BOX-BALL SYSTEMS AND RSK RECORDING TABLEAUX

MARI\-SA COFIE, OLIVIA FUGIKAWA, EMILY GUNAWAN, MADELYN STEWART, AND DAVID ZENG

Abstract. A box-ball system (BBS) is a discrete dynamical system consisting of \( n \) balls in an infinite strip of boxes. During each BBS move, the balls take turns jumping to the first empty box, beginning with the smallest-numbered ball. The one-line notation of a permutation can be used to define a BBS state. This paper proves that the Robinson–Schensted (RS) recording tableau of a permutation completely determines the dynamics of the box-ball system containing the permutation.

Every box-ball system eventually reaches steady state, decomposing into solitons. We prove that the rightmost soliton is equal to the first row of the RS insertion tableau and it is formed after at most one BBS move. This fact helps us compute the number of BBS moves required to form the rest of the solitons. First, we prove that if a permutation has an L-shaped soliton decomposition then it reaches steady state after at most one BBS move. Permutations with L-shaped soliton decompositions include noncrossing involutions and column reading words. Second, we make partial progress on the conjecture that every permutation on \( n \) objects reaches steady state after at most \( n - 3 \) BBS moves. Furthermore, we study the permutations whose soliton decompositions are standard; we conjecture that they are closed under consecutive pattern containment and that the RS recording tableaux belonging to such permutations are counted by the Motzkin numbers.

1. Introduction

![Figure 1. One BBS move from ...ee452361eeee... to ...eeee45e2136e...](image)

1.1. Box-ball systems. The box-ball system\(^1\), or BBS for short, is a dynamical system consisting of discrete time states. At each time state, we have finitely many numbered balls

\(^1\)Our version of the box-ball system was introduced in [Tak93] and is an extension of the box-ball system first invented by Takahashi and Satsuma in [TS90].

1
in an infinite strip of boxes; the boxes are indexed by the integers from left to right, and each box can fit at most one ball. One BBS move is the process of letting each ball jump to the nearest empty box to its right, starting with the smallest-numbered ball (see Figure 1). Given a BBS state at time \( t \), we compute the BBS state at time \( t + 1 \) by applying one BBS move.

Let \( S_n \) denote the set of permutations on \( [n] := \{1, 2, \ldots, n\} \). A permutation \( w \in S_n \) gives a box-ball system state by assigning the one-line notation of the permutation to \( n \) consecutive boxes. We denote an empty box by \( e := n + 1 \), and we usually omit the infinitely many empty boxes to the left of the balls (even though our boxes are indexed by \( \mathbb{Z} \)). Let \( BB^t(X) \) denote the result of applying \( t \) BBS moves to a BBS configuration \( X \). For example, beginning with a configuration

\[
BB^0(X) = 452361\underbrace{eeeeeee\cdots}
\]

at time \( t = 0 \), one BBS move (in which all balls jump once, starting with ball 1 and ending with ball 6) results in the new configuration (at \( t = 1 \))

\[
BB^1(X) = eee45e2136\underbrace{eeee\cdots}.
\]

A second BBS move produces the \( (t = 2) \) configuration

\[
BB^2(X) = \underbrace{eeee}452ee136\underbrace{ee\cdots},
\]

and a third BBS move produces the \( (t = 3) \) configuration

\[
BB^3(X) = \underbrace{eeeee}425ee136\cdots.
\]

At every subsequent time step, the three balls 136 advance three spaces to the right, the pair 25 advances two spaces to the right, and the singleton 4 advances one space to the right. See Figure 2. These blocks are called \textit{solitons} — maximal consecutive increasing sequences of balls that are preserved by all future BBS moves. The configurations where \( t \geq 3 \) are said to be in \textit{steady state}, because each ball is contained in a soliton. The \textit{steady-state time} of this permutation (the number of BBS moves required to reach steady state) is \( t = 3 \).

\begin{figure}[h]
\centering
\begin{tabular}{cccccccccccc}
\hline
\multicolumn{4}{c}{\textbf{t = 0}} & \multicolumn{5}{c}{4 & 5 & 2 & 3 & 6 & 1} & \multicolumn{4}{c}{\cdots} \\
\hline
\multicolumn{4}{c}{\textbf{t = 1}} & \multicolumn{5}{c}{\underbrace{4 & 5 & 2 & 1 & 3 & 6} & \cdots} \\
\hline
\multicolumn{4}{c}{\textbf{t = 2}} & \multicolumn{5}{c}{4 & 5 & 2 \underbrace{1 & 3 & 6} & \cdots} \\
\hline
\multicolumn{4}{c}{\textbf{t = 3}} & \multicolumn{5}{c}{4 & 2 & 5 \underbrace{1 & 3 & 6} & \cdots} \\
\hline
\multicolumn{4}{c}{\textbf{t = 4}} & \multicolumn{5}{c}{4 & 2 & 5 \underbrace{1 & 3 & 6} & \cdots} \\
\hline
\multicolumn{4}{c}{\textbf{t = 5}} & \multicolumn{5}{c}{4 & 2 & 5 \underbrace{1 & 3 & 6} & \cdots} \\
\hline
\end{tabular}
\caption{A box-ball system with the permutation 452361 at \( t = 0 \)}
\end{figure}

Every box-ball system eventually reaches steady state, decomposing into solitons whose sizes are weakly increasing from left to right, i.e., forming an integer partition. We can encode this \textit{soliton decomposition} of the box-ball system in a tableau whose first row is the rightmost soliton, the second row is the second rightmost soliton, and so on. Note that each row of this tableau is necessarily an increasing sequence, but the columns do not have to be increasing. The shape of the soliton decomposition is called the \textit{BBS soliton partition}.
Given a permutation $w$, its soliton decomposition $\text{SD}(w)$ is the soliton decomposition of the box-ball system containing $w$. For example, the soliton decomposition of the permutation $w = 452361$ is

$$\text{SD}(w) = \begin{array}{ccc}
1 & 3 & 6 \\
2 & 5 \\
4
\end{array}$$

1.2. Robinson–Schensted tableaux. A tableau is called *standard* if the entries in its rows and columns are increasing and each of the integers in $[n]$ appears exactly once. A popular way to associate standard tableaux to permutations is via the Robinson–Schensted (RS) correspondence

$$w \mapsto (P(w), Q(w))$$

from $S_n$ onto pairs of standard size-$n$ tableaux of the same shape [Sch61]. The tableau $P(w)$ is called the *insertion tableau* of $w$, and the tableau $Q(w)$ is called the *recording tableau* of $w$. The shape of these tableaux is called the *RS partition* of $w$. For more details, see for example the textbook [Sag01, Chapter 3].

Schensted’s classical theorem says that the size of the first row (respectively, first column) of the RS partition of $w$ is equal to the length of a longest increasing (respectively, decreasing) subsequence of the one-line notation of $w$. A localized version of Schensted’s theorem due to Lewis, Lyu, Pylyavskyy, and Sen interprets the size of the first row and the size of the first column of the BBS soliton partition as certain preserved statistics in a box-ball system. We discuss both theorems in Section 3.

As noted earlier, the soliton decomposition $\text{SD}(w)$ of a permutation $w$ is not necessarily a standard tableau. However, it is shown in [DGGRS21] that $\text{SD}(w)$ is a standard tableau iff its shape coincides with the RS partition of $w$ iff $\text{SD}(w) = P(w)$. This connection between the soliton decomposition of a permutation and its insertion tableau motivates us to define a permutation $w$ to be BBS *good* (good for short) if $\text{SD}(w) = P(w)$. We conjecture that good permutations are closed under consecutive pattern containment (Conjecture 8.5).

1.3. RS recording tableaux. Having seen the relationship between BBS soliton decompositions and RS insertion tableaux described in the previous paragraph, it is natural to ask whether RS recording tableaux may play a role in the study of box-ball systems. Surprisingly, the recording tableau of a permutation determines the BBS dynamics of the permutation, in the following sense.

**Theorem A.** If $\pi, w$ are permutations such that $Q(\pi) = Q(w)$, then the following holds.

1. $\pi$ and $w$ have the same steady-state time (Theorem 4.5)
2. The shape of $\text{SD}(\pi)$ equals the shape of $\text{SD}(w)$ (Theorem 4.7)
3. $\pi$ is good iff $w$ is good (Theorem 4.9)

The last theorem tells us that the recording tableau $Q(w)$ determines whether or not $w$ is good, so we define a standard tableau $T$ to be *good* if $Q(w) = T$ implies $w$ is good, equivalently, if $Q(w) = T$ for some good permutation $w$. We conjecture that good tableaux are counted by the Motzkin numbers (Conjecture 8.6).

1.4. First solitons and steady-state times. In Section 5 (respectively, 6 and 7), we study the number of BBS moves required to create the rightmost soliton (respectively, all solitons).

In Section 5, we prove that applying one BBS move to a permutation is enough to produce the rightmost soliton of the box-ball system.
Theorem B (Theorem 5.5). If $X$ is a BBS configuration corresponding to a permutation $w$, then the rightmost soliton is created after applying at most one BBS move to $X$, and this rightmost soliton is equal to the first row of $P(w)$.

Theorem B is helpful for proving the rest of our results.

Theorem C (Theorem 6.1). If a permutation $w$ has an L-shaped soliton decomposition, that is, the shape of SD($w$) is of the form $(s, 1, 1, \ldots)$, then the steady-state time of $w$ is either 0 or 1.

In Section 6, we also show that permutations whose soliton decompositions are L-shaped include column reading words and noncrossing involutions, so Theorem C covers a large class of permutations.

We also investigate upper bounds of steady-state times. It was conjectured in [DGGRS21, Conjecture 1.1] that the steady-state time of a permutation in $S_n$ is at most $n - 3$. We prove a special case of this conjecture.

Theorem D (Theorem 7.5). All permutations with RS partition $(n - 3, 2, 1)$ have steady-state time at most $n - 3$.

In Section 7.2, we use Bender–Knuth involutions to construct a sequence of tableaux with steady-state times from 0 to $n - 3$.

2. Steady State

A BBS configuration $X$ is a sequence indexed by $\mathbb{Z}$ where each number in $[n]$ (each denoting a ball) appears exactly once, and $e := n + 1$ (denoting an empty box) appears infinitely many times. Let $BB^t(X)$ denote the result of applying $t$ BBS moves to a BBS configuration $X$.

An increasing run of a permutation is a maximal increasing contiguous nonempty subsequence. For example, the increasing runs of the permutation 452361 are 45, 236, and 1. An increasing run of a BBS configuration is a maximal increasing contiguous nonempty sequence of balls.

A soliton is an increasing run that is preserved by all subsequent BBS moves. A BBS configuration is said to be in steady state if every ball is contained in a soliton. The steady-state time of a permutation $w$ is the number of BBS moves required for $w$ to reach steady state.

2.1. Increasing decomposition and steady state. In this section, we give a set of criteria for steady state.

Definition 2.1. The increasing run decomposition of a BBS configuration $X$, denoted by ID($X$), is the table where the rightmost increasing run of $X$ is the first (top) row, the next increasing run to its left is the second row, and so on.

Example 2.2. Let $X = eee e 4 5 2 e e 1 3 e 6 e e \cdots$. Then

\[
\text{ID}(X) = \begin{bmatrix}
1 & 3 & 6 \\
2 \\
4 & 5
\end{bmatrix}
\]

Remark 2.3. A special case of [LLPS19, Lemma 2.1] is that the height of the increasing run decomposition (i.e. the number of increasing runs) is an invariant of the box-ball system, that is, the number of rows in ID($X$) is equal to that of ID($BB^t(X)$) for all $t \in \mathbb{Z}$. See Theorem 3.8.
Remark 2.4. A BBS configuration $X$ is in steady state iff
$$\text{ID}(\text{BB}^t(X)) = \text{ID}(X) \text{ for each } t = 1, 2, 3, \ldots$$
If $X$ is a BBS configuration which is in steady state, then by definition $\text{ID}(X)$ is equal to the soliton decomposition of the BBS system.

The following gives a way to check whether a BBS configuration has reached steady state.

**Proposition 2.5** (Steady-state characterization using ID). A BBS configuration $X$ is in steady state iff
1. the rows of $\text{ID}(X)$ are weakly decreasing in length, and
2. $\text{ID}(\text{BB}(X)) = \text{ID}(X)$

2.2. Configuration array and steady state. A BBS state $X$ can be represented by the configuration array $\text{CA}(X)$ containing the integers from 1 to $n$ as follows: scanning the boxes from right to left, each increasing run becomes a row in the array. A string of $g$ empty boxes indicates that the next row below should be shifted $g$ spaces to the left. Note that this array has increasing rows but not necessarily increasing columns; it may be disconnected and it may not have a valid skew shape.

The following is a corollary of a characterization for steady state (called ‘separation condition’) given in [LLPS19]. For a proof, see [DGGRS21, Section 5].

**Proposition 2.6** (Steady-state characterization using CA). A BBS configuration $X$ is in steady state iff its configuration array $\text{CA}(X)$ is a standard (possibly disconnected) skew tableau whose rows are weakly decreasing in length.

**Example 2.7.** Let $w = 452361$, the example from Figure 2. The following are the box-ball system states from time $t = 0$ to 4 and their configuration arrays.

\[
\begin{align*}
\text{BB}^0(X) &= 452361eeeeeee\ldots & \begin{array}{cccc}
1 & 2 & 3 & 6 \\
4 & 5 & & \\
\end{array} \\
\text{BB}^1(X) &= ee45e2136eeee\ldots & \begin{array}{cccc}
1 & 3 & 6 & \\
2 & & \ & \\
4 & 5 & & \\
\end{array} \\
\text{BB}^2(X) &= eee452ee136ee\ldots & \begin{array}{cccc}
2 & & & \\
1 & 3 & 6 & \\
4 & 5 & & \\
\end{array} \\
\text{BB}^3(X) &= eeee425eee136\ldots & \begin{array}{cccc}
2 & 5 & & \\
1 & 3 & 6 & \\
4 & & & \\
\end{array} \\
\text{BB}^4(X) &= eeeee425eee136\ldots & \begin{array}{cccc}
2 & 5 & & \\
1 & 3 & 6 & \\
4 & & & \\
\end{array}
\end{align*}
\]

In this box-ball system, all configurations at time $t \geq 3$ are in steady state, so the steady-state time of 452361 is 3.
Remark 2.8. The row reading word of a tableau is the permutation formed by concatenating the rows of the tableau from bottom to top. For instance, 425136 is the row reading word of the standard tableau

\[
\begin{array}{c}
1 & 3 & 6 \\
2 & 5 \\
4
\end{array}
\]

It follows from Proposition 2.6 that a permutation has steady-state time 0 iff it is the row reading word of a standard tableau.

3. A localized version of Schensted’s theorem for box-ball systems

Schensted’s theorem explores connection between the RS partition and lengths of increasing and decreasing subsequences. A localized version of Schensted’s theorem by Lewis, Lyu, Pylyavskyy, and Sen explores similar connection between the BBS soliton partition and certain invariants of the BBS system. We describe both theorems in this section.

3.1. Schensted’s theorem and RS partition. In [Sch61, Theorem 1], Schensted gives meaning to the first row and the first column of an RS partition.

Let \( i(w) \) (respectively, \( d(w) \)) denote the size of a longest increasing (respectively, decreasing) subsequence of the one-line notation of a permutation \( w \).

**Theorem 3.1** (Schensted’s theorem). The size of the first row (respectively, first column) of the RS partition of a permutation \( w \) is equal to \( i(w) \) (respectively, \( d(w) \)).

**Example 3.2.** Let \( w = 5623714 \). The longest increasing subsequences are 567, 237, and 234, so \( i(w) = 3 \). The longest decreasing subsequences are 521, 621, 531, and 631, so \( d(w) = 3 \).

The corresponding RS tableaux are

\[
\begin{array}{ccc}
1 & 3 & 4 \\
2 & 6 & 7 \\
5
\end{array}
\] \quad
\[
\begin{array}{ccc}
1 & 2 & 5 \\
3 & 4 & 7 \\
6
\end{array}
\]

Schensted’s theorem has an important generalization in Greene’s theorem ([Gre74, Theorem 3.1]), which interprets the RS partition as sizes of largest unions of increasing and decreasing sequences. For more details, see for example Chapter 3 of the textbook [Sag01].

3.2. Localized Schensted’s theorem and BBS soliton partition. In [LLPS19, Lemma 2.1], Lewis, Lyu, Pylyavskyy, and Sen present a localized version of Greene’s theorem for box-ball systems. In this section we discuss a special case of their result, reframed to match our box-ball convention. We are calling this special case “a localized version of Schensted’s theorem”. The reason is that, when adapted to permutations, the statement of their result can be obtained from the original Schensted’s theorem with RS partition replaced by BBS soliton partition and with “size of a longest decreasing subsequence” replaced by “number of descents plus 1”.

Given a (possibly infinite, possibly finite) sequence \( X \), an integer \( j \) is called a descent of \( X \) if \( X(j) > X(j + 1) \).

**Theorem 3.3** (Localized Schensted’s theorem for permutations). Suppose \( w \) is a permutation.

1. The size of the first row of the BBS soliton partition of \( w \) is equal to \( i(w) \), the size of a longest increasing subsequence of \( w \).
2. The size of the first column of the BBS soliton partition of \( w \) is equal to \( 1 + \left| \{ \text{descents of } w \} \right| \), denoted by \( D(w) \).
Example 3.4. Let \( w = 5623714 \), the permutation from Example 3.2. Then \( i(w) = 3 \) as computed earlier, and \( D(w) = 1 + |\{\text{descents of } 5623714\}| = 1 + |\{2, 5\}| = 3 \). Note that the soliton decomposition
\[
\text{SD}(w) = \begin{array}{ccc}
1 & 3 & 4 \\
2 & 7 \\
5 & 6 
\end{array}
\]
is a nonstandard\(^2\) tableau, and \( \text{sh SD}(w) = (3, 2, 2) \) is less than \( \text{sh P}(w) = (3, 3, 1) \) in the dominance partial order. In particular, \( \text{SD}(w) \neq P(w) \), see Theorem 4.8.

Remark 3.5. Let \( w \) be a permutation.

1. Schensted’s theorem (Theorem 3.1) and localized Schensted’s theorem for permutations (Theorem 3.3) tells us that
\[
\text{(the size of the first row of } P(w)\text{)} = \text{(the size of the first row of } \text{SD}(w)\text{)} = \text{(the size of the first (rightmost) soliton of } w\text{)}.
\]

2. It follows from the definition that
\[
\text{d}(w) \leq D(w).
\]
Combining this with Theorem 3.1 and Theorem 3.3,
\[
\text{(the size of the first column of } P(w)\text{)} \leq \text{(the size of the first column of } \text{SD}(w)\text{)} = \text{(the number of solitons of } w\text{)}.
\]

The statistics in Theorem 3.3 can be defined for general BBS configurations. Recall that a BBS configuration is a sequence indexed by \( \mathbb{Z} \) where each number in \([n]\) (denoting balls) appears exactly once, and \( e := n + 1 \) (denoting empty boxes) appears infinitely many times.

Definition 3.6. Let \( X \) be a BBS configuration with \( n \) distinct balls labeled by \([n]\).

1. Given a finite, increasing sequence \( u \) of balls in \( X \), the penalized length \( \text{of } u \text{ in } X \) is the number of balls in \( u \) minus the number of gaps (i.e., empty boxes) between the first and last balls of \( u \). Let \( I(X) \) denote the maximum penalized length of increasing subsequences of balls in \( X \).

2. Let \( D(X) \) denote the number of descents of \( X \), equivalently, the number of rows of \( \text{ID}(X) \).

Note that, since \( X \) consists of \( n \) balls and the empty boxes have values \( e = n + 1 \), we have \( 1 \leq D(X) \), \( I(X) \leq n \). If the leftmost ball is in box \( j \), then \( j - 1 \) is a descent of \( X \), since \( X(j - 1) = e > X(j) \).

Example 3.7. Consider the following BBS configuration \( X \):
\[
\begin{array}{ccccccccccc}
\text{j} & \ldots & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & \ldots \\
\text{X(j)} & \ldots & \text{e} & \text{e} & 4 & 5 & \text{e} & 2 & 1 & 3 & \text{e} & \ldots
\end{array}
\]
Since the leftmost ball is in box 3, the integer 2 is a descent of \( X \). The other descents of \( X \) are 5 and 6, so \( D(X) = 3 \). The number of descents of \( X \) is equal to the number of rows of \( \text{ID}(X) \):
\[
\begin{array}{ccc}
1 & 3 & 6 \\
2 & \text{e} & \\
4 & 5
\end{array}
\]
\(^2\)The second column of the tableau is 376, a non-increasing sequence, when read from top to bottom.
The penalized length of the length-3 increasing sequence of balls 4 5 6 is 3 − 1 = 2 because there is one empty box between 4 and 6. We have I(X) = 3 because the penalized length of 1 3 6 is 3.

**Theorem 3.8** (Localized Schensted’s theorem for BBS configurations). Suppose X is a BBS configuration with n distinct balls labeled by [n].

1. The statistic I(X) is an invariant of the box-ball system, that is, given another configuration Y in the same box-ball system, I(Y) = I(X).

2. The statistic D(X) is also preserved by BBS moves. That is, given another configuration Y in the same box-ball system, we have D(Y) = D(X); in other words, the number of rows of ID(Y) is equal to that of ID(X).

In particular, the size of the first row (respectively, column) of the soliton decomposition of the box-ball system is equal to I(X) (respectively, D(X)). If w is a permutation in the same box-ball system as X, then I(X) = i(w) and D(X) = D(w).

4. Recording tableau determines BBS dynamics

The recording tableau completely determines the BBS dynamics of a permutation, in the following sense (Proposition 4.6): if two permutations have the same Q-tableau, all BBS configurations in the two corresponding box-ball systems are identical if we remove the ball labels. We prove that the recording tableau determines the steady-state time (Theorem 4.5) and the BBS soliton partition (Theorem 4.7) of a permutation. We also define the notion of a BBS good permutation and prove that the Q-tableau determines whether or not a permutation is good (Theorem 4.9).

4.1. Dual Knuth equivalence. We review the concept of dual Knuth equivalence, which was introduced by Haiman [Hai92].

Two permutations π, w ∈ Sn differ by a dual Knuth relation of the first kind (denoted π K±1 ∼ w), if for some k,

π = \ldots k + 1 \ldots k \ldots k + 2 \ldots \text{ and} \\
w = \ldots k + 2 \ldots k \ldots k + 1 \ldots

or vice versa. They differ by a dual Knuth relation of the second kind (denoted π K±2 ∼ w), if for some k,

π = \ldots k \ldots k + 2 \ldots k + 1 \ldots \text{ and} \\
w = \ldots k + 1 \ldots k + 2 \ldots k \ldots

or vice versa.

The two permutations are dual Knuth equivalent if there is a sequence of permutations such that

π = π₁ K⁺₁ ∼ π₂ K⁺₂ ∼ \cdots ∼ πₖ K⁺ₖ = w

where i, j, . . . , l ∈ {1, 2}.

Two permutations π, w ∈ Sn are said to be Q-equivalent if Q(π) = Q(w). A very helpful fact from [Hai92, Proposition 2.4] tells us that the dual Knuth equivalence classes and Q-equivalence classes coincide.

**Theorem 4.1.** Q(π) = Q(w) iff π and w are dual Knuth equivalent.
4.2. BBS moves preserve dual Knuth equivalence. Recall that a BBS configuration is a sequence indexed by \( \mathbb{Z} \) where each of 1, \ldots, \( n \) (denoting balls) appears exactly once, and \( e := n + 1 \) (denoting empty boxes) appears infinitely many times. We extend the definition of dual Knuth relation to BBS configurations but insist that the two entries being swapped must be balls. Let \( \text{BB}(X) \) denote the result of applying one BBS move to a BBS configuration \( X \).

The following lemma is key to proving the results of this section.

**Lemma 4.2.** Let \( X \) and \( Y \) be two BBS configurations.

1. Suppose \( X \) and \( Y \) differ by a dual Knuth relation of the first kind, say,
   \[
   X = \ldots, k + 1, \ldots, k, \ldots, k + 2, \ldots, \quad Y = \ldots, k + 2, \ldots, k, \ldots, k + 1, \ldots,
   \]
   where \( k + 2 \) represents a ball (as opposed to an empty box \( e = n + 1 \)). Then \( \text{BB}(X) \) and \( \text{BB}(Y) \) also differ by a dual Knuth relation of the first kind such that the relative order of the balls \( k, k + 1, \) and \( k + 2 \) are preserved:
   \[
   \text{BB}(X) = \ldots, k + 1, \ldots, k, \ldots, k + 2, \ldots, \quad \text{BB}(Y) = \ldots, k + 2, \ldots, k, \ldots, k + 1, \ldots.
   \]

2. Suppose \( X \) and \( Y \) differ by a dual Knuth relation of the second kind, say,
   \[
   X = \ldots, k, \ldots, k + 2, \ldots, k + 1, \ldots, \quad Y = \ldots, k + 1, \ldots, k + 2, \ldots, k, \ldots,
   \]
   where \( k + 2 \) represents a ball (as opposed to an empty box). Then \( \text{BB}(X) \) and \( \text{BB}(Y) \) differ by a dual Knuth relation of some kind, and \( \text{BB}(X) \) and \( \text{BB}(Y) \) differ by swapping either \( k, k + 1 \) or \( k, k + 2 \). During the BBS move, consider the situation immediately after the balls 1, 2, \ldots, \( k - 1 \) have finished jumping.
   (a) If, immediately after all balls smaller than \( k \) have jumped, there is at least one empty box between \( k \) and \( k + 1 \), then
   \[
   \text{BB}(X) = \ldots, k, \ldots, k + 2, \ldots, k + 1, \ldots, \quad \text{BB}(Y) = \ldots, k + 1, \ldots, k + 2, \ldots, k, \ldots,
   \]
   that is, \( \text{BB}(X) \) and \( \text{BB}(Y) \) also differ by a dual Knuth relation of the second kind, and the relative order of the balls \( k, k + 1 \), and \( k + 2 \) are the same as in \( X \) and \( Y \), respectively.
   (b) If, immediately after all balls smaller than \( k \) have jumped, there are no empty boxes between \( k \) and \( k + 1 \), then
   \[
   \text{BB}(X) = \ldots, k + 2, \ldots, k, \ldots, k + 1, \ldots, \quad \text{BB}(Y) = \ldots, k + 1, \ldots, k, \ldots, k + 2, \ldots,
   \]
   that is, \( \text{BB}(X) \) and \( \text{BB}(Y) \) differ by a dual Knuth relation of the first kind.

In both cases, the positions of the nonempty boxes in \( \text{BB}(X) \) and \( \text{BB}(Y) \) are the same.

**Example 4.3.** To illustrate case (2b) of Lemma 4.2, consider the BBS configurations
   \[
   X = 451362 \quad \text{and} \quad Y = 452361.
   \]
   They differ by a dual Knuth relation of the second kind where \( k = 1 \), and there is no empty box between \( k = 1 \) and \( k + 2 = 3 \). We have
   \[
   \text{BB}(X) = 45c3126 \quad \text{and} \quad \text{BB}(Y) = 45c2136.
   \]
   Indeed, \( \text{BB}(X) \) and \( \text{BB}(Y) \) differ by a dual Knuth relation of the first kind.

**Lemma 4.4.** Suppose \( X \) and \( Y \) are two BBS configurations that differ by a dual Knuth relation; the two configurations are identical except that two balls \( j \) and \( j + 1 \) are swapped.

   1. Then \( X \) is in steady state iff \( Y \) is in steady state.
   2. We have \( \text{shID}(X) = \text{shID}(Y) \), and the gaps between the increasing runs are the same. Equivalently, the configuration arrays \( \text{CA}(X) \) and \( \text{CA}(Y) \) have the same shape.
Theorem 4.7. If $Q(\pi) = Q(w)$ then $\pi$ and $w$ have the same steady-state time.

Proof. Assume $Q(\pi) = Q(w)$. Then $\pi$ and $w$ are related by a sequence of dual Knuth relations corresponding to swapping two balls $l$ times. Fix $t$, and let $X = BB^t(\pi)$ and $Y = BB^t(w)$. By Lemma 4.2, applied $t$ times, we know that $X$ and $Y$ are also related by a sequence of $l$ dual Knuth relations. Therefore, Lemma 4.4(1), applied $l$ times, tells us that that $X$ is in steady state iff $Y$ is in steady state. So $\pi$ and $w$ first reach steady state at the same time. □

4.3. $Q$ determines the steady-state time of a permutation. We now prove that the recording tableau determines the steady-state time of a permutation. Let $BB^t(X)$ denote the result of applying $t$ BBS moves to a BBS configuration $X$.

Theorem 4.5. If $Q(\pi) = Q(w)$ then $\pi$ and $w$ have the same steady-state time.

Proof. Assume $Q(\pi) = Q(w)$. Then $\pi$ and $w$ are related by a sequence of dual Knuth relations corresponding to swapping two balls $l$ times. Fix $t$, and let $X = BB^t(\pi)$ and $Y = BB^t(w)$. By Lemma 4.2, applied $t$ times, we know that $X$ and $Y$ are also related by a sequence of $l$ dual Knuth relations. Therefore, Lemma 4.4(1), applied $l$ times, tells us that that $X$ is in steady state iff $Y$ is in steady state. So $\pi$ and $w$ first reach steady state at the same time. □

4.4. $Q$ determines the BBS soliton partition of a permutation. In this section, we prove Theorem 4.7, which says that the recording tableau determines the BBS soliton partition of a permutation.

Proposition 4.6. If $Q(\pi) = Q(w)$, then, at every $t$,

(1) the positions of the nonempty boxes in $BB^t(\pi)$ and $BB^t(w)$ are equal;
(2) $sh\ ID(BB^t(\pi)) = sh\ ID(BB^t(w))$, and the gaps between the increasing runs are the same. Equivalently, $CA(BB^t(\pi))$ and $CA(BB^t(w))$ are of identical shape for each $t$.

Proof. Assume $Q(\pi) = Q(w)$. Then $\pi$ and $w$ are related by a sequence of dual Knuth relations corresponding to a sequence of $l$ two-ball swaps. We fix $t$, and let $X = BB^t(\pi)$ and $Y = BB^t(w)$. By Lemma 4.2, applied $t$ times, we know that $X$ and $Y$ are also related by a sequence of $l$ dual Knuth relations, and the nonempty boxes of $X$ and $Y$ are in the same positions. Therefore, Lemma 4.4(2), applied $l$ times, tells us that the configuration arrays $CA(X)$ and $CA(Y)$ are of identical shape. □

Theorem 4.7. If $Q(\pi) = Q(w)$ then $sh\ SD(\pi) = sh\ SD(w)$.

Proof. Suppose $Q(\pi) = Q(w)$. Let $t$ be such that $BB^t(\pi)$ and $BB^t(w)$ are both in steady state. Proposition 4.6 tells us that $sh\ ID(BB^t(\pi)) = sh\ ID(BB^t(w))$. Since $BB^t(\pi)$ and $BB^t(w)$ are in steady state, we have $ID(BB^t(\pi)) = SD(\pi)$ and $ID(BB^t(w)) = SD(w)$. Hence $sh\ SD(\pi) = sh\ SD(w)$. □
4.5. **Good recording tableaux.** In general the soliton decomposition and the RS insertion tableau of a permutation do not coincide. However, the following shows that having a standard soliton decomposition tableau or having a BBS soliton partition which equals the RS partition is enough to guarantee that the soliton decomposition and the RS insertion tableau coincide.

**Theorem 4.8** ([DGGRS21, Theorem 4.2]). If $w$ is a permutation, then the following are equivalent:

1. $\text{SD}(w) = \text{P}(w)$
2. $\text{SD}(w)$ is a standard tableau
3. The shape of $\text{SD}(w)$ equals the shape of $\text{P}(w)$

We say that a permutation $w$ is *BBS good*, or *good* for short, if $\text{SD}(w)$ is a standard tableau.

It turns out that $Q(w)$ determines whether or not $w$ is good.

**Theorem 4.9.** Given a $Q$-equivalence class, either all permutations in it are good or all of them are not good.

Therefore, it makes sense to define a standard tableau $T$ to be *BBS good* if $Q(w) = T$ implies $w$ is good (equivalently, if $Q(w) = T$ for some $w$ which is good).

**Proof of Theorem 4.9.** Let $Q(\pi) = Q(w)$. Assume $\pi$ is good, that is, $\text{SD}(\pi)$ is standard. Then

$$\text{sh} \text{SD}(w) = \text{sh} \text{SD}(\pi) \text{ by Theorem 4.7}$$

$$= \text{sh} \text{P}(\pi) \text{ by Theorem 4.8}$$

$$= \text{sh} \text{P}(w),$$

where the last equality is due to $Q(\pi) = Q(w)$ and the fact that the P-tableau and Q-tableau of a permutation have the same shape. Since $\text{sh} \text{SD}(w) = \text{sh} \text{P}(w)$, Theorem 4.8 tells us that $\text{SD}(w)$ is standard and thus $w$ is good. □

Conjectures about good permutations and good tableaux are given in Section 8.

5. **First soliton is created after one BBS move**

In this section, we prove that applying one BBS move to a permutation $w$ is enough to obtain the first (rightmost) soliton of $\text{SD}(w)$, and this first soliton is equal to the first row of $\text{P}(w)$. Our proof uses the carrier algorithm (explained below) and the localized Schensted’s theorem (see Section 3.2).

5.1. **Carrier algorithm.** The carrier algorithm is a way to transform a BBS configuration at time $t$ into the configuration at time $t+1$. In the algorithm, we move numbers in and out of a carrier in a way that is similar to the insertion rule of the RS algorithm. A version of the carrier algorithm was first introduced in [TM97], and the following definition comes from [Fuk04, Section 3.3].

**Definition 5.1** (Carrier algorithm). Let $X$ be a BBS configuration with $n$ balls, that is, $X$ is a sequence indexed by $\mathbb{Z}$ where each of $1, \ldots, n$ (denoting balls) appears exactly once, and $e := n + 1$ (denoting empty boxes) appears infinitely many times. Let the length-$n$ sequence $C = ee \ldots$ be the initial state of the carrier, and let $j$ be the smallest number such that $X(j) \neq e$. Let $X'$ be a BBS configuration defined as follows.

**Step 1:** (a) If $X(j) < \max C$, let $y$ be the smallest number in the carrier $C$ greater than $X(j)$. Set $X'(j) = y$. Remove $y$ out of $C$ and insert $X(j)$ into $C$. 
(b) If $X(j) \geq \max C$, let $m = \min C$. Set $X'(j) = m$. Remove $m$ out of $C$ and insert $X(j)$ into $C$.

Step 2: Set $j := j + 1$. If $X(k) \neq e$ for some $k \geq j$ or if $C$ still contains balls, repeat Step 1; Otherwise, we are done.

Let $X'(i) = e$ for the rest of the boxes $i$ which have not been assigned a value.

Note that, since we have exactly $n$ balls and the carrier can carry $n$ elements, we have

- $X(j) \neq e$ iff $C$ has a number greater than $X(j)$, which is case (1a) in Step 1, and
- $X(j) = e$ iff $C$ has no number greater than $X(j)$, which is case (1b) in Step 1.

**Theorem 5.2** ([Fuk04, Proposition 3.2]). Running the carrier algorithm once is equivalent to performing a BBS move once. That is, given a BBS state $X$ at time $t$, the BBS configuration $X'$ we get by performing the carrier algorithm is the state at time $t + 1$.

**Remark 5.3.** When we insert a consecutive sequence of balls (for example, when $X$ comes from a permutation), the rule for bumping and inserting numbers into and out of the carrier is the same as the rule for bumping and inserting numbers into and out of the first row of the insertion tableau during the RS algorithm.

**Example 5.4.** We apply the carrier algorithm to the BBS configuration of the box-ball system from Figure 2 at time $t = 2$ $452ee136$ to obtain the configuration at time $t = 3$. For the purpose of proving the main result of this section, it is helpful to break the carrier algorithm into two processes: the first process is to insert into the carrier $C$ all balls and the $e$’s between them. The second process is to “flush” out all balls from $C$ by inserting enough $e$’s into it.

```
begin Process 1: insert all balls
  eeeee 452 5ee136
  ee4 52ee136
  ee 45ee2e136
  ee452 eee136
  ee425 eee136
  ee425 eee136
  ee425 eee136
  ee425 eee136
  ee425 eee136
end Process 1
```

```
begin Process 2: flushing process
  ee425 eee136 eee136 ← e
  ee425 eee136 eee136 ← e
  ee425 eee136 eee136 ← e
  ee425 eee136 eee136 ← e
end Process 2
```

The sequence $ee425 eee136$ to the left of the carrier corresponds to the configuration at time $t = 3$ given in Figure 2.

5.2. **First soliton.** We refer to the rightmost soliton in a steady-state configuration as the first soliton of the box-ball system. The first soliton of a permutation $w$ is the first soliton of the box-ball system containing $w$, that is, the first row of $SD(w)$. Let $Row_1(T)$ denote the first row of a tableau $T$. The following result shows that the first soliton is created after
Theorem 5.5. If \( w \) is a permutation, then we have the following.

1. The first soliton \( \text{Row}_1(\text{SD}(w)) \) contains the ball 1.
2. The first soliton \( \text{Row}_1(\text{SD}(w)) \) of \( w \) is created after at most one BBS move. That is, the rightmost increasing run of \( \text{BB}^t(w) \) is equal to the first soliton of \( w \) for all \( t \geq 1 \).
3. \( \text{Row}_1(\text{SD}(w)) = \text{Row}_1(\text{P}(w)) \).

Proof. Let \( w = w_1w_2 \ldots w_n \in S_n \). If \( w \) has steady-state time 0 then the first soliton of \( w \) is already created, so suppose that \( w \) has steady state time \( m \geq 1 \).

For each time \( t \), let \( a_t \) denote the increasing run containing the ball 1 in the BBS configuration \( \text{BB}^t(w) \). We will prove that \( \text{Row}_1(\text{SD}(w)) \) is equal to \( a_m = a_{m-1} = \cdots = a_2 = a_1 \).

We apply the carrier algorithm to \( w \). Insert all the balls \( w_1, w_2, \ldots, w_n \) of \( w \) into the carrier, and pause immediately after the last ball \( w_n \) of \( w \) is inserted into the carrier. Let \( c \) denote the sequence of balls which is currently in the carrier. Since \( w \) is a permutation, the rule for bumping and inserting numbers into and out of the carrier is the same as the rule for bumping and inserting numbers into and out of the first row of the insertion tableau during the RS algorithm (see Remark 5.3). Therefore, we have

\[
c = \text{Row}_1(\text{P}(w)).
\]

In particular, \( c \) is an increasing sequence starting with the value 1. When we flush \( \text{len}(c) \) copies of \( e \) into the carrier to finish the carrier algorithm, the sequence \( c \) is the rightmost \( \text{len}(c) \) letters of \( \text{BB}(w) \). Thus \( c \) contains the value 1 and is the rightmost increasing run of \( \text{BB}(w) \). Therefore, \( c = a_1 \), and so

\[
a_1 \text{ is the rightmost increasing run of } \text{BB}(w)
\]

and

\[
a_1 = \text{Row}_1(\text{P}(w)). \quad (5.1)
\]

Schensted’s theorem (Theorem 3.1) tells us that the size of \( \text{Row}_1(\text{P}(w)) \) is equal to \( i(w) \), the length of a longest increasing subsequence of \( w \), so

\[
\text{len}(a_1) = i(w). \quad (5.2)
\]

Since \( a_1 \) is the rightmost increasing run of \( \text{BB}(w) \) and \( a_1 \) starts with the value 1, applying a BBS move to \( \text{BB}(w) \) will produce a rightmost increasing run containing \( a_1 \). So the increasing run \( a_2 \) of \( \text{BB}^2(w) \) containing 1 is the rightmost increasing run of \( \text{BB}^2(w) \) and \( a_2 \supseteq a_1 \). By the same reasoning, applying a BBS move to \( \text{BB}^2(w) \) will produce a rightmost increasing run containing \( a_2 \), so the increasing run \( a_3 \) of \( \text{BB}^3(w) \) containing 1 is the rightmost increasing run of \( \text{BB}^3(w) \) and \( a_3 \supseteq a_2 \), and so on. Therefore, the increasing run \( a_m \) is the rightmost increasing run of \( \text{BB}^m(w) \). Since \( \text{BB}^m(w) \) is in steady state,

\[
a_m \text{ is the first soliton of } w,
\]

proving part (1). In addition,

\[
a_m \supseteq \cdots \supseteq a_2 \supseteq a_1.
\]

To prove \( a_m = a_1 \), we proceed to show that \( \text{len}(a_t) \leq \text{len}(a_1) \) for \( t \geq 1 \). First note that the penalized length of \( a_t \) is \( \text{len}(a_t) \) because \( a_t \) is an increasing run. Since \( \text{I}(\text{BB}^t(w)) \) is defined to be the maximum penalized length over all increasing sequences of balls in \( \text{BB}^t(w) \), we have

\[
\text{len}(a_t) \leq \text{I}(\text{BB}^t(w))
\]
Thus, $a_m = a_1$. Since $a_1$ is the rightmost increasing run of $BB(w)$ and $a_m$ is the first soliton of $w$, this equality concludes the proof of part (2) of the theorem.

Finally, we have $a_m = a_1 = \text{Row}_1(P(w))$ by (5.1), proving part (3).

\[ \Box \]

**Corollary 5.6.** Let $w$ be a permutation and suppose that the $k$ rightmost solitons of $w$ are already formed. Then it takes at most one BBS move to create the $(k + 1)\text{th}$ rightmost soliton of $w$.

**Remark 5.7.** Corollary 5.6 does not hold if we replace $w$ with a BBS configuration that has empty boxes between balls. For example, consider the configuration

$$ee45e2136$$

from Figure 2. In this configuration the first soliton 136 has already been created. However, the second soliton 25 is not created until after two BBS moves later.

### 6. L-shaped soliton decompositions

In this section, we prove that permutations with L-shaped soliton decompositions have steady-state time at most 1. We also study noncrossing involutions, nested involutions, and column reading words. We prove that these involutions all have L-shaped soliton decompositions and therefore have steady-state time at most 1.

#### 6.1. L-shaped soliton decompositions

Let $sst(w)$ denote the steady-state time of a permutation $w$.

**Theorem 6.1.** If a permutation $w$ has an L-shaped soliton decomposition, that is, the partition $sh SD(w)$ is of the form $(i, 1, 1, \ldots)$, then $sst(w) \leq 1$.

**Example 6.2.** Let $\pi = 5274163$. Applying the first BBS move, we get

$$e5e7e4136$$

which is in steady state with soliton decomposition

$$SD(\pi) = \begin{bmatrix} 1 & 3 & 6 \\ 4 \\ 2 \\ 7 \\ 5 \end{bmatrix}$$

**Proof of Theorem 6.1.** Let $h := n - i$, so the number of rows of $SD(w)$ is $1 + h$. Theorem 5.5 tells us that the rightmost increasing run of $BB^1(w)$ is equal to the first soliton. Since the number of increasing runs of a BBS configuration is preserved by a BBS move (Remark 2.3), the number of rows of $ID(BB^1(w))$ is $1 + h$. So the shape of $ID(BB^1(w))$ is equal to $(i, 1, 1, \ldots)$ and $BB^1(w)$ is of the form

$$X = b_h \ldots b_{h-1} \ldots b_2 \ldots b_1 \ldots 1 s_2 s_3 \ldots,$$

such that, for each $b_j \in \{b_1, \ldots, b_h\}$, either

1. there is an empty box immediately to the left of $b_j$, or
2. $b_{j+1} b_j$ is a consecutive, decreasing subsequence of $X$.  

Therefore, the configuration array of \(BB^1(w)\) is a standard skew tableau whose row sizes are weakly increasing, so \(BB^1(w)\) is in steady state by Proposition 2.6. □

Next, we point out a characterization of permutations with L-shaped soliton decompositions.

**Lemma 6.3.** Suppose \(w\) is a permutation in \(S_n\). Let \(i\) denote the length of a longest increasing subsequence of \(w\) and \(des\) the number of descents of \(w\). Then \(SD(w)\) is L-shaped if \(i + des \geq n\). In this case, \(SD(w)\) has shape \((i,1,1,\ldots,1)\).

**Proof.** The localized Schensted’s theorem (Theorem 3.3) tells us that the length of the first soliton is \(i\). It also tells us that the length of the first column of the soliton decomposition is \(des + 1\). Since \(SD(w)\) has size \(n\), it must be that \(SD(w)\) is L-shaped iff \(i + des = n\). Furthermore, since \(i + des \leq n\) holds for all permutations in \(S_n\), writing \(i + des = n\) is equivalent to writing \(i + des \geq n\). □

6.2. Noncrossing involutions have L-shaped soliton decompositions.

**Definition 6.4 (Noncrossing involution).** A pair of distinct 2-cycles is called a crossing if they can be written as \((ac)\) and \((bd)\) where \(a < b < c < d\). An involution is called noncrossing if no pair of 2-cycles is a crossing.

For example, the involution with cycle notation \((26)(34)(78)\) is noncrossing, but the involution with cycle notation \((24)(36)(78)\) is not noncrossing, since \((24)\) and \((36)\) is a pair of crossing 2-cycles. Any 2-cycle is a noncrossing involution, as is the identity permutation.

**Proposition 6.5.** If \(w\) is a noncrossing involution, then \(w\) has an L-shaped soliton decomposition. More precisely, let \(w\) be a noncrossing involution in \(S_n\), let \(c\) denote the number of adjacent 2-cycles of \(w\), and let \(k\) denote the number of all 2-cycles of \(w\) (including the adjacent 2-cycles). Then the shape of \(SD(w)\) is \((n - 2k + c,1,1,\ldots,1)\).

The following example illustrates the idea of our proof of Proposition 6.5.

**Example 6.6.** Let \(w = 164352879 = (26)(34)(78) \in S_9\). First, we construct an increasing subsequence of the one-line notation of \(w\). Since \(w\) has three 2-cycles, we know that \(w\) has exactly \(9 - 2(3) = 3\) fixed points: 1, 5, and 9. These three fixed points form an increasing subsequence of \(w\). We have two adjacent 2-cycles \((34)\) and \((78)\), and we can add 3 and 7 to 1 5 9 to form an increasing subsequence of \(w\) of size five: 1 3 5 7 9. So the size of the first soliton of \(w\) is at least 5 by the localized Schensted’s theorem (Theorem 3.3).

Next, we look for the descents of \(w\). The one non-adjacent 2-cycle \((26)\) contributes two descents 2 and 6 – 1 = 5 to \(w\), since \(w(2) = 6 > 4 = w(3)\) and \(w(5) = 5 > 2 = w(6)\). The two adjacent 2-cycles \((34)\) and \((78)\) contribute one descent each to \(w\) because \(w(3) = 4 > 3 = w(4)\) and \(w(7) = 8 > 7 = w(8)\). We have found at least 4 descents of \(w\). So the size of the first column is at least 4 + 1 = 5 by the localized Schensted’s theorem (Theorem 3.3).

The size of \(SD(w)\) is 9, so its shape must be \((5,1,1,1,1)\). Indeed,

\[
SD(w) = \begin{pmatrix}
1 & 2 & 5 & 7 & 9 \\
8 \\
3 \\
4 \\
6
\end{pmatrix}
\]
Proof of Proposition 6.5. Let \( w \) be a noncrossing involution in \( S_n \) which is not the identity permutation, and let \( k \geq 1 \) denote the number of all 2-cycles of \( w \). First, we construct an increasing subsequence of the one-line notation of \( w \).

Since the only values changed by \( w \) are the ones in the 2-cycles, \( w \) has \( n - 2k \) fixed points. First, consider the case where \( n > 2k \), so that \( w \) indeed has fixed points. The \( n - 2k \) fixed points of \( w \) form an increasing subsequence \( a_1a_2 \ldots a_{n-2k} \) of \( w \).

Let \( c \geq 0 \) be the number of adjacent 2-cycles of \( w \), and consider the adjacent 2-cycles of \( w \) listed from smallest to largest, that is, \( i_1 < i_2 < \cdots < i_c \). Note that each of the adjacent 2-cycles simply swaps \( i_j \) and \( i_j + 1 \), so \( i_1i_2 \ldots i_c \) is an increasing subsequence of \( w \). Furthermore, if \( w \) has fixed points we can insert \( i_1, i_2, \ldots, i_c \) into the increasing subsequence \( a_1a_2 \ldots a_{n-2k} \) of \( w \) to form a longer subsequence of \( w \) of size \( n - 2k + c \). Let \( i \) denote the size of a longest increasing subsequence of \( w \); we have shown that \( i \geq n - 2k + c \).

Next, let’s compute the number of descents of \( w \). First, consider a non-adjacent 2-cycle \( (xz) \) where \( x + 1 < z \). We claim that

\[ x \text{ is a descent of } w. \]

Either \( x + 1 \) is a fixed point or \( x + 1 \) is part of a 2-cycle. If \( x + 1 \) is a fixed point, then \( w(x + 1) = x + 1 \) and we have \( w(x) = z > x + 1 = w(x + 1) \), so \( x \) is a descent of \( w \). If \( x + 1 \) is part of a 2-cycle \((x + 1, y)\), then \( y \) must be smaller than \( z \) because \( w \) is a noncrossing involution. Therefore, \( w(x) = z > y = w(x + 1) \), so again \( x \) is a descent of \( w \). Using a similar argument, we can show that

\[ z - 1 \text{ is a descent of } w. \]

For each adjacent 2-cycle \((x, x + 1)\),

the number \( x \) is a descent of \( w \)

because \( w(x) = x + 1 > x = w(x + 1) \). In total, we have shown that \( w \) has at least \( 2k - c \) descents. If we let \( \text{des} \) denote the number of descents of \( w \), we have \( \text{des} \geq 2k - c \).

We have shown that \( i \geq n - 2k + c \) and that \( \text{des} \geq 2k - c \). Since \( (n - 2k + c) + (2k - c) = n \), we have \( i + \text{des} \geq n \), so \( \text{SD}(w) \) is L-shaped with shape \((i, 1, 1, \ldots, 1)\) by Lemma 6.3.

Remark 6.7. Not all involutions with L-shaped soliton decompositions are noncrossing involutions. For instance, the involution \( \pi = 5274163 = (15)(37) \) from Example 6.2 has a crossing and has an L-shaped soliton decomposition.

The following result is a consequence of Theorem 6.1 and Proposition 6.5.

Corollary 6.8. All noncrossing involutions have steady-state time at most 1.

Two families of tableaux that correspond to noncrossing involutions are discussed next.

6.3. Nested involutions have L-shaped soliton decompositions.

Definition 6.9. A pair of distinct 2-cycles is called a nesting if they can be written as \((ad)\) and \((bc)\) where \(a < b < c < d\). An involution is called nested if every pair of 2-cycles is a nesting.
Example 6.10. Any 2-cycle is a nested involution, as is the identity permutation. The
involutions (15)(24) and (17)(25)(34) are nested involutions, but (23)(45)(17) is not because
the pair (23) and (45) is not a nesting.

Corollary 6.11. If \( w \) is a nested involution then \( \text{sst}(w) \leq 1 \).

Proof. Since a nested involution is a noncrossing involution, by Corollary 6.8 its steady-state
time is at most 1. \( \square \)

The following lemma is a special case of [Pos09, Theorem 5.2]. The forward direction of
the lemma can be proven by applying the inverse RS algorithm, and the reverse direction by
Schensted’s theorem (Theorem 3.1).

Lemma 6.12. Suppose \( w \) is an involution. Then the RS partition of \( w \) is L-shaped iff \( w \) is a
nested involution.

The following tells us that nested involutions are (BBS) good, but all other noncrossing
involutions are not good.

Proposition 6.13. Suppose \( w \) is noncrossing. Then \( w \) is good iff \( w \) is a nested involution.

Proof. Suppose \( w \) is noncrossing. By Proposition 6.5, the BBS soliton partition of \( w \) is
L-shaped. The permutation \( w \) is good iff the RS partition of \( w \) is equal to the BBS soliton
partition of \( w \) (Theorem 4.8). This equality holds iff the RS partition of \( w \) is L-shaped, which
is true iff \( w \) is a nested involution (Lemma 6.12). \( \square \)

Remark 6.14. The previous proposition implies that if an involution is good, then it either
has a crossing or it is a nested involution. (The converse is false: we can find an involution
which has a crossing but is not good. For instance, the involution \( \pi = 5274163 = (15)(37) \)
from Example 6.2 has a crossing and is not good.)

6.4. Column reading words have L-shaped soliton decompositions. Remark 2.8 tells
us that a permutation has steady-state time 0 iff it is the row reading word of a standard
tableau. In this section, we prove that the column reading word of a standard tableau has
steady-state time at most 1.

Definition 6.15. The column reading word or column word of a tableau \( T \) is the word
obtained by reading the columns of \( T \) bottom to top, from left to right.

If \( w \) is the column word of a standard tableau \( T \), then \( P(w) = T \) (see [Ful96, Section 2.3]).
For instance, 63174285 is the column word of the standard tableau

\[
T = \begin{bmatrix}
1 & 2 & 5 \\
3 & 4 & 8 \\
6 & 7
\end{bmatrix} = P(63174285).
\]

Definition 6.16. The column superstandard tableau of shape \( \lambda \) is the tableau of shape \( \lambda \)
which is obtained by filling the columns top to bottom, from left to right, with the integers
1, 2, 3, \ldots, \( n \), in this order.

The following lemma can be deduced from applying the inverse RS algorithm and from the
fact that all column reading words of standard tableaux of the same shape are dual Knuth
equivalent. The forward direction of the lemma is stated in [GHKU21, Lemma 4.2].

Lemma 6.17. A permutation \( w \) is the column reading word of a standard tableau iff \( Q(w) \)
is column superstandard.
For example, let \( w = 63174285 \) be the column word from the previous example. We have

\[
P(w) = \begin{array}{ccc}
1 & 2 & 5 \\
3 & 4 & 8 \\
6 & 7 &
\end{array}
\quad Q(w) = \begin{array}{ccc}
1 & 4 & 7 \\
2 & 5 & 8 \\
3 & 6 &
\end{array}
\]

where \( Q(w) \) is the column superstandard tableau of shape \((3, 3, 2)\).

**Remark 6.18.** If \( T \) is a column superstandard tableau, the one-line notation of the involution \( \pi \) where \( P(\pi) = Q(\pi) = T \) is the column word of \( T \). Equivalently, the cycle notation for \( \pi \) can be described as follows. Take each column of \( T \) and “fold” it in the middle. Each pair of entries that touch gives us a 2-cycle of \( \pi \), and the entry in the center of the column (if the column has odd length) gives us a fixed point of \( \pi \). Therefore, \( \pi \) is a noncrossing involution. If the second column has length at least 2, then \( \pi \) is not a nested involution (see Definition 6.9).

For example, consider \( \pi \in S_9 \) where \( P(\pi) = Q(\pi) = \begin{array}{ccc}1 & 6 & 9 \\
2 & 7 &
\end{array} \begin{array}{c}3 \\
4 \\
5 \\
6 \\
7 \\
8 \\
9
\end{array} \)

Then \( \pi \) is the column word \( 543218769 \) in one-line notation and \( \pi = (15)(24)(3)(68)(7)(9) \) in cycle notation, so \( \pi \) is a noncrossing involution which is not nested.

**Proposition 6.19.** If \( w \) is the column reading word of a standard tableau (equivalently, if \( Q(w) \) is a column superstandard tableau), then the steady-state time of \( w \) is 0 or 1.

**Proof.** Let \( w \) be the column reading word of a standard tableau (equivalently, \( Q(w) \) is a column superstandard tableau). Let \( \pi \) be the involution such that \( Q(\pi) = Q(w) \). Since \( Q(\pi) \) is column superstandard, Remark 6.18 tells us that \( \pi \) is a noncrossing involution. Therefore, we have \( \operatorname{sst}(\pi) \leq 1 \) by Corollary 6.8. Since the recording tableau of a permutation determines steady-state time (Theorem 4.5), we have \( \operatorname{sst}(w) = \operatorname{sst}(\pi) \leq 1 \).

\[
\begin{array}{c}
\end{array}
\]

7. A Maximum Steady-state Time

The following theorem and conjecture are given in [DGGRS21]. If \( n \geq 5 \), let

\[
\hat{Q}_n = \begin{array}{ccc}
1 & 2 & 5 \\
3 & 4 \ldots n-2 \ldots 1 \\
\end{array}
\]

**Theorem 7.1 ([DGGRS21, Theorem 6.7]).** If \( n \geq 5 \) and \( Q(w) = \hat{Q}_n \), then the steady-state time of \( w \) is \( n - 3 \).

**Conjecture 7.2 ([DGGRS21, Conjecture 1.1]).** Let \( n \geq 5 \) and \( w \in S_n \). If \( Q(w) \) is not equal to \( \hat{Q}_n \), then the steady-state time of \( w \) is less than \( n - 3 \).

Since the recording tableau of a permutation determines its steady-state time (Theorem 4.5), if \( T \) is a standard tableau, we can define the **steady-state time of** \( T \) to be the steady-state time for all permutations \( w \) such that \( Q(w) = T \). Let \( \operatorname{sst}(T) \) denote the steady-state time of a standard tableau \( T \). In Section 7.1, we prove a partial result: the maximum steady-state
time for tableaux of shape \((n - 3, 2, 1)\) is \(n - 3\). In Section 7.2, we present a chain of tableaux that have steady-state times from 0 to \(n - 3\).

7.1. Maximum steady-state time for tableaux of shape \((n-3, 2, 1)\).

**Lemma 7.3.** If \(w \in S_n\) and

\[
\text{sh}(P(w)) = (n - 3, 2, 1) = \begin{array}{cccc}
\hspace{1cm} & \hspace{1cm} & \hspace{1cm} & \\
\end{array}
\]

then either \(\text{sh}(\text{SD}(w))\) is \((n - 3, 2, 1)\) or \((n - 3, 1, 1, 1)\).

**Proof.** The fact that the size of the first row of \(\text{SD}(w)\) is \(n - 3\) follows from Remark 3.5(1). The size of the first column of \(\text{SD}(w)\) is at least 3 by Remark 3.5(2).

**Lemma 7.4.** Let \(n \geq 5\) and \(w \in S_n\). Suppose that at time \(t \geq 1\) we have the (non-steady-state) BBS configuration

\[
X = \text{BB}^t(w) = \ldots \begin{array}{cc}
\text{increasing run} \\
\begin{array}{cccc}
\text{m copies of } e's \\
\begin{array}{cccc}
\text{1 y}_2 \text{y}_3 & \ldots & \text{y}_{n-3} \\
\text{x} & \text{e} & \text{e} & \ldots & \text{e} \\
\text{increasing run}
\end{array}
\end{array}
\begin{array}{cccc}
\text{a} & \text{b} & \text{x} \\
\end{array}
\end{array}
\ldots
\]

where

- \(\text{a} < \text{b}\) is an increasing run and \(\text{b} > \text{x}\),
- \(1 < y_2 < y_3 < \cdots < y_{n-3}\) is the rightmost increasing run,
- \(m \geq 0\) is the number of empty boxes between \(\text{b}\) and \(\text{x}\).

Then we have the following.

1. \(X\) first reaches steady state after we apply \(m + 1\) additional BBS moves; that is, \(\text{BB}^m(X)\) is not a steady-state configuration, but \(\text{BB}^{m+1}(X)\) is. In other words, \(\text{sst}(w) = t + m + 1\).

(2) If \(\text{a} < \text{x}\), then \(\text{SD}(w) = \begin{array}{cccc}
\text{a} & \text{x} & \text{b} \\
\end{array}\); otherwise, \(\text{SD}(w) = \begin{array}{cccc}
\text{y}_2 & \text{y}_3 & \ldots \\
\text{x} & \text{b} & \text{a} \\
\end{array}\). In either case, \(\text{SD}(w)\) is standard, that is, \(w\) is a \text{(BBS) good permutation.}

**Proof.** By Theorem 5.5, the rightmost increasing run \(1y_2y_3 \ldots y_{n-3}\) is the first soliton.

If \(m > 0\), we apply \(m\) additional BBS moves to \(X\). At each BBS move, the first soliton will move forward \(n - 3 \geq 2\) boxes and the increasing block \(\text{ab}\) will move forward 2 boxes, and the singleton block \(\text{x}\) will move forward 1 box, so that the number of spaces between \(\text{ab}\) and \(\text{x}\) decreases by 1 after each BBS move. The two blocks \(\text{ab}\) and \(\text{x}\) touch in the configuration

\[
\text{BB}^m(X) = \ldots \text{a} \text{b} \text{x} \ldots
\]

Since \(\text{x} < \text{b}\), we have

\[
\text{BB}^{m+1}(X) = \begin{cases}
\ldots \text{b} \text{ax} \ldots & \text{if } \text{a} < \text{x} \\
\ldots \text{a} \text{xb} \ldots & \text{if } \text{x} < \text{a} \\
\text{soliton} & \text{soliton}
\end{cases}
\]

Let \(T\) be the configuration array of \(\text{BB}^{m+1}(X)\) (see Section 2.2). If there is at least one empty box between these three balls and the first soliton, the inequalities involving the numbers \(a, b, x, 1,\) and \(y_2\) guarantee that \(T\) is a standard skew tableau whose rows are weakly
decreasing in length. If there is no gap between these three balls and the first soliton, we must have \( w \in S_5 \) where
\[
BB^{m+1}(w) = \begin{cases} 
\ldots b \stackrel{\text{soliton}}{\longrightarrow} 1 y_2 & \text{if } a < x \\
\ldots a \stackrel{\text{soliton}}{\longrightarrow} b \rightarrow 1 y_2 & \text{if } x < a
\end{cases}
\]

If \( a < x \), we claim that \( y_2 < x \). Otherwise, we would have \( a < x < y_2 \), making \( I(BB^{m+1}(w)) \geq 3 \), contradicting the fact that \( 1 y_2 \) is the first soliton. By similar argument, if \( x < a \), we must have \( y_2 < b \).

Therefore, \( T \) is a standard skew tableau whose rows are weakly decreasing in length. Thus \( BB^{m+1}(X) \) is in steady state by Proposition 2.6. Since the order that the balls appear in \( BB^m(X) \) is different than in \( BB^{m+1}(X) \), we know that \( BB^m(X) \) is not yet in steady state. □

**Theorem 7.5.** If the RS partition of \( w \) is \( (n-3, 2, 1) \), then \( sst(w) \leq n - 3 \).

**Proof.** Suppose \( w \in S_n \) and with RS partition \( (n-3, 2, 1) \). Lemma 7.3 tells us that \( sh(SD(w)) \) is either \( (n-3,1,1,1) \) or \( (n-3,2,1) \). If \( sh(SD(w)) = (n-3,1,1,1) \), then by Theorem 6.1 we have \( sst(w) \leq 1 \). So suppose we have
\[
sh(SD(w)) = (n-3,2,1). \tag{7.2}
\]

At time \( t = 0 \), let the \( n \) balls \( w_1 w_2 \ldots w_n \) of \( w \) be in boxes 1 through \( n \). We apply one BBS move to \( w \) and consider all possibilities for the configuration \( BB^1(w) \) at time \( t = 1 \). By Theorem 5.5, we know the first soliton has been formed by \( t = 1 \), so we only need to consider the possibilities for the remaining three balls. By Remark 2.3, the number of rows in \( ID(BB^1(w)) \) is equal to that of \( SD(w) \), so \( ID(BB^1(w)) \) has three rows. Thus, the remaining three balls form a length-2 increasing run \( ab \) and a length-1 (singleton) increasing run \( x \).

If the length-1 block is to the left of the length-2 block at \( t = 1 \), then \( BB^1(w) \) is already in steady state because Theorem 5.5 tells us that the rightmost soliton won’t interact with the three balls after \( t = 1 \). Therefore, \( sst(w) \leq 1 \).

So suppose the length-2 block is to the left of the length-1 block at \( t = 1 \), that is,
\[
BB^1(w) = \begin{cases} 
\underbrace{e \ldots e}_{k \text{ copies of } e's} \underbrace{a b \ldots e}_{m \text{ copies of } e's} \underbrace{x \ldots e}_{\ell \text{ copies of } e's} \underbrace{1 y_2 y_3 \ldots y_{n-3}}_{n \text{ boxes}} & \text{first soliton}
\end{cases}
\]

where
- \( a < b \)
- \( a \) is in box \( k+1 \)
- \( m \geq 0 \) is the number of empty boxes between \( b \) and \( x \),
- \( \ell \geq 0 \) is the number of empty boxes between \( x \) and the ball 1.

First, observe that \( x < b \). Otherwise, we would have \( a < b < x \), and eventually the increasing run \( ab \) would catch up to \( x \), forming a length-3 soliton \( abx \); this would mean that \( sh(SD(w)) = (n-3,3) \), contradicting (7.2). Thus, \( BB^1(w) \) is of the form (7.1) in Lemma 7.4, so

\[
sst(w) = 1 + m + 1 = m + 2.
\]

Finally, observe that \( k \geq 2 \) because \( a < b \). We also know that \( k + m + 3 + \ell = n \) because ball 1 is in box \( n+1 \) at time 1. Putting these together, we have

\[
m = n - k - 3 - \ell \leq n - 2 - 3,
\]
\[ m + 2 \leq n - 3, \]
proving \( \text{sst}(w) \leq n - 3. \]

7.2. **Tableaux with increasing steady-state times via Bender–Knuth involution.** In this section, we create a sequence of \( n - 2 \) good tableaux whose steady-state times are from 0 to \( n - 3 \).

**Definition 7.6** (Bender–Knuth involution). Let \( T \) be a standard tableau with shape \( \lambda \) and size \( n \). Then for each \( i \in \{1, \ldots, n - 1\} \), \( \sigma_i \) is a map from the set of all standard tableaux of shape \( \lambda \) to itself. The map \( \sigma_i \) swaps the numbers \( i \) and \( i + 1 \) in \( T \) if the tableau resulting from switching \( i \) and \( i + 1 \) is a standard tableau. If the tableau resulting from switching \( i \) and \( i + 1 \) is not a standard tableau then \( \sigma_i(T) = T \).

**Example 7.7.** For instance,
\[
T = \begin{array}{ccc}
1 & 3 & 6 \\
2 & 5 & \\
4 & & \\
\end{array} \quad \neq \quad \sigma_2(T) = \begin{array}{ccc}
1 & 2 & 6 \\
3 & 5 & \\
4 & & \\
\end{array}
\]
but \( \sigma_3(\sigma_2(T)) = \sigma_2(T) \).

Corollary 6.8 and Proposition 6.13 tell us that all noncrossing involutions have steady-state time 0 or 1 and that most noncrossing involutions are bad. In combinatorics, the “noncrossing” objects and the “nonnesting” objects are often equinumerous, so it is natural to ask for a nonnesting analog of these results. The next proposition tells us that, for \( t \in \{0, \ldots, n - 3\} \), there is a “nonnesting” involution \( W_t \) with steady-state time \( t \).

**Proposition 7.8.** Let
\[
Q_0 := Q_0(n) = \begin{array}{ccc}
1 & 3 & 6 & 7 & 8 & \ldots & n \\
2 & 5 & & & & & \\
4 & & & & & & \\
\end{array}
\]
Then we have the following.
- \( \text{sst}(Q_0) = 0 \)
- \( \text{sst}(\sigma_2(Q_0)) = 1 \)
- \( \text{sst}(\sigma_k \ldots \sigma_5 \sigma_4 \sigma_2(Q_0)) = k - 2 \), for each \( k = 4, 5, \ldots, n - 1 \).

Furthermore, each tableau in the sequence of tableaux
\[
Q_0, \sigma_2(Q_0), \sigma_4 \ldots \sigma_5 \sigma_4 \sigma_2(Q_0)
\]
is (BBS) good.

**Proof.** The involution \( W_0 = \text{RS}^{-1}(Q_0, Q_0) \) is \((14)(35)\) in cycle notation and 4251367\ldots n in one-line notation. Since the latter is the row reading word of a standard tableau (namely, \( Q_0 \)), Remark 2.8 tells us that \( W_0 \) has steady-state time 0.

Next, consider
\[
\sigma_2(Q_0) = \begin{array}{ccc}
1 & 2 & 6 & 7 & \ldots & n \\
3 & 5 & & & & \\
4 & & & & & \\
\end{array} \quad \sigma_4 \sigma_2(Q_0) = \begin{array}{ccc}
1 & 2 & 6 & 7 & \ldots & n \\
3 & 4 & & & & \\
5 & & & & & \\
\end{array}
\]
\(\sigma_5\sigma_4\sigma_2(Q_0) = \begin{bmatrix}
1 & 2 & 5 & 7 & \ldots & n \\
3 & 4 \\
6 
\end{bmatrix} \)

By performing the inverse RS algorithm, we see that the involutions whose RS tableaux are \(\sigma_2(Q_0)\), \(\sigma_4\sigma_2(Q_0)\), and \(\sigma_5\sigma_4\sigma_2(Q_0)\) are \((14)(25)\), \((13)(25)\), and \((13)(26)\), respectively. Their steady-state times are 1, 2, and 3, respectively.

We now calculate the steady-state time for the rest of the tableaux in this sequence. Fix \(6 \leq k \leq n - 1\), and let

\[Q_k := \sigma_k \sigma_{k-1} \ldots \sigma_6 \sigma_5 \sigma_4 \sigma_2(Q_0) = \begin{bmatrix}
1 & 2 & 5 & 6 & \ldots & k & k+2 & \ldots & n \\
3 & 4 \\
k+1 
\end{bmatrix} \]

Its corresponding involution is \(W_k := \text{RS}^{-1}(Q_k, Q_k) = (13)(2, k+1)\). We will show that \(W_k\) has steady-state time \(k - 2\). The configuration at time \(t = 0\) is the one-line notation of \(W_k\):

\[3 \ 4 \ 5 \ 6 \ldots \ k \ 2 \ (k+2) \ldots n \]

At \(t = 1\) we have the configuration

\[\text{BB}^1(W_k) = ee \ 3(k+1) \ \underbrace{e \ldots e}_{k-4 \text{ copies}} \ \underbrace{3 \ldots 4 \ldots 1256 \ldots k(k+2) \ldots n}_{n-k-1 \text{ copies}} \]

which is of the form given in (7.1) in Lemma 7.4. Therefore, \(\text{sst}(W_k) = 1 + (k - 4) + 1 = k - 2\) and \(W_k\) is good. Indeed, we have

\[\text{BB}^{k-4}(W_k) = 3(k+1)e4e \ldots 1256 \ldots k(k+2) \ldots n \]
\[\text{BB}^{k-3}(W_k) = 3(k+1)4e \ldots 3e1256 \ldots k(k+2) \ldots n \]
\[\text{BB}^{k-2}(W_k) = (k+1)34e \ldots 3e1256 \ldots k(k+2) \ldots n \]

so \(\text{BB}^{k-2}(W_k)\) is in steady state, but \(\text{BB}^{k-3}(W_k)\) is not; in addition, \(\text{SD}(W_k) = \text{P}(W_k) = Q_k\), so \(Q_k\) is good.

**Example 7.9.** Consider \(w = 452361\). Using Proposition 7.8, we can create a sequence of tableaux that have steady-state times 0, 1, 2, and 3:

\[Q_0 = \begin{bmatrix}
1 & 3 & 6 \\
2 & 5 \\
4 
\end{bmatrix}, \sigma_2(Q_0) = \begin{bmatrix}
1 & 2 & 6 \\
3 & 5 \\
4 
\end{bmatrix}, \sigma_4\sigma_2(Q_0) = \begin{bmatrix}
1 & 2 & 6 \\
3 & 4 \\
5 
\end{bmatrix}, \sigma_5\sigma_4\sigma_2(Q_0) = \begin{bmatrix}
1 & 2 & 5 \\
3 & 4 \\
6 
\end{bmatrix} \]

The corresponding involutions are

\((14)(35), \ (14)(25), \ (13)(25), \ \text{and} \ (13)(26)\) in cycle notation, and

\(425136, \ 453126, \ 351426, \ \text{and} \ 361452\) in one-line notation,

in this order.

8. Further Directions

Recall that a permutation \(w\) is \((BBS)\) good if \(\text{SD}(w)\) is standard (equivalently, \(\text{SD}(w) = \text{P}(w)\), due to Theorem 4.8). If a permutation is not good, let us call it bad.
8.1. Classical permutation patterns. A permutation, or pattern, $\sigma$ is said to be contained in, or to be a subpermutation of, another permutation $w$ if $w$ has a (not necessarily contiguous) subsequence whose elements are in the same relative order as $\sigma$, alternatively, $w$ has a subsequence whose standardization is equal to $\sigma$. If $w$ does not contain $\sigma$, we say that $w$ avoids $\sigma$. For example, $314592687$ contains $1423$ because the subsequence $4968$ (among others) is ordered in the same way as $1423$. On the other hand, $314592687$ avoids $3241$ since $314592687$ has no subsequence ordered in the same way as $3241$. For more details, see for example the note [Bev15].

The above notion of pattern containment and pattern avoidance is sometimes referred to as classical. It turns out that classical pattern avoidance is too restrictive to be used to find all good permutations. The following shows that there are good permutations which contain bad patterns.

Example 8.1. A good permutation may have a subpattern which is not good.

a.) The permutation $25143$ is good, but it has a subpermutation $2143$ which is bad.

b.) The permutation $35142$ is good, but its subpermutation $3142$ is bad.

c.) Let $w = 42513$, which is a good permutation, and let $\sigma = 4253$, a subsequence of $w$. The standardization of $\sigma$ is $3142$, which is a bad permutation.

Remark 8.2. Example 8.1 shows that the good permutations are not closed under classical pattern containment. This means that the set of good permutations cannot be characterized by a set of classically avoided patterns.

Although it is impossible to characterize good permutations using classical pattern avoidance, we can give an instance where classical pattern avoidance can be used to find a (proper) subset of good permutations. The following is straightforward to prove using a localized version of Greene’s theorem (see [DGGRS21, Section 2.2]) and Theorem 4.8.

Proposition 8.3. If $w$ avoids both the classical pattern $2143$ and $3142$, then $w$ is good.

Remark 8.4. The converse of Proposition 8.3 is false. As shown in Example 8.1, there are good permutations which have the classical pattern $2143$ or $3142$.

8.2. Consecutive permutation patterns. A permutation, or pattern, $\sigma$ is said to be a consecutive pattern of another permutation $w$ if $w$ has a consecutive subsequence whose elements are in the same relative order as $\sigma$. Otherwise, $w$ is said to avoid $\sigma$ as a consecutive pattern. For example, $314592687$ contains $2413$ because the subsequence $5926$ is ordered in the same way as $2413$. On the other hand, $314592687$ avoids $321$ since $314592687$ has no consecutive subsequence ordered in the same way as $321$ (although $314592687$ contains the classical pattern $321$).

We conjecture that good permutations are closed under consecutive pattern containment; that is, if a permutation is good, then any consecutive subpermutation is also good.

Conjecture 8.5. If a permutation $w$ is good, then the standardization of every consecutive subpattern of $w$ is also good.

8.3. Motzkin numbers. The $n$th Motzkin number is the number of ways to draw nonintersecting chords between $n$ labeled points on a circle. They also count the number of Motzkin paths, 4321-avoiding involutions, along with many other objects [OEIS, A001006].

Conjecture 8.6. The number of size-$n$ good tableaux is equal to the $n$th Motzkin number.
Remark 8.7. Since drawing nonintersecting chords between labeled points on a circle is equivalent to determining a noncrossing involution, we get that the number of noncrossing involutions in $S_n$ is equal to the $n$th Motzkin number. However, Proposition 6.13 shows that some noncrossing involutions are good and some noncrossing involutions are not, so the set of good involutions is not equal to the set of noncrossing involutions.

It is also known that the number of nonnesting involutions in $S_n$ is equal to the $n$th Motzkin number. Proposition 6.13 illustrates that the set of good involutions is not equal to the set of nonnesting involutions.

Acknowledgments

This work was supported by the Summer Undergraduate Math Research at Yale (SUMRY 2021) program and the NSF (REU Site grant DMS-2050398). We thank Sergi Elizalde and Joel Lewis for helpful discussions, and Su Ji Hong, Matthew Li, Raina Okonogi-Neth, Mykola Sapronov, Dash Stevanovich, and Hailey Weingord for useful input and feedback during SUMRY 2022. This research also benefited from the open-source software SageMath [Dev21].

References

[Bev15] David Bevan, Permutation patterns: basic definitions and notation, 2015, Preprint arXiv:1506.06673.
[Dev21] The Sage Developers, Sagemath, the Sage Mathematics Software System (Version 9.3), 2021, https://www.sagemath.org.
[DGGRS21] Ben Drucker, Eli Garcia, Emily Gunawan, Aubrey Rumbolt, and Rose Silver, RSK tableaux and box-ball systems, 2021, Preprint arXiv:2112.03780.
[Fuk04] Kaori Fukuda, Box-ball systems and Robinson-Schensted-Knuth correspondence, Journal of Algebraic Combinatorics 19 (2004), no. 1, 67–89.
[Ful96] William Fulton, Young tableaux: With applications to representation theory and geometry, London Mathematical Society Student Texts, Cambridge University Press, 1996.
[GHKU21] Maria Gillespie, Jacob Hocevar, Ananya Kulshrestha, and Kosha Upadhyay, Iterating the RSK bijection, Involve 14 (2021), no. 3, 475–494.
[Gre74] Curtis Greene, An extension of Schensted’s theorem, Advances in Math. 14 (1974), 254–265.
[Hai92] Mark D. Haiman, Dual equivalence with applications, including a conjecture of Proctor, Discrete Math. 99 (1992), no. 1-3, 79–113.
[LLPS19] Joel Lewis, Hanbaek Lyu, Pavlo Pylyavskyy, and Arnab Sen, Scaling limit of soliton lengths in a multicolor box-ball system, 2019, Preprint arXiv:1911.04458.
[OEIS] OEIS Foundation Inc. (2022), The On-Line Encyclopedia of Integer Sequences, http://oeis.org, [Online; accessed 4-September-2022].
[Pos09] Jacob Post, Combinatorics of arc diagrams, Ferrers fillings, Young tableaux and lattice paths, http://summit.sfu.ca/item/9667, 2009.
[Sag01] Bruce E. Sagan, The symmetric group, second ed., Graduate Texts in Mathematics, vol. 203, Springer-Verlag, New York, 2001, Representations, combinatorial algorithms, and symmetric functions.
[Sch61] C. Schensted, Longest increasing and decreasing subsequences, Canadian J. Math. 13 (1961), 179–191.
[Tak93] Daisuke Takahashi, On some soliton systems defined by using boxes and balls, Proceedings of the international symposium on nonlinear theory and its applications (NOLTA’93), 1993, pp. 555–558.
[TM97] Daisuke Takahashi and Junta Matsukidaira, Box and ball system with a carrier and ultradiscrete modified KdV equation, Journal of Physics A: Mathematical and General 30 (1997), no. 21, L733.
[TS90] Daisuke Takahashi and Junkichi Satsuma, A soliton cellular automaton, J. Phys. Soc. Japan 59 (1990), no. 10, 3514–3519.