By definition, Procedure \ref{procedure:3.6.2} must have added all $n_i$ copies of $p - h$ and all $n_i$ copies of $p - h - 1$ to $\beta[j_h^\hat{a}]$, so $I'$ must be identical to $I'^*$ (since $I'$ was not obtained by lengthening $I'^*$). Since $I'^*$ must lie outside $\beta[j_h]$, $I'^*$ has entry $p$ of depth $p - (p - h) + 1 = h$, entry $p - 1$ of depth $p - 1 - (p - h) + 1 = h$, and one common lane number $i^*$ for all entries not exceeding $p$, by Item \ref{procedure:3.6.2} for $\beta$. As already shown, $\beta[a^\downarrow]$ is a $(p - 1)^{\ast}$-plateau with identical $l$-lanes to those of $\beta$, for all $l \geq p$. Since $\beta[a^\downarrow]$ contains the same number of newly added copies of $p - h$ as newly added copies of $p - h - 1$, the elements of $B_p \cap \beta[a^\downarrow]$ must be equinumerous with the elements of $B_p \cap \beta[j_h^\hat{a}]$, so $I'^*$ has the same lane number $i^*$ as $\beta[a^\downarrow]$. By Lemma \ref{lemma:3.7.9} for $\beta[a^\downarrow]$, all entries of $I'$ not exceeding $p - 1$ have lane number $i^*$ and entry $p - 1$ has depth $h$. Entry $p$ of $I'$ also has lane number $i^*$ and has depth $h + 1$ because $\beta[a^\downarrow]$ and $\beta[a^\downarrow]$ have identical $l$-lanes.

Lastly, let $I$ denote the rightmost element of $B_p$ in $\beta$. Let $C_{a^\downarrow}$ denote the set of deleted elements of $\beta[a^\downarrow]$, in the context of applying Procedure \ref{procedure:3.6.2} to $\beta[a^\downarrow]$, where the smallest entry to be added to $\beta[a^\downarrow]$ is $p - a^\uparrow$ $\geq 1$ $\geq p - s' - 1$. By Item \ref{procedure:3.6.2} for $\beta$, no element of $B_p$ left of $I$ can be lengthened by Procedure \ref{procedure:3.6.2}. It follows that Procedure \ref{procedure:3.6.2} must lengthen the first $n_i$ elements of $B_p \cap C_{ht(z)}$ after $I$. Therefore, if $I' \in B_p$ contains the head of some $p$-lane, then all elements of $B_p$ in $\beta[a^\downarrow]$ must be elements of $C_{a^\downarrow}$; we have $B_p \cap C_{ht(z)} \subset B_p \cap C_{a^\downarrow}$, because $B_p \cap C_{a^\downarrow} \subset C_{a^\downarrow}$, $B_p \cap C_{ht(z)} \subset B_p \cap C_{a^\downarrow}$, $B_p \cap C_{ht(z)} \subset B_p \cap C_{a^\downarrow}$ for all $p - a^\uparrow$ $< b^\uparrow$ $\leq p - 1$, and $B_p \subset B_p$. This completes the proof of Item \ref{procedure:3.6.2}.

### 3.8 Growth Algorithm

Let $R' = (\mu_1, \mu_2, \ldots, \mu_n)$ be a $B(\infty)$ rigged configuration of type $A_n$. For any $d \in [n]$, label from left to right the stretches of $\mu_d$ by $\mu_d^b, \mu_d^b, \ldots, \mu_d^b$. The following lemma determines how the riggings of $\mu_d$ constrain $\mu_d-1$. If $z$ is a stretch of $\mu_j$ intersecting the $l$th column $y_l$ of $\mu_j$, we say that $z$ spans $y_l$. If $y_l$ is the $l$th column of any other partition, we also say that $z$ spans $y_l$.

**Lemma 3.8.1.** The rigging $r^b_d$ of the row $\mu_d^b$ containing $\mu_d^b$ can be written as

$$r^b_d = \sum_{i=1}^{l} -cb(\mu_d^b) + acon_{d^b},$$

where $cb(\mu_d^b)$ is the number of contributing boxes in $\mu_d^b$, and $0 \leq acon_{d^b} \leq |\mu_d^b|$ is the above contribution to $\mu_d^b$ from $\mu_d-1$; to be precise, the above contribution to $\mu_d^b$ is the number of columns of $\mu_d-1$ spanning the stretch $\mu_d^b$ that end in a contributing box.

**Proof.** Follows from Lemma \ref{lemma:3.3.9}. To be precise, the rigging of the row $\mu_d^b$ is determined by the number of contributing boxes added to the first $|\mu_d^b|$ columns of the $d$th partition and the number of contributing boxes that will be added to the corresponding columns of the $(d - 1)$th partition; the former number corresponds to the sum $\sum_{i=1}^{l} -cb(\mu_d^b)$, while the latter number corresponds to the sum $\sum_{i=1}^{l} acon_{d^b}$. \qed
Remark 3.8.2. Notice that fixing the riggings of $\mu_d$ is the same as fixing all $acon^b_i$. Once we fix $acon^b_i$ for all $i \leq l$, then the number of columns of $\mu_{d-1}$ spanned by $\mu^b_i$ ending in a contributing box is completely determined (and must equal $acon^b_i$), for all $i \leq l$. In what follows, we will always write the rigging of a given row in the form shown in Lemma 3.8.1.

Here we give an algorithm for growing all $A_n$ rigged configurations, which can be used to check inductively (starting with the last partition and going backward) whether a given tuple of rigged partitions is a legitimate $A_n$ rigged configuration. Given $\mu_d$ and all $acon^b_i$ fixed (equivalently, all riggings fixed), this algorithm produces $\mu_{d-1}$ by utilizing Procedures 3.6.2 and 3.6.8 to add noncontributing boxes and contributing boxes beneath the stretches of $\mu_d$ one stretch at a time, from left to right. We will always follow Convention 3.6.1 on labeling the stretches of $\mu_d$ and the stretches of $\mu_d$. By combining Theorem 3.3.11, Lemma 3.3.4, Lemma 3.3.10 and Lemma 3.7.10 we have the following growth algorithm characterizing all $A_n$ rigged configurations:

Theorem 3.8.3. Let $\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)$ be a tuple of rigged partitions. Then $\Lambda$ is a $B(\infty)$ rigged configuration of $A$-type if and only if $\Lambda$ satisfies the following:

1. $\lambda_n$ must consist of a single row with rigging $r_n = -|\lambda_n| + acon_n$, where $acon_n$ is an integer $0 \leq acon_n \leq |\lambda_n|$.

2. In general, $\lambda_{n-i-1}$ is determined by $\lambda_n, \lambda_{n-1}, \ldots, \lambda_{n-i}$ as follows. Label the stretches of $\lambda_{n-i}$ by $\lambda^b_{n-i}, \lambda^b_{n-i-1}, \lambda^b_{n-i-2}, \ldots, \lambda^b_{n-i-r}$, with corresponding rows $\lambda^1_{n-i}, \lambda^1_{n-i-1}, \lambda^1_{n-i-2}, \ldots, \lambda^1_{n-i-r}$ respectively. Write the rigging of row $\lambda^b_{n-i}$ as $r^b_{n-i} = \sum_{m=1}^j -bc(\lambda_{n-i})^b_m + acon^b_{n-i}$.

To begin, we have $\lambda_{n-i-1} \supset \lambda_{n-i}$. Label the stretches of $\lambda_{n-i}$ by $s^b_1, s^b_2, \ldots, s^b_j$, where $s^b_1$ is simply a copy of $\lambda^b_{n-i}$. We can now describe $\lambda_{n-i-1}$ by specifying how many boxes many boxes $\lambda_{n-i-1}$ can have beneath each stretch $s^b_j$ and what the riggings are.

Fix $j \in [k]$, let $\eta_1^j$ and $\eta_2^j$ denote the first row and second row beneath $s^b_j$ in $\lambda_{n-i-1}$, respectively. We have $|\eta_1^j| = ncb(\lambda_{n-i-1})^b_j + cb(\lambda_{n-i-1})^b_j$ and $|\eta_2^j| = cb(\lambda_{n-i-1})^b_j$, where $0 \leq ncb(\lambda_{n-i-1})^b_j \leq |s^b_j|$, $0 \leq cb(\lambda_{n-i-1})^b_j \leq ncb(\lambda_{n-i-1})^b_j$, $0 \leq cb(\lambda_{n-i-1})^b_j \leq |s^b_j|$, and $cb(\lambda_{n-i-1})^b_j = acon^b_j$.

(a) Let $\delta(\lambda_{n-i-1}) := \max ncb(\lambda_{n-i-1}) - \max (\lambda_{n-i-1})^t$. At most min(\delta(\lambda_{n-i-1}), 2) rows can exist beneath $s^b_1$ in $\lambda_{n-i-1}$. Any row of length $|\eta_1^1|$ has rigging $r^b_{n-i-1} := -bc(\lambda_{n-i-1})^b_j + acn^b_{n-i-1}$, any row of length $|\eta_1^1|$ has rigging $r^b_{n-i-1} := -cb(\lambda_{n-i-1})^b_j + acn^b_{n-i-1}$, and any row of length $|s^b_1|$ has rigging $r^b_{n-i-1} := r^b_{n-i-1} - cb(\lambda_{n-i-1})^b_j + acn^b_{n-i-1}$.

(i) For any $2 \leq m \leq k$, we now determine the rigging for the stretch $s^b_m$ given that we have already done so for $s^b_1, s^b_2, \ldots, s^b_{m-1}$. Let $N(b_m)$ be the height of the column of $\lambda_{n-i-1}$ on the left of $s^b_m$. At most min($N(b_m) - \text{ht}(s^b_m), 2$) rows can exist beneath $s^b_m$ in $\lambda_{n-i-1}$. Let $U_m$ denote the uppermost row of $\lambda_{n-i-1}$ with length at most $Y_m$, and let $r(b_m)$ denote the rigging of $U_m$. There are three cases:

i. If $|U_m| < Y_m$ and $|\eta_2^m| \neq 0$, then any row of length $Y_m + |\eta_2^m|$ has rigging $r^b_{n-i-1} = r(b_m) - bc(\lambda_{n-i-1})^b_m + acn^b_{n-i-1}$ and any row of length $Y_m + |\eta_1^m|$ has rigging $r^b_{n-i-1} = r(b_m) - cb(\lambda_{n-i-1})^b_m + acn^b_{n-i-1}$, where $0 \leq acn^b_{n-i-1} \leq Y_m + |\eta_2^m| - |U_m|$ and $0 \leq acn^b_{n-i-1} \leq |\eta_1^m| - |\eta_2^m|$. 

29
ii. If \(|U_{b_m}| < Y_m\) and \(|\eta_2^m| = 0\), then any row of length \(Y_m + |\eta_2^m|\) has rigging \(r_{b_{n-1}} = r(b_m) - cb(\lambda_{n-1})^{b_m} + acon_{b_{n-1}} \), where \(0 \leq acon_{b_{n-1}} \leq Y_m + |\eta_2^m| - |U_{b_m}| \).

iii. Otherwise, any row of length \(Y_m + |\eta_2^m|\) has rigging \(r_{b_{n-1}} = r(b_m) - cb(\lambda_{n-1})^{b_m} + acon_{b_{n-1}}\) and any row of length \(Y_m + |\eta_1^m|\) has rigging \(r_{b_{i-1}} = r(b_m) - cb(\lambda_{n-1})^{b_m} + acon_{b_{n-1}}\), where \(0 \leq acon_{b_{n-1}} \leq cb(\lambda_{n-1})^{b_m} + acon_{b_{n-1}}\) and \(0 \leq acon_{b_{i-1}} \leq |\eta_1^m| - |\eta_2^m|\).

In all cases, any row of length \(Y_m + |s^{b_m}|\) has rigging \(r_{b_{n-1}} = r(b_m) + acon_{b_{n-1}}\), where \(0 \leq acon_{n-1} \leq |s^{b_m}| - |\eta_1^m|\).

(c) Finally, to determine the first row \(\lambda_{n-1}^{b_{n-1}}\) of \(\lambda_{n-1}\), we compare \(|\eta_1^k|\) with \(|\lambda_{n-1}^{b_{n-1}}|\).

i. If \(|\eta_1^k| < |\lambda_{n-1}^{b_{n-1}}|\) and \(|\lambda_{n-1}^{b_{n-1}}| > |\lambda_{n-1}^{b_{n-1}}|\), then \(|\lambda_{n-1}^{b_{n-1}}| = |\lambda_{n-1}^{b_{n-1}}| + |\eta_1^k|\), with rigging \(r_{b_{n-1}}\).

ii. Otherwise, we have \(|\lambda_{n-1}^{b_{n-1}}| = |\lambda_{n-1}^{b_{n-1}}| + cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}}\), where \(cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}}\) can be any nonnegative integer. Let \(r(\lambda_{n-1}^{b_{n-1}})\) denote the rigging of \(\lambda_{n-1}^{b_{n-1}}\).

A. If \(|\lambda_{n-1}^{b_{n-1}}| = |\lambda_{n-1}^{b_{n-1}}|\), then \(\lambda_{n-1}^{b_{n-1}}\) has rigging \(r_{b_{n-1}} = r(\lambda_{n-1}^{b_{n-1}}) + cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}} + acon_{b_{n-1}}\), where \(0 \leq acon_{b_{n-1}} \leq |\lambda_{n-1}^{b_{n-1}}| - |\lambda_{n-1}^{b_{n-1}}|\).

B. If \(|\lambda_{n-1}^{b_{n-1}}| > |\lambda_{n-1}^{b_{n-1}}|\), then \(\lambda_{n-1}^{b_{n-1}}\) has rigging \(r_{b_{i-1}} = r(\lambda_{n-1}^{b_{n-1}}) + cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}} + cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}} + acon_{b_{n-1}}\), where \(0 \leq acon_{b_{n-1}} \leq |\lambda_{n-1}^{b_{n-1}}| - |\lambda_{n-1}^{b_{n-1}}|\).

Remark 3.8.4. Here, \(ncb(\lambda_{n-1})^{b_{n-1}}\) denotes the number of noncontributing boxes in the first row beneath \(s^{b_{n-1}}\), \(cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}}\) denotes the number of contributing boxes in the first row beneath \(s^{b_{n-1}}\), and \(cb(\lambda_{n-1}^{b_{n-1}})^{b_{n-1}}\) denotes the number of contributing boxes in the second row beneath \(s^{b_{n-1}}\).

Proof. Observe that, in short, the theorem states that the full range of boxes allowed under Theorem 3.3.11 can indeed be added to \(\lambda_{n-1}\); in other words, the constraints imposed by Theorem 3.3.11 are tight. Starting with the empty rigged configuration \(R\), the construction of \(A\) using Procedures 3.6.2 and 3.6.8 is described item by item as follows.

1. By Lemma 3.7.10, \(\lambda_n\) can be formed by using Procedure 3.6.8 to add \(|\eta_1^m|\) contributing boxes to the empty \(n\)th partition, and the resulting \((n-1)\)st partition is \(\lambda_{n-1}\).

2. The base case of \(\lambda_n\) justifies the inductive hypothesis that the \((n-1)\)st partition is \(\lambda_{n-1}\) (with riggings of zero) after the partitions \(\lambda_n, \lambda_{n-1}, \ldots, \lambda_{n-1}\), have been constructed, in that order, by Procedures 3.6.2 and 3.6.8. Lemma 3.7.10 ensures that Procedures 3.6.2 and 3.6.8 can add all the boxes Theorem 3.3.11 allows under \(s^{b_{n-1}}\). To be precise, Procedure 3.6.2 will be used to add the \(ncb(\lambda_{n-1})^{b_{n-1}}\) noncontributing boxes to the first row beneath \(s^{b_{n-1}}\), Procedure 3.6.8 will be used to add the \(cb(\lambda_{n-1})^{b_{n-1}}\) contributing boxes after these \(ncb(\lambda_{n-1})^{b_{n-1}}\) noncontributing boxes in the same row, and Procedure 3.6.8 will be used to add the \(cb(\lambda_{n-1})^{b_{n-1}}\) contributing boxes beneath these \(ncb(\lambda_{n-1})^{b_{n-1}}\) noncontributing boxes. Finally, by Lemma 3.8.1 a total of \(acon_{n-1}\) contributing boxes must be added beneath \(s^{b_{n-1}}\) to account for the positive contribution to \(\lambda_{n-1}\). In the cases (a) and (b), Theorem 3.3.11 determines how many rows can be added beneath \(s^{b_{n-1}}\) and how many contributing boxes can be added beneath the corresponding stretch of the \((n-1)\)st partition.
(a) Notice that, for either of the Procedures 3.6.2 and 3.6.8 to work, we must have 
\( p - h(t_1) + 1 \geq 1 \) or equivalently \( (p - 1) - h(t_1) > 0 \). Indeed, by Lemma 3.7.10, 
\( \min(p - 1) - h(t_1), 2) \) is the number of rows that can be added beneath the first 
stretch by Procedures 3.6.2 and 3.6.8. 

By Theorem 3.3.11 at most \( \min(\delta(n_{i-1}), 2) \) rows can exist beneath \( s^{b_1} \) in \( \lambda_{n_{i-1}} \). 
To prove the converse, let \( H_1 \leq \text{max}_r \) denote the number of rows in \( \lambda_r \). Then \( \lambda_r \) 
has \( H_{n-i-1} = H_{n-i-1} - 1 \leq \text{max}_r n_{i-1} - 1 \) rows. By above, \( \min(n - i - 1 - H_{n-i-1}, 2) = \min(n - i - H_{n-i-1}, 2) \) is exactly the number of rows allowed to be added beneath \( s^{b_1} \) by 
Procedures 3.6.2 and 3.6.8. Consider the cases \( n - i - 1 \geq \frac{n+1}{2} \) and \( n - i - 1 \leq \frac{n+1}{2} \). 

If \( n - i - 1 \leq \frac{n+1}{2} \), then \( \delta(n_{i-1}) = n - i - 1 - H_{n-i-1} \) since \( \text{max}_r n_{i-1} = n - i - 1 \). 
Suppose \( n - i - 1 > \frac{n+1}{2} \). Then \( n - i - 1 - H_{n-i-1} \geq n - i - 1 - (\text{max}_r n_{i-1} - 1) = n - i - 1 - (n - (n - i) + 1 - 1) = n - i - 1 - 3 = n - i - 3 \) by Lemma 3.3.5 since 
\( i < \frac{n+3}{2} \). Also, \( \delta(n_{i-1}) = \text{max}_r n_{i-1} - H_{n-i-1} = n - (n - i - 1) + 1 - H_{n-i-1} = i + 3 - H_{n-i} \geq i + 3 - \text{max}_r n_{i} = i + 3 - (i + 1) = 2 \) by Lemma 3.3.5. It follows 
that \( \min(\delta(n_{i-1}), 2) = 2 = \min(n - i - H_{n-i-1}, 2) \) as well. Thus, in both cases 
\( \min(\delta(n_{i-1}), 2) \) rows can indeed be added beneath \( s^{b_1} \). 

Now we prove the riggings formulas. In the case \( \delta(n_{i-1}) = 0 \), both \( \eta_1^i \) and \( \eta_2^i \) must 
be empty. Since \( \delta(n_{i-1}) = 0 \), no rows can be added beneath the first stretch of 
the \((n - i - 2)\)nd partition, which already has \( \text{max}_r n_{i-2} \) rows by Lemma 3.3.4 and 
hence we must have \( \text{acorn}_{n_{i-1}} = 0 \). The claims for the other case is obvious.

(b) By Theorem 3.3.11 at most \( \min(N(b_m) - h(s^{b_m}), 2) \) rows can exist beneath \( s^{b_m} \) in 
\( \lambda_{n_{i-1}} \); if more than \( N(b_m) - h(s^{b_m}) \) rows is added beneath \( s^{b_m} \), the result will not 
be a valid partition. The converse, that \( \min(N(b_m) - h(s^{b_m}), 2) \) rows can indeed be 
added beneath \( s^{b_m} \), follows from Lemma 3.7.10.

i. Suppose \( |U_{b_m} | < Y_m \) and \( |\eta_2^m| \neq 0 \), let \( h_1 \) denote the bottommost row of length 
\( Y_m + |\eta_2^m| \), and let \( h_2 \) denote the stretch of \( \lambda_{n_{i-1}} \) contained in \( h_1 \). Since \( h_2 \) 
contains \( cb(\lambda_{n_{i-1}})_1^{b_{m-1-1}} + cb(\lambda_{n_{i-1}})_2^{b_{m-1}} \) contributing boxes and \( |h_2| = Y_m + 
|\eta_2^m| - |U_{b_m} | \), we obtain the formula for \( r_{n_{i-1}}^{b_{m-2}} \).

ii. Suppose \( |U_{b_m} | < Y_m \) and \( |\eta_2^m| = 0 \). Then \( \lambda_{n_{i-1}} \) has no row of length \( Y_m + |\eta_2^m| \). 
Let \( g_1 \) denote the bottommost row of length \( Y_m + |\eta_2^m| \), and let \( g_2 \) denote the 
stretch of \( \lambda_{n_{i-1}} \) contained in \( g_1 \). Since \( g_2 \) contains \( cb(\lambda_{n_{i-1}})_1^{b_{m-2}} \) contributing 
boxes and \( |g_2| = Y_m + |\eta_2^m| - |U_{b_m} | \), we obtain the formula for \( r_{n_{i-1}}^{b_{m-2}} \).

iii. In the last case, Let \( t_1 \) denote the bottommost row of length \( Y_m + |\eta_2^m| \), and let 
\( t_2 \) denote the stretch of \( \lambda_{n_{i-1}} \) contained in \( t_1 \). Since \( t_2 \) contains \( cb(\lambda_{n_{i-1}})_1^{b_{m-2}} \) contributing 
boxes and \( |t_2| = cb(\lambda_{n_{i-1}})_2^{b_{m-2}} \), we obtain the formula for \( r_{n_{i-1}}^{b_{m-2}} \).

The other riggings are easily expressed in terms of the rigging of the bottommost row 
beneath \( s^{b_m} \).

(c) The first row \( \lambda_{n_{i-1}}^{b_{m-1}} \) of \( \lambda_{n_{i-1}} \) is determined as follows.

i. If \( |\eta_1^i| < |\lambda_{n_{i-1}}^1| \) and \( |\lambda_{n_{i-1}}^1| > |\lambda_{n_{i-1}}^2| \), then no more contributing boxes can be 
added after \( \eta_1^i \) in the first row, so \( \lambda_{n_{i-1}}^{b_{m-1}} = |\lambda_{n_{i-1}}^1| + |\eta_1^i| \), with rigging \( r_{n_{i-1}}^{b_{m-1}} \).

ii. If \( |\eta_1^i| = |\lambda_{n_{i-1}}^1| \) or \( |\lambda_{n_{i-1}}^1| = |\lambda_{n_{i-1}}^2| \), then any number \( cb'(\lambda_{n_{i-1}})^{b_{k}} \) of contributing 
boxes can be added after \( \eta_1^i \) in the first row without affecting the rigging of \( \lambda_{n_{i-1}} \). 
A. Since stretch \( \lambda_{n_{i-1}}^{b_{k+1}} \) contains \( cb'(\lambda_{n_{i-1}})^{b_{k}} \) contributing boxes and has length 
\( cb'(\lambda_{n_{i-1}})^{b_{k}} = |\lambda_{n_{i-1}}^1| - |\lambda_{n_{i-1}}^2| \), we obtain the formula for \( r_{n_{i-1}}^{b_{k+1}} \).
B. Since \(|b^i_1| = |\lambda_{n-i}^1|\) and stretch \(\lambda_{n-i}^1\) contains \(cb(\lambda_{n-i-1})^{b_1} + cb(\lambda_{n-i-1})^{b_2}\) contributing boxes and has length \(|\lambda_{n-i-1}^1| - |\lambda_{n-i-1}^2|\), we obtain the formula for \(a_{n-i}^{b_k} \).

**Example 3.8.5.** Let

\[
\lambda_n = \begin{array}{c}
\hline
\hline
\hline
\end{array}
\]

with rigging \(-3 + acon_n\), and fix \(acon_n = 2\). Then one possible choice of \(\lambda_{n-1}\) is

\[
\lambda_{n-1} = \begin{array}{c}
\hline
\hline
\end{array}
\]

where the bottom row has rigging \(-2 + acon_{n-1}\) and the top row has rigging \(-4 + acon_{n-1}^2\). This is obtained by adding two contributing boxes to the second row and two contributing boxes to the end of the first row (whose first three boxes are noncontributing boxes).

Suppose we now fix \(acon_{n-1}^1 = 2\) and \(acon_{n-1}^2 = 4\). Then one possible choice for \(\lambda_{n-2}\) is

\[
\lambda_{n-2} = \begin{array}{c}
\hline
\hline
\end{array}
\]

where the third row has rigging \(-2 + acon_{n-2}^1\), second row has rigging \(-3 + acon_{n-2}^1 + acon_{n-2}^2\), first row has rigging \(-5 + acon_{n-2}^1 + acon_{n-2}^2 + acon_{n-2}^3\). This choice for \(\lambda_{n-2}\) is constructed by first starting with \(\lambda_{n-2}^* = \begin{array}{c}
\hline
\hline
\end{array}\) (with rigging 0), then adding two noncontributing boxes to the second row, two contributing boxes to the third row, one noncontributing box to the first row, one contributing box to the second row, and two contributing boxes to the first row, in that order.

### 3.9 Determining the Cascading Sequence of a Rigged Configuration

Based on Theorem 3.8.3, we now give the algorithm for determining the cascading sequence of a rigged configuration. Assume \(\Lambda = (\lambda_1, \lambda_2, \ldots, \lambda_n)\) is a \(B(\infty)\) rigged configuration of \(A\)-type.

**Theorem 3.9.1.** The following algorithm constructs the cascading sequence \(\alpha\) corresponding to \(\Lambda\):

1. Start with the empty string \(\alpha^0\). Add \(|\lambda_n|\) copies of lower subintervals \((n)\) to \(\alpha^0\), obtaining \(\alpha^1\), which accounts for \(\lambda_n\).

2. In general, suppose that we have constructed the cascading sequence \(\alpha^i\) which accounts for \(\lambda_n, \lambda_{n-1}, \ldots, \lambda_{n-i}\). We want to construct \(\alpha^{i+1}\) that accounts for \(\lambda_n, \lambda_{n-1}, \ldots, \lambda_{n-i}, \lambda_{n-i-1}\).

   Label the stretches of \(\lambda_{n-i}\) by \(\tilde{\lambda}^b_{n-i}, \tilde{\lambda}^{b_2}_{n-i}, \ldots, \tilde{\lambda}^{b_k}_{n-i}\), with corresponding rows \(\lambda^{b_1}_{n-i}, \lambda^{b_2}_{n-i}, \ldots, \lambda^{b_k}_{n-i}\) respectively. Write the rigging of row \(\lambda^{b_j}_{n-i}\) as

   \[
   r^{b_j}_{n-i} = \sum_{m=1}^{j} -cb(\lambda_{n-i})^{b_m} + acon_{n-i}^{b_m}.
   \]

   Label the stretches of \(\lambda_{n-i}\) by \(s^{b_1}, s^{b_2}, \ldots, s^{b_k}\). Let \(w^{b_1}, w^{b_2}, \ldots, w^{b_k}\) denote the stretches of the copy of \(\lambda_{n-i}\) sitting inside \(\lambda_{n-i-1}\).
(a) For \( m = 1, 2, \ldots, k \), let \( l_m \) denote the number of boxes in the second row beneath \( w_{bm} \) and let \( l_m \leq u_m \leq |w_{bm}| \) denote the number of boxes in the first row beneath \( w_{bm} \). For \( m \) ranging through \( 1, 2, \ldots, k \) in that order, first apply Procedure 3.6.2 to add \( l_m + u_m - \text{acon}_{bm} \) noncontributing boxes beneath \( s_{bm} \), then apply Procedure 3.6.8 to add \( l_m \) contributing boxes beneath these added noncontributing boxes, and finally apply Procedure 3.6.8 to add \( \text{acon}_{bm} - l_m \) contributing boxes to the first row beneath \( s_{bm} \), updating the cascading sequence (starting from \( \alpha^1 \)) with each application of each procedure.

(b) Suppose we have added all the boxes required beneath the stretches of \( \lambda_{n-i} \). Let \( g \) denote the resulting first row. Apply Procedure 3.6.8 to add \( |\lambda_{n-i-1}^1| - |g| \) contributing boxes to \( g \), updating the cascading sequence. This completes the construction of \( \lambda_{n-i-1} \), and the resulting cascading sequence is the desired \( \alpha^{i+1} \).

Proof. By assumption, \( \Lambda \) is a legitimate rigged configuration. This algorithm works by comparing \( \Lambda \) with the rigged configuration corresponding to the cascading sequence constructed so far, seeing what boxes need to be added to construct the next partition of \( \Lambda \), and then applying Procedure 3.6.2 and Procedure 3.6.8 to add the boxes required. The full proof is similar to that of Theorem 3.8.3 and is a matter of bookkeeping.

Now let us look at some examples of how to obtain the cascading sequence given a rigged configuration using the algorithm described above.

Example 3.9.2. Consider the following \( A_{10} \) rigged configuration \( R = (\nu_1, \nu_2, \ldots, \nu_{10}) \) (in top-bottom order) where \( \nu_i \) is the \( i \)th rigged partition whose \( j \)th row has rigging \( r_{ij} \):

```
0 0 0 0
0 0 0 0
-1 0 0 0
0 0 0 0
-2 0 0 0
-3 0 0 0
-5 -4 -4 -4
-3 -3 -3 -3
1 0 0 0
-1 -1 -1 -1
```

From the viewpoint of its cascading sequence, \( R \) is constructed (by the growth algorithm) in the following process (where newly added letters or lower subintervals at each stage are marked
with a prime (‘)):

\[
(10')(10')(10')(10') \rightarrow \mathbb{1} (9', 10)(9', 10)(9', 10)(9', 10)
\]
\[
\rightarrow \mathbb{2} (9, 10)(9, 10)(9, 10)(9, 10)(8, 9)'(8, 9)'(8, 9)
\]
\[
\rightarrow \mathbb{3} (8', 9, 10)(8', 9, 10)(8', 9, 10)(9, 10)(7', 8, 9)(7', 8, 9)(7', 8, 9)
\]
\[
\rightarrow \mathbb{4} (8, 9, 10)(8, 9, 10)(8, 9, 10)(9, 10)(7, 8, 9)(7, 8, 9)(7, 8, 9)
\]
\[
(6, 7, 8)'(6, 7, 8)'(6, 7, 8)'(6, 7, 8)
\]
\[
\rightarrow \mathbb{5} (8, 9, 10)(8, 9, 10)(8, 9, 10)(8', 9, 10)(7, 8, 9)(7, 8, 9)(7, 8, 9)
\]
\[
(6, 7, 8)(6, 7, 8)(6, 7, 8)
\]
\[
\rightarrow \mathbb{6} (8, 9, 10)(8, 9, 10)(8, 9, 10)(8, 9, 10)(7, 8, 9)(7, 8, 9)(7, 8, 9)
\]
\[
(6, 7, 8)(6, 7, 8)(6, 7, 8)(6, 7, 8)
\]
\[
\rightarrow \mathbb{7} (7', 8, 9, 10)(7', 8, 9, 10)(7', 8, 9, 10)(7', 8, 9, 10)(6', 7, 8, 9)
\]
\[
(6', 7, 8, 9)(6', 7, 8, 9)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(6', 7, 8)
\]
\[
\rightarrow \mathbb{8} (7, 8, 9, 10)(7, 8, 9, 10)(7, 8, 9, 10)(7, 8, 9, 10)(6, 7, 8, 9)
\]
\[
(6, 7, 8, 9)(6, 7, 8, 9)(5, 6, 7, 8)(5, 6, 7, 8)(5, 6, 7, 8)(6', 7, 8)
\]

**Explanation of the above process:** We started out by adding four 10-boxes, which completes Partition 10. Since \( \text{rig}_{10}^3 = -1 = -4 + 3 \), we first added four noncontributing 9-boxes in \( \mathbb{1} \), and then added three contributing 9-boxes beneath these noncontributing boxes, which completes Partition 9 and adds three noncontributing 8-boxes. Since \( \text{rig}_{9}^2 = 0 = -3 + 3 \) and \( \text{rig}_{9}^1 = 1 = -3 + 4 \), we first added three noncontributing 8-boxes beneath the first row in \( \mathbb{3} \) (along with three noncontributing 7-boxes), and then added three contributing 8-boxes beneath the second row in \( \mathbb{4} \) (along with three noncontributing 7-boxes and three noncontributing 6-boxes), and then added one noncontributing 8-box to the first row in \( \mathbb{5} \), and then added one contributing 8-box beneath the first row in \( \mathbb{6} \) (along with one noncontributing 7-box to the first row). This completes Partition 8. Now, Partitions 5-7 all have zero riggings, while the remaining partitions are empty. To complete Partition 7, we added three noncontributing 7-boxes beneath the second row in \( \mathbb{7} \) (along with three noncontributing 6-boxes to the second row and three noncontributing 5-boxes to the first row). Finally, we added one noncontributing 6-box to the first row in \( \mathbb{8} \) to complete Partition 6. This gives us the desired rigged configuration.

**Example 3.9.3.** Consider the following \( A_{10} \) rigged configuration \( S = (\nu_1, \nu_2, \ldots, \nu_{10}) \) (in top-bottom order) where \( \nu_i \) is the \( i \)th rigged partition whose \( j \)th row has rigging \( \text{rig}_{j}^{i} \):
From the viewpoint of cascading sequences, $S$ is constructed in the following process:

$$(10')(10')(10')(10') \rightarrow (9', 10)(9', 10)(9', 10)(9', 10)$$

$$(9', 10)(9', 10)(9', 10)(9', 10)(8', 9')(8', 9')$$

$$(8', 9')(8', 9')(8', 9')(8', 9')(9, 10)(9, 10)(7', 8, 9)(7', 8, 9)(7', 8, 9)$$

$$(6, 7, 8')(6, 7, 8')(6, 7, 8')(6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)(5', 6, 7, 8)$$

Explanation of the above process:

We started out by adding four 10-boxes, which completes Partition 10. Since $\text{rig}_{10}^1 = -1$ =
−4 + 3, we first added four noncontributing 9-boxes in (1), and then added three contributing 9-boxes in (2) beneath these noncontributing boxes, which completes Partition 9 and adds three noncontributing 8-boxes. Since \( \text{rig}_9^3 = 0 = -3 + 3 \) and \( \text{rig}_9^1 = 1 = -3 + 4 \), we first added three noncontributing 8-boxes beneath the first row in (3) (along with three noncontributing 7-boxes), and then added three contributing 8-boxes beneath the second row in (4) (along with three noncontributing 7-boxes and three noncontributing 6-boxes), and then added one noncontributing 8-box to the first row in (5), and then added one contributing 8-box beneath the first row in (6) (along with one noncontributing 7-box to the first row). This completes Partition 8. Since \( \text{rig}_8^3 = -3 + 0 \) and \( \text{rig}_8^1 = \text{rig}_8^2 = -4 + 0 \), there are no contributing 7-boxes to add. In (7), we added two noncontributing 7-boxes to the third row. In (8), we added a noncontributing 7-box to the second row. In (9), we added a noncontributing 6-box to the second row. This completes Partition 6, and yields the desired rigged configuration.

4 Further Discussions

One can try to characterize \( \mathcal{B}(\infty) \) rigged configurations in the types \( B, C, D, G \), by modifying or extending the methods used in this paper. One can also try to find a non-recursive characterization of \( \mathcal{B}(\infty) \) rigged configurations, which describes the \( i \)th rigged partition without reference to the \( (i + 1) \)st partition.

References

[1] A. Berenstein, A. Zelevinsky: String bases for quantum groups of type \( A_r \). Adv. Soviet Math. 16, Part 1, 51–89 (1993)

[2] A. Berenstein, A. Zelevinsky: Canonical bases for the quantum group of type \( A_r \) and piecewise-linear combinatorics. Duke Math. J. 82 (3), 473–502 (1996)

[3] Jin Hong and Hyeonmi Lee: Young tableaux and crystal \( \mathcal{B}(\infty) \) for finite simple Lie algebras. Journal of Algebra 320, 3680–3693 (2008)

[4] Jin Hong, Hyeonmi Lee, and Roger Tian: Rigged Configuration Descriptions of the Crystals \( \mathcal{B}(\infty) \) and \( \mathcal{B}(\lambda) \) for Special Linear Lie Algebras. Preprint. arXiv:1604.04357

[5] M. Kashiwara: Crystalizing the \( q \)-analogue of universal enveloping algebras. Comm. Math. Phys. 133, no. 2, 249–260 (1990)

[6] P. Littelmann: Crystal graphs and Young tableaux. J. Algebra 175, no. 1, 65–87 (1995)

[7] George Lusztig: Canonical bases arising from quantized enveloping algebras. J. Amer. Math. Soc. 3, no. 2, 447–498 (1990)

[8] The Sage-Combinat community, Sage-Combinat: enhancing Sage as a toolbox for computer exploration in algebraic combinatorics, http://combinat.sagemath.org, 2008.

[9] SageMath, the Sage Mathematics Software System (Version 6.2.1), The Sage Developers, 2015, http://www.sagemath.org.

[10] Ben Salisbury and Travis Scrimshaw: A rigged configuration model for \( \mathcal{B}(\infty) \). J. Combin. Theory Ser. A 133, 29–57 (2015)
[11] Ben Salisbury and Travis Scrimshaw: *Connecting marginally large tableaux and rigged configurations via crystals*. Preprint. arXiv:1505.07040.

[12] Anne Schilling: *Crystal structure on rigged configurations*. Int. Math. Res. Not. IMRN 27, Art. ID 97376 (2006)