

Bijections for Generalized Wilf Equivalences

Melanie Ferreri

Department of Mathematics, College of William & Mary, Williamsburg, VA, 23185, USA
 Email: mjferreri@wm.edu

Received: October 16, 2024, **Accepted:** March 16, 2025, **Published:** April 4, 2025
 The authors: Released under the CC BY-ND license (International 4.0)

ABSTRACT: Starting with an inclusion-exclusion proof of a combinatorial identity, a direct bijection can be produced using recursive subtraction (sometimes with a direct combinatorial description). We apply this method to identities for generalized Wilf equivalences among consecutive patterns in inversion sequences, giving direct bijective proofs of some generalized Wilf equivalences shown by Auli and Elizalde. We also give new bijective proofs of a stronger relation among some consecutive patterns.

Keywords: Inversion sequences; Inclusion-exclusion; Patterns; Wilf equivalence
2020 Mathematics Subject Classification: 05A05; 05A19

1. Introduction

The study of patterns in inversion sequences was initiated by Corteel, Martinez, Savage, and Weselcouch [3], as well as Mansour and Shattuck [6]. Auli and Elizalde [1, 2] analogously studied *consecutive* patterns in inversion sequences, and characterized generalized Wilf equivalences among consecutive patterns of length 4. In many cases they gave direct bijective proofs. In others, they gave a collection of bijections among supersets of the desired sets and showed the equivalences by applying inclusion-exclusion.

A proof of this type can be “bijectivized” by recursively subtracting the maps between supersets until we have mappings directly between the exact desired sets. For those equivalences that they do not prove with direct bijections, we extend their work by using recursively defined bijections to produce matchings that exhibit the equivalences.

We describe the recursive maps obtained via their maps and give an iterative interpretation for certain cases, which yields a bijective proof of these equivalences.

For the remaining equivalences not covered in the previous case, we show the recursive map in the same way; although for this case, we did not obtain a simpler direct description of the resulting map. However, the resulting examples suggest that a stronger result can be shown. We find a new bijective algorithm that proves a stronger relation among patterns that implies super strong Wilf equivalence.

The takeaway is that in this setting, it is often worth considering the implicit bijection that results from the inclusion-exclusion proof. As we’ve seen in this case, this can lead to a direct description of a bijection showing the result and can give insight toward possible stronger results.

Much of the content of this paper is from a project described in the author’s PhD thesis, [5].

2. Equivalence relations for patterns in inversion sequences

Let $\pi = \pi(1)\pi(2)\cdots\pi(n) \in \mathfrak{S}_n$. An *inversion* is a pair (i, j) such that $i < j$ and $\pi(i) > \pi(j)$; that is, a pair of indices where the elements of $[n]$ are “out of order.” An *inversion sequence* is an integer sequence $\epsilon = \epsilon_1\epsilon_2\cdots\epsilon_n$ such that $0 \leq \epsilon_i < i$ for all $1 \leq i \leq n$. We denote by I_n the set of inversion sequences of length n . This set bijects with \mathfrak{S}_n ; for a permutation π , we can view the i th entry of an inversion sequence as counting the number of entries to the left of $\pi(i)$ which are greater than $\pi(i)$.

A *pattern* is a sequence $p = p_1p_2\cdots p_r$ with $p_i \in \{0, 1, \dots, r - 1\}$ for all $1 \leq i \leq r$. We also require that j can only appear in p if $j - 1$ also appears.

If $w = w_1w_2 \cdots w_k$ is a word over the integers, its *reduction* is given by replacing all occurrences of its i th smallest entry by $i - 1$ for all i .

A sequence ϵ *contains* the pattern p if there is a subsequence $\epsilon_{i_1}, \epsilon_{i_2}, \dots, \epsilon_{i_r}$ whose reduction is p . Otherwise, ϵ *avoids* p . An inversion sequence ϵ contains the *consecutive pattern* $p = p_1p_2 \dots p_r$ if there is a consecutive subsequence of ϵ whose reduction is p . We denote that a pattern is consecutive by underlining it. If the first term of the subsequence is ϵ_i , we say the subsequence is an *occurrence* of p in position i , or simply that i is an occurrence of p in ϵ . A *vincular* or *generalized* pattern is a pattern in which some subset of the entries may be required to appear consecutively, while other entries may not. Such patterns are often written with the subsequences that are meant to appear consecutively underlined. We denote by $I_n(p)$ the set of inversion sequences of length n which avoid p .

Example 2.1. $\epsilon = 00123021$ contains 012. It also contains 210 (witness: $\epsilon_5\epsilon_7\epsilon_8$), but avoids 210. That is, it has a decreasing subsequence of length 3 but not a consecutive decreasing subsequence of length 3. So $\epsilon \in I_8(\underline{210})$.

Denote by $\text{Em}(p, \epsilon)$ the set of all occurrences of p in ϵ , so $\text{Em}(p, \epsilon)$ is a list of indices that tell us the locations of the first positions of all consecutive subsequences with reduction p appearing in ϵ . The consecutive patterns p, q are said to be

- *Wilf equivalent* if $|I_n(p)| = |I_n(q)|$ for all n .
- *strongly Wilf equivalent* if for all n and m , the number of inversion sequences of length n containing m occurrences of p is equal to the number of inversion sequences of length n containing m occurrences of q .
- *super strongly Wilf equivalent* if

$$|\{\epsilon \in I_n : \text{Em}(p, \epsilon) = T\}| = |\{\epsilon \in I_n : \text{Em}(q, \epsilon) = T\}|$$

for all $n, T \subseteq [n]$.

In [1], Auli and Elizalde show generalized Wilf equivalences between all 75 consecutive patterns of length 4. It turns out that all of their demonstrated equivalences among consecutive patterns of length 4 are super strong, and they conjecture that for consecutive patterns of length r , Wilf equivalence implies strong Wilf equivalence. Here we will write $p \sim q$ when p and q are super strongly Wilf equivalent.

A consecutive pattern p is *non-overlapping* if it cannot overlap itself in more than one entry. That is, if p is a pattern of length r , no two occurrences of p can be less than $r - 1$ apart.

Example 2.2. 1. The consecutive pattern 0102 is not non-overlapping; the sequence 010203 has an occurrence of this pattern at positions 1 and 3; these occurrences overlap in two entries (the third and fourth entries in the sequence).

2. The consecutive pattern 1000 is non-overlapping. Suppose we want to construct a sequence ϵ with 1000 occurring in both positions i and $i + 1$. Since the pattern occurs at i , we have $\epsilon_i > \epsilon_{i+1} = \epsilon_{i+2} = \epsilon_{i+3}$. But also, since the pattern occurs at $i + 1$, we have $\epsilon_{i+1} > \epsilon_{i+2}$; which is a contradiction. Similarly, we could not have this pattern occurring at both i and $i + 2$. So the pattern can overlap itself in at most one entry.

We say the consecutive patterns p and q of length r are *mutually non-overlapping* if occurrences of p and q in an inversion sequence cannot overlap in more than one entry.

We also consider exchanging occurrences of p for occurrences of q in a sequence. The process of changing an occurrence of p to an occurrence of q is given in [1] as follows: Let $p = \underline{p_1p_2 \cdots p_r}$ and $q = \underline{q_1q_2 \cdots q_r}$ be consecutive patterns which agree on their first and last entries, and which both contain all elements in $\{0, 1, \dots, d\}$ for some d , which is the maximal entry in both sequences. Let $\epsilon \in I_n$ such that ϵ has an occurrence of p at position i . Then define

$$f : \{0, \dots, d\} \rightarrow \{\epsilon_i, \dots, \epsilon_{i+r-1}\}$$

(with the latter set excluding repetition) to be the order-preserving bijection such that

$$\epsilon_i \cdots \epsilon_{i+r-1} = f(p_1) \cdots f(p_r).$$

Then define a new sequence $\epsilon' = \epsilon'_1 \cdots \epsilon'_n$ where $\epsilon'_j = \epsilon_j$ for $j \neq i, \dots, i + r - 1$; i.e. when j is not a position that coincides with the occurrence of p at position i . For the remaining positions, assign

$$\epsilon'_i \epsilon'_{i+1} \cdots \epsilon'_{i+r-1} = f(q_1)f(q_2) \cdots f(q_r).$$

This operation is a *change* of the occurrence of p at position i in ϵ to an occurrence of q . If the resulting sequence ϵ' is an inversion sequence, this change is said to be *valid*.

Example 2.3. Let $p = \underline{0102}$, $q = \underline{0112}$, and $\epsilon = 0101213$. The sequence ϵ has an occurrence of p at position 4. To change this to an occurrence of q , we let $f(0) = 1$, $f(1) = 2$, and $f(2) = 3$. Then we write $\epsilon' = 010f(0)f(1)f(1)f(2) = 0101223$. This yields a sequence with an occurrence of q at position 4.

A pattern $p = \underline{p_1 \dots p_r}$ is said to be *changeable* for $q = \underline{q_1 \dots q_r}$ if, for all $i \in \{1, \dots, r\}$,

$$q_i \leq \max(\{p_j : 1 \leq j \leq i\} \cup \{p_j - j + 1 : i < j \leq r\}).$$

Two patterns p, q are *interchangeable* if they are changeable for one another.

Lemma 4.8 of [1] shows that p is changeable for q if and only if, for any inversion sequence ϵ and any occurrence of p in ϵ , it is valid to change the occurrence of p to an occurrence of q . In other words, one can always change an occurrence of p to an occurrence of q while keeping ϵ as an inversion sequence.

For patterns that are non-overlapping, mutually non-overlapping, and interchangeable, Auli and Elizalde [1] show their Wilf equivalence via a bijection which simultaneously changes all occurrences of p into q and vice versa — this can be done at once because the occurrences cannot possibly overlap. This takes care of the equivalences (ii) – (vii), (ix), (xi), (xii), and (xiv) in Theorem 2.2 of [1] (also included in Theorem 2.1 below).

This actually proves an even stronger statement than super strong Wilf equivalence. These bijections show

$$|\{\epsilon \in I_n : \text{Em}(p, \epsilon) = S, \text{Em}(q, \epsilon) = T\}| = |\{\epsilon \in I_n : \text{Em}(p, \epsilon) = T, \text{Em}(q, \epsilon) = S\}|$$

for all n and $S, T \subseteq [n]$. When two patterns p, q satisfy this relation, we will say that p and q are *reciprocal*, and we will denote this by $p \stackrel{R}{\sim} q$. As pointed out in [1], this is equivalent to saying that the sets of positions where p and q occur have a symmetric joint distribution over I_n . Although $\stackrel{R}{\sim}$ is not an equivalence relation, we have that if p and q are reciprocal, then $p \sim q$. We will revisit this definition in Section 4.1.

After we are done, we will have the following updated list of equivalences and reciprocal relations, where those with asterisks are newly proven bijectively:

Theorem 2.1 (Extension of Theorem 2.2 of [1]). *A complete list of the generalized Wilf equivalences between consecutive patterns of length 4 is as follows:*

- | | | |
|--|---|---|
| (i) * $\underline{0102} \stackrel{R}{\sim} \underline{0112}$. | (vi) $\underline{1200} \sim \underline{1210} \sim \underline{1220}$. | (xi) $\underline{2001} \sim \underline{2011} \sim \underline{2101} \sim \underline{2201}$. |
| (ii) $\underline{0021} \sim \underline{0121}$. | (vii) $\underline{0211} \sim \underline{0221}$. | (xii) $\underline{2012} \sim \underline{2102}$. |
| (iii) $\underline{1002} \sim \underline{1012} \sim \underline{1102}$. | (viii) * $\underline{1000} \sim \underline{1110}$. | (xiii) * $\underline{2010} \stackrel{R}{\sim} \underline{2110} \stackrel{R}{\sim} \underline{2120}$. |
| (iv) $\underline{0100} \sim \underline{0110}$. | (ix) $\underline{1001} \sim \underline{1011} \sim \underline{1101}$. | (xiv) $\underline{3012} \sim \underline{3102}$. |
| (v) $\underline{2013} \sim \underline{2103}$. | (x) * $\underline{2100} \sim \underline{2210}$. | |

All of the other (super strong) Wilf equivalence classes contain just one pattern.

For the equivalences without asterisks demonstrated by Auli and Elizalde, the patterns are shown to be reciprocal as well, as explained in [1]. The remaining (super strong) Wilf equivalences are proven by Auli and Elizalde in [1] using an inclusion-exclusion argument with bijections $\{\epsilon \in I_n : \text{Em}(p, \epsilon) \supseteq S\} \rightarrow \{\epsilon \in I_n : \text{Em}(q, \epsilon) \supseteq S\}$ which change occurrences of p at positions S to occurrences of q .

Applying the method of bijective inclusion-exclusion as described in [4], we can use these maps to obtain the direct bijection recursively.

3. Bijections defined recursively

We will first focus on nonoverlapping consecutive patterns; in particular, the equivalences $\underline{1000} \sim \underline{1110}$ and $\underline{2100} \sim \underline{2210}$. For consecutive patterns p and q , we are interested in finding a map which takes a sequence ϵ , with p occurring exactly at a set of indices T , and generate a new sequence ϵ' , which has q occurring at exactly the indices in T .

For a consecutive pattern p and set $T \subseteq [n]$, let

$$I_{n,p}(\supseteq T) = \{\epsilon \in I_n : \text{Em}(p, \epsilon) \supseteq T\}.$$

Similarly define

$$I_{n,p}(= T) = \{\epsilon \in I_n : \text{Em}(p, \epsilon) = T\}.$$

Suppose p and q are nonoverlapping consecutive patterns, and suppose there is a bijection

$$\varphi_{\geq T} : I_{n,p}(\supseteq T) \rightarrow I_{n,q}(\supseteq T)$$

which changes the occurrences of p at positions T into occurrences of q at positions T . For our purposes, the bijection $\varphi_{\geq T}$ will be given by applying the change operation of Auli and Elizalde [1] described in Section 1 to each occurrence of p indexed by T .

For each $\epsilon \in I_{n,p}(= T)$, we can define $\varphi_{=T}(\epsilon)$ recursively as follows:

The base cases occur when T is a maximal set of occurrences for p or q ; that is, when there is no inversion sequence whose set of occurrences of p or q properly contains T . In this case, the map $\varphi_{\geq T}$ will actually be a matching between $I_{n,p}(= T)$ and $I_{n,q}(= T)$, since T is maximal so $I_{n,p}(\supseteq T) = I_{n,p}(= T)$.

For a non-maximal set T , suppose we have bijections $\varphi_{=S}$ for all S properly containing T . Then define

$$\begin{aligned} \varphi_{=T} : (\epsilon \in I_{n,p}(= T)) &\xrightarrow{\varphi_{\geq T}} (\epsilon' \in I_{n,q}(= S) \subseteq I_{n,q}(\supseteq T)) \xrightarrow{\varphi_{=S}^{-1}} (\epsilon'' \in I_{n,p}(= S)) \\ &\xrightarrow{\varphi_{\geq T}} (\epsilon''' \in I_{n,q}(= R)) \xrightarrow{\varphi_{=R}^{-1}, \varphi_{\geq T}}_{(\text{repeat until})} (\epsilon'' \dots' \in I_{n,q}(= T)) \end{aligned}$$

where $S = \text{Em}(q, \varphi_{\geq T}(\epsilon))$; i.e. the set of occurrences of q in the image of ϵ after applying $\varphi_{\geq T}$, and R is defined similarly as the set of occurrences of q in the sequence after applying the most recent arrow. By the hypothesis, each map applied is a bijection, so this gives a matching between $I_{n,p}(= T)$ and $I_{n,q}(= T)$.

Note: This process computes the subtraction as in [4] of the maps $\phi_{=S}$ for all $S \supseteq T$ from $\phi_{\geq T}$. A more in-depth explanation of this, along with some examples, can be found in Section 1.1 of [5].

Example 3.1. Suppose $p = 1000$ and $q = 1110$. The following examples illustrate how the recursive mapping works. The vertical lines indicate recursion depth, with the leftmost line corresponding to a depth of 0.

(a) $\epsilon = 001000$

We currently have p at $\{3\}$; we want q at $\{3\}$.

$\varphi_{=\{3\}}(001000)$:

| $\varphi_{\geq\{3\}}(001000) = 001110$

| q occurs at $\{3\}$, so we are done.

001110

We finally obtain $001000 \mapsto 001110$.

Since there was only one occurrence and it worked to just swap it, this works the same way on this sequence as the other bijections would.

(b) $\epsilon = 0021110$

We currently have p at $\{3\}$; we want q at $\{3\}$.

$\varphi_{=\{3\}}(0021110)$:

| $\varphi_{\geq\{3\}}(0021110) = 0022210$

| q occurs at $\{3\}$, so we are done.

0022210

We finally obtain $0021110 \mapsto 0022210$.

The initial sequence had another copy of q at position 4, but this got overwritten. Going the opposite direction, 0022210 maps to 0021110 (we just change the q to a p).

(c) $\epsilon = 002111110$

We currently have p at $\{3\}$; we want q at $\{3\}$.

$\varphi_{=\{3\}}(002111110)$:

| $\varphi_{\geq\{3\}}(002111110) = 002221110$

| q occurs at $\{3, 6\}$, so apply $\varphi_{=\{3,6\}}^{-1}$

| $\varphi_{=\{3,6\}}^{-1}(002221110)$:

| | $\varphi_{\geq\{3,6\}}^{-1}(002221110) = 002111000$

| | p occurs at $\{3, 6\}$, so we are done.

| 002111000

| $\varphi_{\geq\{3\}}(002111000) = 002221000$

| q occurs at $\{3\}$, so we are done.

002221000

We finally obtain $002111110 \mapsto 002221000$.

The following pictures show the underdiagonal lattice paths of the original sequence and the sequences after each application of $\varphi_{\geq\{3\}}$.

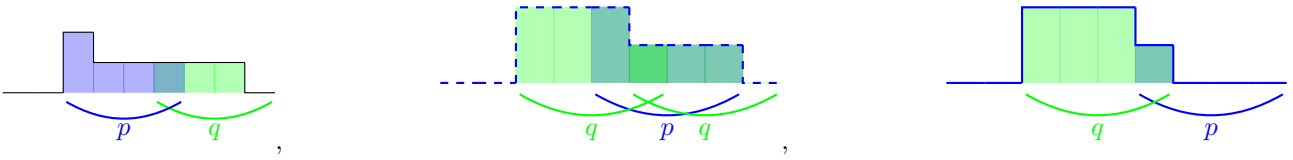


Figure 1: Underdiagonal lattice path diagrams of sequences 002111110, 002221110, 002221000.

(d) $\epsilon = 010001111110$

We currently have p at $\{2\}$; we want q at $\{2\}$.

$\varphi_{=\{2\}}(010001111110)$:

| $\varphi_{\geq\{2\}}(010001111110) = 011101111110$

| q occurs at $\{2, 10\}$, so apply $\varphi_{=\{2,10\}}^{-1}$

| $\varphi_{=\{2,10\}}^{-1}(011101111110)$:

| | $\varphi_{\geq\{2,10\}}^{-1}(011101111110) = 0100011111000$

| | p occurs at $\{2, 10\}$, so we are done.

| 0100011111000

| $\varphi_{\geq\{2\}}(0100011111000) = 0111011111000$

| q occurs at $\{2, 8\}$, so apply $\varphi_{=\{2,8\}}^{-1}$

| $\varphi_{=\{2,8\}}^{-1}(0111011111000)$:

| | $\varphi_{\geq\{2,8\}}^{-1}(0111011111000) = 0100011100000$

| | p occurs at $\{2, 8\}$, so we are done.

| 0100011100000

| $\varphi_{\geq\{2\}}(0100011100000) = 0111011100000$

| q occurs at $\{2, 6\}$, so apply $\varphi_{=\{2,6\}}^{-1}$

| $\varphi_{=\{2,6\}}^{-1}(0111011100000)$:

| | $\varphi_{\geq\{2,6\}}^{-1}(0111011100000) = 0100010000000$

| | p occurs at $\{2, 6\}$, so we are done.

| 0100010000000

| $\varphi_{\geq\{2\}}(0100010000000) = 0111010000000$

| q occurs at $\{2\}$, so we are done.

0111010000000

We finally obtain $010001111110 \mapsto 0111010000000$.

The underdiagonal lattice paths of the original sequence, and the sequences after each application of $\varphi_{\geq\{2\}}$, are shown in Figure 2.

3.1 Iterative interpretation

For nonoverlapping consecutive patterns, we can translate this into an iterative process.

Lemma 3.1. Let $\varphi_{\geq S}^{-1}$ be as previously defined, and let $\epsilon \in I_n$ such that $\text{Em}(q, \epsilon) = S$ and $\text{Em}(p, \epsilon) = U$. If p and q are non-overlapping consecutive patterns of length 4, then $\text{Em}(p, \varphi_{\geq S}^{-1}(\epsilon)) \subseteq S \cup U$.

Proof. We show that applying $\varphi_{\geq S}^{-1}$ (i.e., changing q at positions S to p) to ϵ cannot create new occurrences of p outside of S or U .

Applying $\varphi_{\geq S}^{-1}$ only affects the sequence entries coinciding with the pattern occurrences indexed by S , by changing those occurrences of q to occurrences of p . Suppose for contradiction that there was an occurrence p not indexed by U that was created in a position also not indexed by S . Then it would necessarily overlap the same entries as the q indexed by S that were previously there and were changed to p . (In the case of the consecutive patterns of length 4, each pair of non-overlapping Wilf equivalent patterns agree on the first and fourth entries, so the only entries changed would be the second or third entries of the previous q . Thus the overlap would have occurred on the second or third entries). This implies that we would have two copies of p overlapping in more than one entry, which contradicts the assumption that p is non-overlapping. So there cannot have been any “extra” occurrences of p created outside of S (or U , where there may have been preexisting occurrences of p); thus $\text{Em}(p, \varphi_{\geq S}^{-1}(\epsilon)) \subseteq S \cup U$. \square

Lemma 3.2. Let $\varphi_{=T}$ be the previously defined recursive bijection, and let $\epsilon \in I_n$ with $\text{Em}(p, \epsilon) = T$. If p and q are non-overlapping consecutive patterns of length 4, then the recursion depth will not exceed 1.

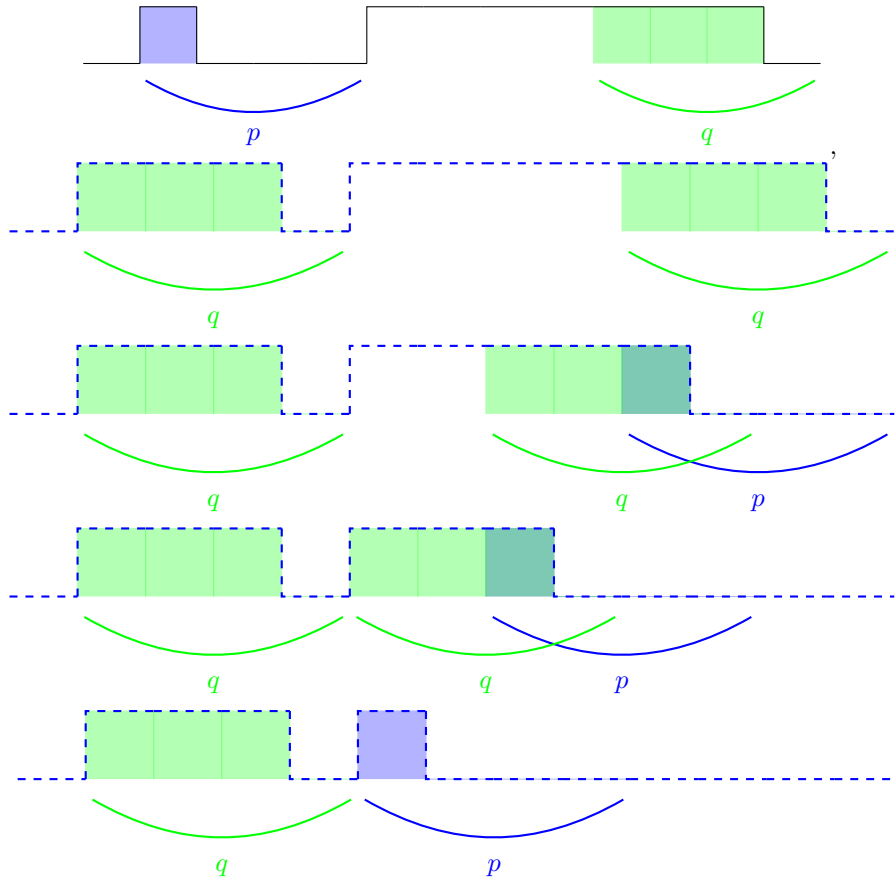


Figure 2: Underdiagonal lattice path diagrams of sequences 01000111111110, 01110111111110, 0111011111000, 0111011100000, 0111010000000.

Proof. We will show that after (i) turning all p at positions T to q , and then (ii) turning all q at positions $S \supseteq T$ to p , there will be no p at a position outside of S remaining in the sequence.

Suppose to the contrary that after step (ii), there is still some p at position $r \notin S$. Since $T \subseteq S$, this means $r \notin T$, so this occurrence of p at r was not in the original sequence. Then it was either created in step (i) when p was changed to q or in step (ii) when q was changed to p .

In step (i), all p were changed to q . If this had created an extra occurrence of p in the sequence, then the new occurrences of p would necessarily overlap one of the newly created occurrences of q at a position indexed by T . Then since the q overlapping p at r is at a position in $T \subseteq S$, in step (ii) this q would then be changed to a p , erasing the p at position r (e.g. $\underline{100000} \rightarrow \underline{1110000} \rightarrow \underline{100000}$). We know that this p will be erased because if it was preserved, then we would have two overlapping instances of p , but p is non-overlapping.

Otherwise, it must be the case that the new occurrence of p at r was created in step (ii). By Lemma 3.1, it is not possible for a new p to appear after step (ii) that was not either already in the sequence after step (i) or in a position where a q was after step (i). We have just checked that any newly created p from step (i) will be erased after step (ii), and the positions, where q appeared after step (i) are all indexed by S , so step (ii) cannot have created a new p at a position outside of S .

Thus, if a sequence ϵ had q at $S \supseteq T$ after the first arrow, then ϵ has p at exactly S after the second arrow. So $\varphi_{=S}^{-1}(\epsilon)$ will have been computed without increasing the recursion depth. \square

Theorem 3.1. *Let p and q be non-overlapping consecutive patterns of length 4. The previously defined recursive bijection φ matches the following iterative description. Let $\epsilon \in I_n$.*

1. Find $T := \text{Em}(p, \epsilon)$.
2. Change all occurrences of p in T to q ; let ϵ' be the resulting sequence.
3. Now let $S = \text{Em}(q, \epsilon')$. Change all of the occurrences of q in $S \setminus T$ to p .
4. Repeat step (c) until $S \setminus T = \emptyset$, so we now have a sequence ϵ' with q occurring exactly at T .

Proof. By Lemma 3.2, the recursion depth will never exceed 1 when computing $\varphi(\epsilon)$. So the mapping will continue to apply $\varphi_{=S}^{-1}$ and $\varphi_{\geq T}$, for each new set S containing T , until $T = S$.

Moreover, the initial occurrences of p at T will swap back and forth between p and q at each step. We know that the original occurrences of p will not be erased because this could only happen if some q appeared at position i overlapping the initial p and then was changed to a p , erasing some original occurrence. However, if this q appeared in the original sequence, then the first arrow would change the p to q , erasing the q at position i .

The q could not have appeared on the second arrow by the same reasoning as in the proof of Lemma 3.2 (changing p to q cannot create new occurrences of q outside of where the p were).

Therefore, the original occurrences of p at T will switch back and forth, and at the end will turn into occurrences of q at T , so we can view the bijection as changing the initial occurrences of p to occurrences of q and then changing only entries off of T . This matches exactly the iterative description. \square

The previous theorem gives a direct bijective proof of the super-strong Wilf equivalences $\underline{1000} \sim \underline{1110}$ and $\underline{2100} \sim \underline{2210}$.

Example 3.2. *Again let $p = \underline{1000}$, $q = \underline{1110}$. The sequence $\epsilon = 002111110$ has p at $T = \{3\}$. We change this to 002221110 and have q at $S = \{3, 6\}$. Then $S \setminus T = \{6\}$, so we change the q at 6 back to p , and obtain 002221000 . This now has $S = \{3\} = T$, so we are done.*

Going the other direction, 002221000 has q at $\{3\}$, so we change it to p and obtain 002111000 . Now we have p at $\{3, 6\}$ so we change the p at 6 back to q and obtain 002111110 .

4. Patterns that are not mutually non-overlapping

The remaining equivalences are (i) $\underline{0102} \sim \underline{0112}$ and (xiii) $\underline{2010} \sim \underline{2110} \sim \underline{2120}$. Auli and Elizalde prove these equivalences similarly as before, by constructing maps that exchange the patterns, and then arguing by inclusion-exclusion. The maps that exchange the patterns are slightly more complicated since these patterns can overlap themselves. The bijection for swapping overlapping copies of the pattern $\underline{0102}$ to overlapping copies of a pattern $\underline{0112}$ is given by Auli and Elizalde [1], as follows: Since $\underline{0102}$ can overlap itself in two entries, a string of overlapping occurrences of this pattern might look something like: $01020304\dots$. To get the pattern $\underline{0112}$ instead, this would then be changed into $01122334\dots$. Viewing these sequences as underdiagonal lattice paths, this can be represented by making the second “0” of each occurrence match the height of the entry to its left. This process changes all occurrences of $\underline{0102}$ in a sequence to $\underline{0112}$. For the relation (xiii), they state that similarly constructed maps can be used for changing between $\underline{2010}$, $\underline{2110}$, and $\underline{2120}$.

We can still define a bijection recursively for this case, the same way as before. However, the iterative algorithm from Theorem 3.1 does not match the recursive process in this case. For example, if we try to replicate the same iterative process we used before on $\epsilon = 010223$, we would obtain 011223 , which has q at $\{1, 3\}$, and then changing q at 3 back to p gives 011213 , so now we have q only at $\{1\}$, as desired. But also, applying the iterative process to $\epsilon = 010213$, which has p at $\{1\}$, we get 011213 , which also has q at position $\{1\}$, so we would stop. But now we’ve mapped two different sequences to the same sequence! Under the recursive map, we would instead send $010223 \mapsto 011203$ and $010213 \mapsto 011213$. Because the recursion depth can exceed 1 in this case, the argument for the iterative algorithm described in Theorem 3.1 does not apply.

While we do not have a more direct combinatorial interpretation of this bijection, in Section 4.1 we give an alternative bijection which has a more straightforward description.

Here we include examples of how the recursive bijection works for this case. Viewing the inversion sequences as underdiagonal lattice paths, we can visualize the steps of the recursive map as illustrated below.

Example 4.1. (a) *For $\epsilon = 010223$, the recursive map is computed as follows.*

Let $p = \underline{0102}$ and $q = \underline{0112}$. We currently have p at $\{1\}$; we want q at $\{1\}$.

$\varphi_{=\{1\}}(010223)$:

| $\varphi_{\geq\{1\}}(010223) = 011223$

| q occurs at $\{1, 3\}$, so apply $\varphi_{=\{1,3\}}^{-1}$

| $\varphi_{=\{1,3\}}^{-1}(011223)$:

| | $\varphi_{\geq\{1,3\}}^{-1}(011223) = 010203$

| | p occurs at $\{1, 3\}$, so we are done.

| 010203

| $\varphi_{\geq\{1\}}(010203) = 011203$

| q occurs at $\{1\}$, so we are done.

011203

We finally obtain $010223 \mapsto 011203$.

The following pictures show the underdiagonal lattice paths of the original sequence, and the sequences after each application of $\varphi_{\geq\{1\}}$.

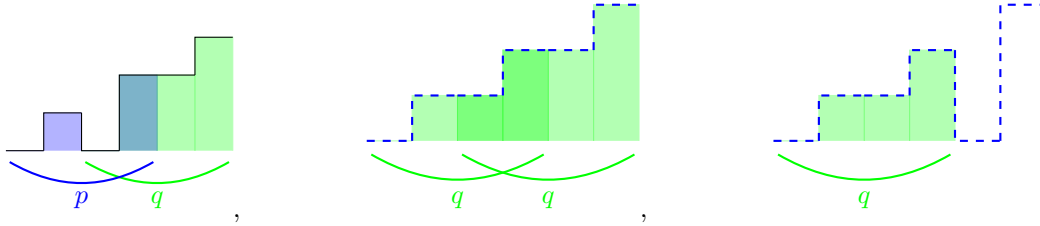


Figure 3: Underdiagonal lattice path diagrams for 010223, 011223, 011203.

(b) For $\epsilon = 010213$, we currently have p at $\{1\}$, and want q at $\{1\}$.

$\varphi_{=\{1\}}(010213)$:

| $\varphi_{\geq\{1\}}(010213) = 011213$

| q occurs at $\{1\}$, so we are done.

011213

We finally obtain $010213 \mapsto 011213$.

The following pictures show the underdiagonal lattice paths of the original sequence and the sequence after the application of $\varphi_{\geq\{1\}}$.

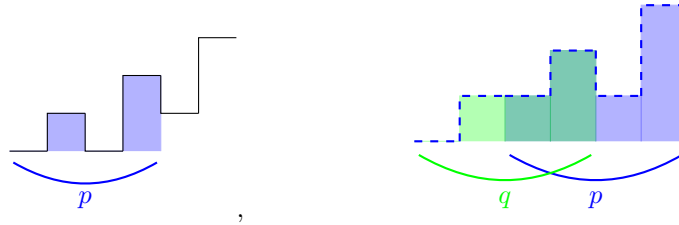


Figure 4: Underdiagonal lattice path diagrams for 010213, 011213.

For a sequence that requires larger recursion depth, the steps can become more complicated, as seen in the next example.

Example 4.2. Suppose $p = 0102$, $q = 0112$. Let $\epsilon = 01022304$.

We currently have p at $\{1\}$; we want q at $\{1\}$.

$\varphi_{=\{1\}}(01022304)$:

| $\varphi_{\geq\{1\}}(01022304) = 01122304$

| q occurs at $\{1, 3\}$, so apply $\varphi_{=\{1,3\}}^{-1}$

| $\varphi_{=\{1,3\}}^{-1}(01122304)$:

| | $\varphi_{\geq\{1,3\}}^{-1}(01122304) = 01020304$

| | p occurs at $\{1, 3, 5\}$, so apply $\varphi_{=\{1,3,5\}}$

| | $\varphi_{=\{1,3,5\}}(01020304)$:

| | | $\varphi_{\geq\{1,3,5\}}(01020304) = 01122334$

| | | q occurs at $\{1, 3, 5\}$, so we are done.

| | 01122334

| | $\varphi_{\geq\{1,3\}}^{-1}(01122334) = 01020334$

| | p occurs at $\{1, 3\}$, so we are done.

| 01020334

| $\varphi_{\geq\{1\}}(01020334) = 01120334$

| q occurs at $\{1, 5\}$, so apply $\varphi_{=\{1,5\}}^{-1}$

| $\varphi_{=\{1,5\}}^{-1}(01120334)$:

| | $\varphi_{\geq\{1,5\}}^{-1}(01120334) = 01020304$

| | p occurs at $\{1, 3, 5\}$, so apply $\varphi_{=\{1,3,5\}}$

| | $\varphi_{=\{1,3,5\}}(01020304)$:

| | | $\varphi_{\geq\{1,3,5\}}(01020304) = 01122334$

| | | q occurs at $\{1, 3, 5\}$, so we are done.

| | 01122334

| | $\varphi_{\geq\{1,5\}}^{-1}(01122334) = 01022324$

| | p occurs at $\{1, 5\}$, so we are done.

| 01022324

| $\varphi_{\geq\{1\}}(01022324) = 01122324$
 | q occurs at $\{1, 3\}$, so apply $\varphi_{=\{1,3\}}^{-1}$
 | $\varphi_{=\{1,3\}}^{-1}(01122324)$:
 | | $\varphi_{\geq\{1,3\}}^{-1}(01122324) = 01020324$
 | | p occurs at $\{1, 3\}$, so we are done.
 | 01020324
 | $\varphi_{\geq\{1\}}(01020324) = 01120324$
 | q occurs at $\{1\}$, so we are done.

01120324

We finally obtain $01022304 \mapsto 01120324$.

The following pictures show the underdiagonal lattice paths of the original sequence 01022304, and the resulting sequences after each application of $\varphi_{\geq\{0\}}$.

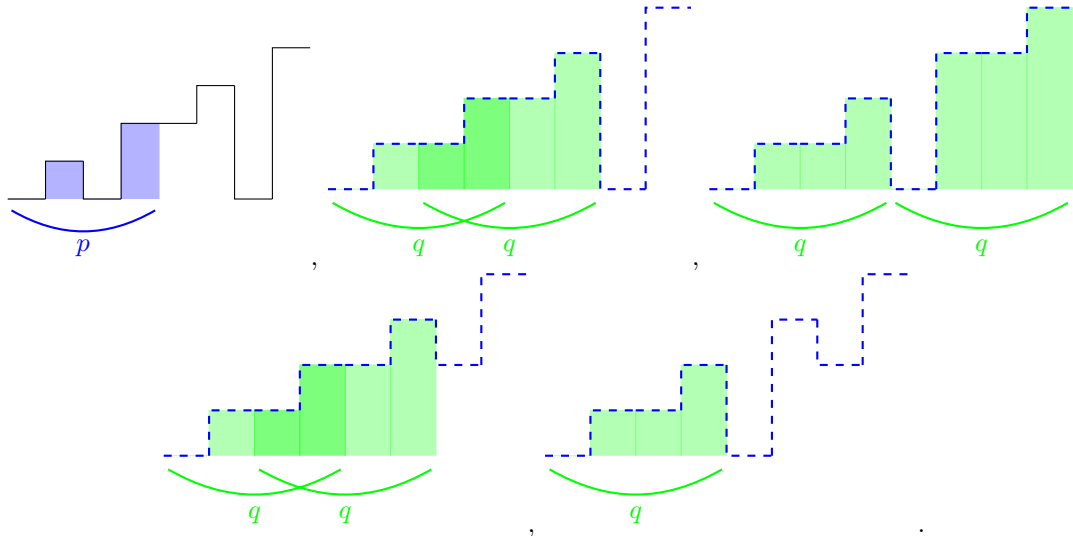


Figure 5: Underdiagonal lattice path diagrams for 01022304, 01122304, 01120334, 01122324, 01120324.

Remark 4.1. The sequences for which the number of required applications of φ_{\geq} and φ_{\geq}^{-1} is maximized are not unique. For example, when $n = 8$, the following two computations both result in 19 function calls: $01120314 \mapsto 01021304$, $01130537 \mapsto 01031517$.

For $01120314 \mapsto 01021304$, we can see what looks like a direct relationship between the input and output. A larger example can be found in Appendix A. This class of examples motivated the development of the algorithm in the following section.

4.1 Reciprocal relations

While the direct description of the previous bijection is not straightforward to deduce, we can actually produce a stronger result. Recall the following definition:

Definition 4.1. Let p and q be consecutive patterns. We say that p and q are reciprocal if for all pairs $S, T \subseteq [n]$,

$$|\{\epsilon \in I_n : \text{Em}(p, \epsilon) = S, \text{Em}(q, \epsilon) = T\}| = |\{\epsilon \in I_n : \text{Em}(p, \epsilon) = T, \text{Em}(q, \epsilon) = S\}|.$$

In other words, we can swap the occurrences of p and q in an inversion sequence. When p and q satisfy this relation, we denote this by $p \stackrel{R}{\sim} q$.

It turns out that the consecutive patterns 0102 and 0112 are reciprocal as well. Algorithm 1 takes a sequence seq and switches all instances of $p = \underline{0102}$ and $q = \underline{0112}$ in seq .

After collecting several upcoming results about the algorithm, we see that applying the algorithm twice can be visualized by the diagrams shown in Figure 6.

Algorithm 1 Bijection to show that $p = 0102$ and $q = 0112$ are reciprocal

Require: sequence seq with occurrences of p and q at positions S and T , respectively.

Ensure: sequence with occurrences of q and p at positions S and T , respectively.

$\text{Em}(p) \leftarrow$ indices of occurrences of p in seq

$\text{Em}(q) \leftarrow$ indices of occurrences of q in seq

$\text{last} \leftarrow \text{null}$

for each position i in seq **do**

$\text{newEm}(p) \leftarrow$ current indices of occurrences of p in seq

$\text{newEm}(q) \leftarrow$ current indices of occurrences of q in seq

if $i - 2 \in \text{Em}(p)$ **then**

\triangleright The i th entry is the third position of an original occurrence of p in seq .

$\text{last} \leftarrow \text{seq}[i]$

\triangleright Store whatever is currently in position i into last .

$\text{seq}[i] \leftarrow \text{seq}[i - 1]$

\triangleright Copy the entry directly to the left into position i

\triangleright (this changes the original p to a q).

else if $i - 2 \in \text{Em}(q)$ **then**

\triangleright The i th entry is the third position of an original occurrence of q in seq .

$\text{last} \leftarrow \text{seq}[i]$

\triangleright Store whatever is currently in position i into last .

$\text{seq}[i] \leftarrow \text{seq}[i - 2]$

\triangleright Copy whatever digit is two to the left into position i

\triangleright (this changes the original q to a p).

\triangleright Now if the i th entry of seq is the third position of a new (unwanted) occurrence of p or q in seq ...

else if $i - 2 \in \text{newEm}(p)$ **then**

$\text{temp} \leftarrow \text{seq}[i]$

$\text{seq}[i] \leftarrow \text{last}$

$\text{last} \leftarrow \text{temp}$

\triangleright Copy whatever is in last into position i , and store the old digit in last .

else if $i - 2 \in \text{newEm}(q)$ **then**

$\text{temp} \leftarrow \text{seq}[i]$

$\text{seq}[i] \leftarrow \text{last}$

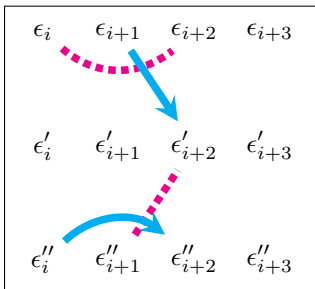
$\text{last} \leftarrow \text{temp}$

end if

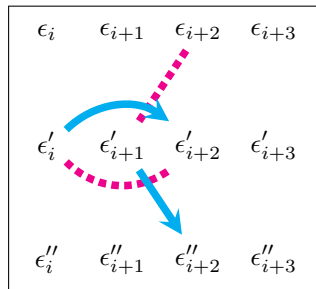
end for

return seq

if ϵ has an occurrence of p at i :



if ϵ has an occurrence of q at i :



if an “extra” occurrence of p appears at i :

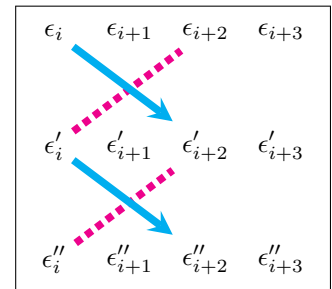


Figure 6: Diagrams of replacements made by Algorithm 1 in each case of an occurrence at i . The dotted edges in the diagrams show equalities between elements that cause a replacement to be made (additional required inequalities are not indicated). The solid edges show how entries in the sequence are copied to other positions in the sequence. So each edge denotes an equality of sequence entries.

Example 4.3.

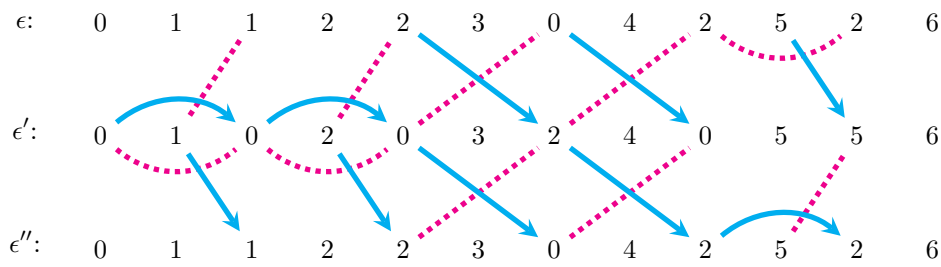


Figure 7: Diagram of replacements made when applying Algorithm 1 twice to the sequence $\epsilon = 011223042526$.

The following results will show that this algorithm gives a bijection between

$$\{\epsilon \in I_n : \text{Em}(p, \epsilon) = S \text{ and } \text{Em}(q, \epsilon) = T\} \text{ and } \{\epsilon \in I_n : \text{Em}(q, \epsilon) = S \text{ and } \text{Em}(p, \epsilon) = T\}$$

for all $S, T \subseteq [n]$. First, we show that given an inversion sequence as the input, the algorithm will always output an inversion sequence.

Proposition 4.1. *Let $\epsilon = \epsilon_1\epsilon_2 \cdots \epsilon_n$ be an inversion sequence, and let ϵ' be the resulting sequence after applying Algorithm 1 to ϵ . Then ϵ' is also an inversion sequence.*

Proof. Since ϵ is an inversion sequence, we know that $0 \leq \epsilon_i < i$ for all i . If Algorithm 1 replaces any entries of ϵ , the entries will be replaced with entries from earlier in the sequence. That is, if ϵ_j is replaced, then $\epsilon'_j = \epsilon_i$ for some $i < j$. Then

$$0 \leq \epsilon'_j = \epsilon_i < i < j,$$

so $0 \leq \epsilon'_j < j$ for all j . Also, if ϵ_j is not replaced, we have $0 \leq \epsilon'_j = \epsilon_j < j$. Thus ϵ' is also an inversion sequence. \square

Lemma 4.1. *Let ϵ be an initial sequence to which the algorithm is applied. Suppose i is an occurrence of p or q in the sequence at any step in the algorithm. Then among the set of entries $\{\epsilon_i, \epsilon_{i+1}, \epsilon_{i+2}, \epsilon_{i+3}\}$, only ϵ_i or ϵ_{i+2} could have been changed from the original sequence.*

Proof. The algorithm only makes replacements at the third position of either occurrences of p or q that appeared in the original sequence (even if they are not currently occurrences when the algorithm reaches that position) or occurrences of p or q that are created as entries are replaced.

The consecutive patterns 0102 and 0112 can only overlap themselves or each other in one or two entries. In particular, the third entry of one pattern occurrence could either not be part of another pattern occurrence, or be the first entry of another pattern occurrence. Since these are the only entries in the sequence that are changed, and these are the only positions where they can appear in a pattern occurrence, the result follows. \square

We will refer to the occurrences of p or q that appeared in the original sequence as *original* p or q . A position satisfying this is considered an original p or q even if it is not an occurrence of p or q in the sequence at some point during the algorithm. We say an occurrence is a *transient* p or q if it is a non-original occurrence of p or q that appears during the algorithm as a result of replacements.

That is, if ϵ is the initial sequence and ϵ' is the sequence resulting from applying the algorithm to ϵ , we say there is

- an *original* occurrence of p at i if $\epsilon_i = \epsilon_{i+2}$, and $\epsilon_i < \epsilon_{i+1} < \epsilon_{i+3}$,
- an *original* occurrence of q at i if $\epsilon_i < \epsilon_{i+1} = \epsilon_{i+2} < \epsilon_{i+3}$,
- a *transient* occurrence of p at i if i is not an original occurrence of p , but $\epsilon'_i = \epsilon_{i+2}$, and $\epsilon'_i < \epsilon_{i+1} < \epsilon_{i+3}$, and
- a *transient* occurrence of q at i if i is not an original occurrence of q , but $\epsilon'_i < \epsilon_{i+1} = \epsilon_{i+2} < \epsilon_{i+3}$.

Note: It follows from these definitions that if there is a transient p or q in position i , then $\epsilon'_i \neq \epsilon_i$. We will also show that the non-original occurrences of p or q are indeed “transient” as these occurrences are removed later in the algorithm.

Lemma 4.2. *Let ϵ be an inversion sequence to which the algorithm is applied, and let ϵ' be the resulting sequence. If an entry ϵ_i in the sequence is replaced during the algorithm,*

(i) $\epsilon_i < \epsilon_{i+1}$, and

(ii) the replacement entry ϵ'_i is less than ϵ_{i+1} .

Proof. (i): The third entry in an occurrence of either pattern will be less than the fourth entry. The algorithm only makes replacements at the third position of either original occurrences of p or q , or transient occurrences of p or q . In the case of an original occurrence of p or q at index $i - 2$, we have $\epsilon_i < \epsilon_{i+1}$. In the case of a transient occurrence of p or q at index $i - 2$, we have $\epsilon_i < \epsilon_{i+1}$.

(ii): The first time that an entry in the sequence will be replaced is in the first original occurrence of p or q . Suppose the first such occurrence is at position j . Since no replacements have been made so far, we know the original p or q is currently unchanged. So $\epsilon_j < \epsilon_{j+3}$, and $\epsilon_{j+1} < \epsilon_{j+3}$. The entry in position j or $j + 1$ will replace the entry in position $j + 2$. So the result holds for the position of the first (leftmost) replacement in the algorithm.

We will proceed by induction. Suppose the result holds for all k such that $0 \leq k < i$.

The algorithm will only replace an entry if it is in the third position of either an original occurrence of p or q , or a transient occurrence of p or q in the current sequence.

Suppose i is the third position of an original p or q . Either the original p or q in that position is completely unchanged, or the first digit was changed, by Lemma 4.1.

If the original p or q is completely unchanged, then we have that ϵ_{i-2} and ϵ_{i-1} are both less than ϵ_{i+1} , and the value of one of them will replace ϵ_i . In either case, the replaced digit will still be less than ϵ_{i+1} .

If the first entry of the original p or q was changed, then by the inductive hypothesis, we know that $\epsilon'_{i-2} < \epsilon_{i-1}$. Also, at the point in the algorithm when the first entry ϵ_{i-2} was replaced, the second, third, and fourth entries of the occurrence were all still unchanged. So the second entry is still less than the fourth, and the (replaced) first entry is less than the second, so the first entry is also less than the fourth. Then when the third digit is changed by replacing it with the first or second entry, the replacement is still less than the entry directly to its right.

Otherwise, suppose that i is the third position of a transient occurrence of p or q . In this case, whatever is held in the variable `last` will replace the entry in position i .

Since $i - 2$ is a transient occurrence of p or q , we know by Lemma 4.1 that this occurrence must have resulted from a replacement in position $i - 2$.

Then ϵ_{i-2} is now stored in `last`. By Lemma 4.2(i), since ϵ_{i-2} was replaced, we have that $\epsilon_{i-2} < \epsilon_{i-1}$. That is, the entry stored in `last` is less than ϵ_{i-1} .

Since $i - 2$ is a transient occurrence of p or q , it follows that $\epsilon_{i-1} < \epsilon_{i+1}$. So, the entry stored in `last` is less than ϵ_{i+1} . Since ϵ_i is replaced with whatever entry is in `last`, the result holds in this case as well. \square

Corollary 4.1. *In Algorithm 1, it is not possible to generate a transient q .*

Proof. Let ϵ be the initial sequence, and let ϵ' be the sequence resulting from applying the algorithm. Suppose to the contrary that a transient occurrence of q appears at position i , so i is not an original occurrence of q , but $\epsilon'_i < \epsilon_{i+1} = \epsilon_{i+2} < \epsilon_{i+3}$. Since i is not an original occurrence of q , we know that the following must not all be true:

$$\epsilon_i < \epsilon_{i+1} = \epsilon_{i+2} < \epsilon_{i+3}.$$

It follows that the inequality that did not hold is $\epsilon_i < \epsilon_{i+1}$. This implies $\epsilon'_i \neq \epsilon_i$, so there must have been a replacement at position i . However, since the inequality $\epsilon_i < \epsilon_{i+1}$ does not hold, this contradicts Lemma 4.2. \square

Remark 4.2. *In light of the previous corollary, we can shorten the pseudocode for the algorithm to remove the conditional case for the transient q since it will never be used.*

The following result will also be useful.

Corollary 4.2. *Let $\epsilon = \epsilon_1\epsilon_2 \cdots \epsilon_n$ be the reversal of an inversion sequence, and let ϵ' be the resulting sequence after applying Algorithm 1 to ϵ . Then ϵ' is also the reversal of an inversion sequence.*

Proof. A sequence $\epsilon = \epsilon_1\epsilon_2 \cdots \epsilon_n$ is the reversal of an inversion sequence if and only if $0 \leq \epsilon_i \leq n - i$ for all i . As in the proof of Proposition 4.1, if ϵ_j is replaced, then $\epsilon'_j = \epsilon_i$ for some $i < j$. By Lemma 4.2(ii), we know $\epsilon'_j < \epsilon_{j+1}$. Then,

$$0 \leq \epsilon_i = \epsilon'_j < \epsilon_{j+1} \leq n - j - 1,$$

so $0 \leq \epsilon'_j < n - j - 1 < n - j$. Also, if ϵ_j is not replaced, we have $0 \leq \epsilon'_j = \epsilon_j \leq n - j$. Thus ϵ' is also the reversal of an inversion sequence. \square

Lemma 4.3. *Let ϵ be an inversion sequence to which the algorithm is applied, and let ϵ' be the resulting sequence. If i is a transient occurrence of p , then ϵ'_{i+2} is not equal to ϵ'_i .*

Proof. Since i is a transient occurrence of p , we know by Lemma 4.1 that this occurrence must have resulted from a replacement in position i , meaning that i was the third position in a transient p or original p or q . (If ϵ_i had not been changed, then i is an original occurrence of p or q .) Also by Lemma 4.1, this must have been the most recent replacement made when the algorithm reaches position $i + 2$, so ϵ_i is the entry stored in **last**.

Since i is a transient occurrence of p or q , we know $\epsilon'_i \neq \epsilon_i$, and the entry in position $i + 2$ will be replaced with the entry in **last**, so $\epsilon'_{i+2} = \epsilon_i$. The result follows. \square

Theorem 4.1. *Algorithm 1 exchanges occurrences of p and q in a sequence $\epsilon_1 \dots \epsilon_n$.*

Proof. Let ϵ' be the sequence that results after applying the algorithm to a sequence ϵ . We will show that $\text{Em}(p, \epsilon) = \text{Em}(q, \epsilon')$ and $\text{Em}(q, \epsilon) = \text{Em}(p, \epsilon')$.

Suppose there is an original p or q in ϵ at position i . If the first two entries in this occurrence were unchanged when the third digit is reached, then by construction, the algorithm will convert the pattern to q or p , respectively.

By Lemma 4.1, the only other possibility is that the first digit was changed by the algorithm before the third digit was reached. By Lemma 4.2, even if the first digit was changed, it is still less than the second digit. So if i is the position of an original p , and the entry in position i was changed, we still have $\epsilon'_i < \epsilon_{i+1}$ ($= \epsilon'_{i+1}$) and $\epsilon_{i+1} < \epsilon_{i+3}$ ($= \epsilon'_{i+3}$). Then replacing ϵ_{i+2} with ϵ_{i+1} yields an occurrence of q in position i , as the resulting sequence will have $\epsilon'_i < \epsilon'_{i+1} = \epsilon_{i+2} < \epsilon'_{i+3}$. Similarly, if i is the position of an original q , copying the entry in ϵ_i to position $i + 2$ will yield an occurrence of p in position i .

So we have that $\text{Em}(p, \epsilon) \subseteq \text{Em}(q, \epsilon')$, and $\text{Em}(q, \epsilon) \subseteq \text{Em}(p, \epsilon')$.

It remains to show that there will not be any occurrences of p or q in ϵ' that were not in $\text{Em}(p, \epsilon)$ or $\text{Em}(q, \epsilon)$. By Lemma 4.1, we have that “extra” occurrences of q cannot be generated by the algorithm.

If a new (transient) occurrence of p is created at position i in the sequence by the algorithm, this will be corrected by replacing the third entry in this occurrence with the entry stored in **last**. This will change an occurrence of p to a non-occurrence, as long as ϵ'_{i+2} is not equal to ϵ'_i . This is guaranteed by Lemma 4.3.

So the resulting sequence ϵ' will not have any occurrences of p that were not in $\text{Em}(q, \epsilon)$, and will not have any occurrences of q that were not in $\text{Em}(p, \epsilon)$. Thus the result holds. \square

Lemma 4.4. *The entry $\epsilon'_i \neq \epsilon_i \iff i - 2$ is an original p , transient p , or original q .*

Proof. (\Leftarrow): If $\epsilon'_i \neq \epsilon_i$ then it must be the case that i was the third position of a pattern occurrence at some point in the sequence. (this is the only scenario where an entry could be changed).

(\Rightarrow): If $i - 2$ is an original p , we have $\epsilon'_{i-1} = \epsilon_{i-1}$ (by Lemma 4.1), $\epsilon_{i-1} > \epsilon_i$ and $\epsilon'_i = \epsilon_{i-1}$. So $\epsilon'_i \neq \epsilon_i$.

If $i - 2$ is an original q , we have $\epsilon_{i-2} < \epsilon_{i-1} = \epsilon_i$ and $\epsilon'_i = \epsilon'_{i-2}$. If $\epsilon'_{i-2} = \epsilon_{i-2}$ then $\epsilon'_i \neq \epsilon_i$.

If $\epsilon'_{i-2} \neq \epsilon_{i-2}$ then $\epsilon'_{i-2} < \epsilon_{i-1}$ still by Lemma 4.2. So $\epsilon'_i = \epsilon'_{i-2} < \epsilon_{i-1} = \epsilon_i$; thus $\epsilon'_i \neq \epsilon_i$.

If $i - 2$ is a transient p , we have $\epsilon_{i-2} \neq \epsilon'_{i-2} = \epsilon_i$, and $\epsilon'_i = \epsilon_{i-2}$. So $\epsilon'_i \neq \epsilon_i$. \square

Theorem 4.2. *The mapping described by Algorithm 1 is an involution.*

Proof. Let ϵ be an inversion sequence, let ϵ' be the resulting sequence after applying Algorithm 1 to ϵ , and let ϵ'' be the resulting sequence after applying the algorithm to ϵ' . We will show by induction that $\epsilon''_i = \epsilon_i$ for all i , so $\epsilon'' = \epsilon$.

Base case: The first entry will never be changed by the algorithm, because it is not the third entry of any pattern occurrence. So $\epsilon_1 = \epsilon'_1 = \epsilon''_1$.

Now assume $\epsilon''_k = \epsilon_k$ for all $k < i$. We will show $\epsilon''_i = \epsilon_i$.

(a) First suppose that $\epsilon'_i \neq \epsilon_i$.

By Lemma 4.4, Algorithm 1 will only replace ϵ_i if one of the following hold:

- (i) ϵ_i is the third entry of an original p . Then $\epsilon_i = \epsilon_{i-2}$. Also by Theorem 4.1, after applying the algorithm twice we know that $i - 2$ will be an original p in ϵ'' , so $\epsilon''_i = \epsilon''_{i-2}$. By the inductive hypothesis, $\epsilon''_{i-2} = \epsilon_{i-2}$. So we have $\epsilon_i = \epsilon_{i-2} = \epsilon''_{i-2} = \epsilon''_i$.
- (ii) ϵ_i is the third entry of an original q . In this case, $\epsilon_i = \epsilon_{i-1}$. Also by Theorem 4.1, after applying the algorithm twice we know $\epsilon''_i = \epsilon''_{i-1}$. By the inductive hypothesis, $\epsilon''_{i-1} = \epsilon_{i-1}$, so $\epsilon_i = \epsilon_{i-1} = \epsilon''_{i-1} = \epsilon''_i$.
- (iii) ϵ_i is the third entry of a transient p . Since i is third entry of transient p , we have $\epsilon'_{i-2} = \epsilon_i < \epsilon_{i-1} < \epsilon_{i+1}$. and $\epsilon_{i-2} = \epsilon'_i$. By the inductive hypothesis, $\epsilon''_{i-2} = \epsilon_{i-2}$, so $\epsilon''_{i-2} = \epsilon'_i$. Moreover, $\epsilon'_{i-1} = \epsilon_{i-1}$ and $\epsilon'_{i+1} = \epsilon_{i+1}$ by Lemma 4.1. So we have

$$\epsilon''_{i-2} = \epsilon'_i = \epsilon_{i-2} < \epsilon_{i-1} = \epsilon'_{i-1} < \epsilon'_{i+1} = \epsilon_{i+1}$$

Thus $i - 2$ is a transient p in ϵ' . So $\epsilon''_i = \epsilon'_{i-2}$; thus $\epsilon_i = \epsilon'_{i-2} = \epsilon''_i$.

So if $\epsilon'_i \neq \epsilon_i$, we have $\epsilon''_i = \epsilon_i$.

(b) Otherwise suppose $\epsilon'_i = \epsilon_i$. By Lemma 4.4, i is not the third entry of a pattern occurrence that appears during the algorithm. We will show that this is also true in ϵ' , so $\epsilon'_i = \epsilon''_i$.

(i) We know by Theorem 4.1 that $i - 2$ is an original p in ϵ if and only if $i - 2$ is an original q in ϵ' and $i - 2$ is an original q in ϵ if and only if $i - 2$ is an original p in ϵ' . It follows that $i - 2$ is neither an original p or q in ϵ' , so in these cases, $\epsilon'_i = \epsilon''_i$.

(ii) The only other possibility is that $i - 2$ is a transient p in ϵ' . Then $\epsilon''_{i-2} = \epsilon'_i < \epsilon'_{i-1} < \epsilon'_{i+1}$.

We know

$$\begin{aligned} \epsilon''_{i-2} &= \epsilon_{i-2}, && \text{(by the inductive hypothesis)} \\ \epsilon''_{i-1} &= \epsilon_{i-1}, && \text{(by the inductive hypothesis)} \\ \epsilon'_i &= \epsilon_i. && \text{(by assumption)} \end{aligned}$$

so by substitution we have $\epsilon_{i-2} = \epsilon_i < \epsilon_{i-1} < \epsilon'_{i+1}$.

i. Now, if $\epsilon_{i+1} = \epsilon'_{i+1}$, we have $\epsilon_{i-2} = \epsilon_i < \epsilon_{i-1} < \epsilon_{i+1}$ which implies that there was an original p in ϵ at $i - 2$, which is a contradiction.

ii. Otherwise, $\epsilon_{i+1} \neq \epsilon'_{i+1}$, which by Lemma 4.4 implies that $i - 1$ is an original p , original q , or transient p in ϵ . Then either $\epsilon'_{i+1} = \epsilon'_i$ (if $i - 1$ is an original p), ϵ'_{i-1} (if $i - 1$ is an original q), or ϵ_{i-1} (if $i - 1$ is a transient p).

So ϵ'_{i+1} is equal to either $\epsilon'_i = \epsilon_i$ or $\epsilon'_{i-1} = \epsilon_{i-1}$. However, we already had that $\epsilon_i < \epsilon_{i-1} < \epsilon'_{i+1}$, so this is a contradiction.

So $i - 2$ is not a transient p in ϵ' ; thus $\epsilon''_i = \epsilon'_i$.

It follows that whenever $\epsilon'_i = \epsilon_i$, we have $\epsilon''_i = \epsilon'_i$. So again we have $\epsilon''_i = \epsilon_i$.

In all possible cases, $\epsilon''_i = \epsilon_i$, so the result holds by induction. □

Corollary 4.3. *Let ϵ be an inversion sequence and let ϵ' be the sequence resulting from applying Algorithm 1 to ϵ . If there is a transient p at i in ϵ , then there is a transient p at i in ϵ' .*

Proof. This is show in case (iii) of part (a) of the proof of Theorem 4.2, relying on the hypothesis that $\epsilon''_k = \epsilon_k$ for all $k < i$, which we now know is true for all k . □

Remark 4.3. *This is not the same bijection as the one found recursively. For example, we see in Example 6 that the previous bijection maps 010223 to 011203. Algorithm 1 would map 010223 to 011213.*

5. The remaining equivalences

The remaining set of equivalences,

$$\underline{2010} \sim \underline{2110} \sim \underline{2120},$$

can be shown similarly. Note that $\underline{2010}$ is the reversal of the pattern 0102 , and $\underline{2110}$ is the reversal of the pattern 0112 . So if a sequence ϵ of length n has an occurrence of $\underline{2010}$ at position i , its reversal will have an occurrence of 0102 at position $n - i - 2$.

So to show the equivalence $\underline{2010} \sim \underline{2110}$, we can modify the bijection by first reversing the input sequence, applying Algorithm 1 to this, and then reversing the output sequence. By Corollary 4.2, we know that this process will send inversion sequences to inversion sequences. So, this will give a bijection between

$$\{\epsilon \in I_n : \text{Em}(\underline{2010}, \epsilon) = S \text{ and } \text{Em}(\underline{2110}, \epsilon) = T\}$$

and

$$\{\epsilon \in I_n : \text{Em}(\underline{2110}, \epsilon) = S \text{ and } \text{Em}(\underline{2010}, \epsilon) = T\}$$

for all $S, T \subseteq [n]$.

Also note that converting between $\underline{2120}$ and $\underline{2110}$ requires the third entry to be replaced by the value in either the first or second entry, similarly to the case for switching between $\underline{0102}$ and $\underline{0112}$. Indeed, reassigning $p = \underline{2120}$ and $q = \underline{2110}$, Algorithm 1 as described previously yields a bijection between $\{\epsilon \in I_n : \text{Em}(\underline{2120}, \epsilon) = S \text{ and } \text{Em}(\underline{2110}, \epsilon) = T\}$ and $\{\epsilon \in I_n : \text{Em}(\underline{2110}, \epsilon) = S \text{ and } \text{Em}(\underline{2120}, \epsilon) = T\}$ for all $(S, T) \subseteq [n]$. In order to verify this, we will need to make slight modifications in the proofs of the previous lemmas. First, Lemma 4.1 still holds since $\underline{2120}$ and $\underline{2110}$ can only overlap themselves or each other in one or two entries.

For $p = \underline{2120}$ and $q = \underline{2110}$, we define original and transient pattern occurrences analogously as before:

If ϵ is the initial sequence and ϵ' is the sequence resulting from applying the algorithm to ϵ , we say there is

- an *original* occurrence of p at i if $\epsilon_i = \epsilon_{i+2} > \epsilon_{i+1} > \epsilon_{i+3}$,
- an *original* occurrence of q at i if $\epsilon_i > \epsilon_{i+1} = \epsilon_{i+2} > \epsilon_{i+3}$,
- a *transient* occurrence of p at i if i is not an original occurrence of p , but $\epsilon'_i = \epsilon_{i+2} > \epsilon_{i+1} > \epsilon_{i+3}$, and
- a *transient* occurrence of q at i if i is not an original occurrence of q , but $\epsilon'_i > \epsilon_{i+1} = \epsilon_{i+2} > \epsilon_{i+3}$.

Note: These are the same definitions we gave before, with all inequalities reversed. The results shown for the case when $p = \underline{0102}$ and $q = \underline{0112}$ have analogs for $p = \underline{2120}$ and $q = \underline{2110}$, obtained by reversing the inequalities. So, Algorithm 1 still works to show that $\underline{2120} \stackrel{R}{\sim} \underline{2110}$.

Lastly, we note that collecting all of these results, we have reciprocal relations among almost all pairs given in Theorem 2.1, with the exceptions of $\underline{2010} \sim \underline{2120}$, $\underline{1000} \sim \underline{1110}$, and $\underline{2100} \sim \underline{2210}$. These cannot be reciprocal due to asymmetry in how the patterns can overlap:

For $\underline{2120}$ and $\underline{2010}$, the sequence 00212010 has $\underline{2120}$ at position 3 and $\underline{2010}$ at position 5. However, there is no inversion sequence ϵ with $\underline{2010}$ at position 3 and $\underline{2120}$ at position 5. Having $\underline{2120}$ at position 5 would require $\epsilon_5 > \epsilon_6 > \epsilon_8$, and also having $\underline{2010}$ at position 3 would require $\epsilon_3 > \epsilon_5$, so combined, we have $\epsilon_3 > \epsilon_5 > \epsilon_6 > \epsilon_8$. Since the entries in inversion sequences are all nonnegative integers, it follows that $\epsilon_3 \geq 3$. However, if ϵ is an inversion sequence, we must have $\epsilon_3 < 3$, so this is not possible.

The sequence 21110 has $\underline{1000}$ at position 1 and $\underline{1110}$ at position 2. However, we cannot have a sequence with $\underline{1000}$ at position 2 and $\underline{1110}$ at position 1, since the $\underline{1110}$ at position 1 would require that the second and third entries of the sequence are equal, but having $\underline{1000}$ at position 2 would require that the second and third entries of the sequence are not equal.

Similarly, $\underline{22100}$ has $\underline{2100}$ at position 2 and $\underline{2210}$ at position 1. However, we cannot create a sequence with $\underline{2100}$ at position 1 and $\underline{2210}$ at position 2 — the first condition would require the second and third entries to be different, but the second condition requires them to be equal.

A. Extended example

Example A.1. $\epsilon = 011203140516$

We currently have p at $\{\}$; we want q at $\{\}$.

```

 $\varphi_{=\{\}}(011203140516)$ :
|  $\varphi_{\geq\{\}}(011203140516) = 011203140516$ 
|  $q$  occurs at  $\{1\}$ , so apply  $\varphi_{=\{1\}}^{-1}$ 
|  $\varphi_{=\{1\}}^{-1}(011203140516)$ :
| |  $\varphi_{\geq\{1\}}^{-1}(011203140516) = 010203140516$ 
| |  $p$  occurs at  $\{1, 3\}$ , so apply  $\varphi_{=\{1,3\}}$ 
| |  $\varphi_{=\{1,3\}}(010203140516)$ :
| | |  $\varphi_{\geq\{1,3\}}(010203140516) = 011223140516$ 
| | |  $q$  occurs at  $\{1, 3\}$ , so we are done.
| | 011223140516
| |  $\varphi_{\geq\{1\}}^{-1}(011223140516) = 010223140516$ 
| |  $p$  occurs at  $\{1\}$ , so we are done.
| 010223140516
|  $\varphi_{\geq\{\}}(010223140516) = 010223140516$ 
|  $q$  occurs at  $\{3\}$ , so apply  $\varphi_{=\{3\}}^{-1}$ 
|  $\varphi_{=\{3\}}^{-1}(010223140516)$ :
| |  $\varphi_{\geq\{3\}}^{-1}(010223140516) = 010203140516$ 
| |  $p$  occurs at  $\{1, 3\}$ , so apply  $\varphi_{=\{1,3\}}$ 
| |  $\varphi_{=\{1,3\}}(010203140516)$ :
| | |  $\varphi_{\geq\{1,3\}}(010203140516) = 011223140516$ 
| | |  $q$  occurs at  $\{1, 3\}$ , so we are done.
| | 011223140516
| |  $\varphi_{\geq\{3\}}^{-1}(011223140516) = 011213140516$ 
| |  $p$  occurs at  $\{3, 5\}$ , so apply  $\varphi_{=\{3,5\}}$ 
| |  $\varphi_{=\{3,5\}}(011213140516)$ :
| | |  $\varphi_{\geq\{3,5\}}(011213140516) = 011223340516$ 
| | |  $q$  occurs at  $\{1, 3, 5\}$ , so apply  $\varphi_{=\{1,3,5\}}^{-1}$ 
| | |  $\varphi_{=\{1,3,5\}}^{-1}(011223340516)$ :

```

$\varphi_{\geq\{1,3,5\}}^{-1}(011223340516) = 010203040516$
p occurs at $\{1, 3, 5, 7\}$, so apply $\varphi_{=\{1,3,5,7\}}$
 $\varphi_{=\{1,3,5,7\}}(010203040516)$:
 $\varphi_{\geq\{1,3,5,7\}}(010203040516) = 011223344516$
q occurs at $\{1, 3, 5, 7\}$, so we are done.
011223344516
 $\varphi_{\geq\{1,3,5\}}^{-1}(011223344516) = 010203044516$
p occurs at $\{1, 3, 5\}$, so we are done.
010203044516
 $\varphi_{\geq\{3,5\}}(010203044516) = 010223344516$
q occurs at $\{3, 5, 7\}$, so apply $\varphi_{=\{3,5,7\}}^{-1}$
 $\varphi_{=\{3,5,7\}}^{-1}(010223344516)$:
 $\varphi_{\geq\{3,5,7\}}^{-1}(010223344516) = 010203040516$
p occurs at $\{1, 3, 5, 7\}$, so apply $\varphi_{=\{1,3,5,7\}}$
 $\varphi_{=\{1,3,5,7\}}(010203040516)$:
 $\varphi_{\geq\{1,3,5,7\}}(010203040516) = 011223344516$
q occurs at $\{1, 3, 5, 7\}$, so we are done.
011223344516
 $\varphi_{\geq\{3,5,7\}}^{-1}(011223344516) = 011213141516$
p occurs at $\{3, 5, 7, 9\}$, so apply $\varphi_{=\{3,5,7,9\}}$
 $\varphi_{=\{3,5,7,9\}}(011213141516)$:
 $\varphi_{\geq\{3,5,7,9\}}(011213141516) = 011223344556$
q occurs at $\{1, 3, 5, 7, 9\}$, so apply $\varphi_{=\{1,3,5,7,9\}}^{-1}$
 $\varphi_{=\{1,3,5,7,9\}}^{-1}(011223344556)$:
 $\varphi_{\geq\{1,3,5,7,9\}}^{-1}(011223344556) = 010203040506$
p occurs at $\{1, 3, 5, 7, 9\}$, so we are done.
010203040506
 $\varphi_{\geq\{3,5,7,9\}}(010203040506) = 010223344556$
q occurs at $\{3, 5, 7, 9\}$, so we are done.
010223344556
 $\varphi_{\geq\{3,5,7\}}^{-1}(010223344556) = 010203040556$
p occurs at $\{1, 3, 5, 7\}$, so apply $\varphi_{=\{1,3,5,7\}}$
 $\varphi_{=\{1,3,5,7\}}(010203040556)$:
 $\varphi_{\geq\{1,3,5,7\}}(010203040556) = 011223344556$
q occurs at $\{1, 3, 5, 7, 9\}$, so apply $\varphi_{=\{1,3,5,7,9\}}^{-1}$
 $\varphi_{=\{1,3,5,7,9\}}^{-1}(011223344556)$:
 $\varphi_{\geq\{1,3,5,7,9\}}^{-1}(011223344556) = 010203040506$
p occurs at $\{1, 3, 5, 7, 9\}$, so we are done.
010203040506
 $\varphi_{\geq\{1,3,5,7\}}(010203040506) = 011223344506$
q occurs at $\{1, 3, 5, 7\}$, so we are done.
011223344506
 $\varphi_{\geq\{3,5,7\}}^{-1}(011223344506) = 011213141506$
p occurs at $\{3, 5, 7\}$, so we are done.
011213141506
 $\varphi_{\geq\{3,5\}}(011213141506) = 011223341506$
q occurs at $\{1, 3, 5\}$, so apply $\varphi_{=\{1,3,5\}}^{-1}$
 $\varphi_{=\{1,3,5\}}^{-1}(011223341506)$:
 $\varphi_{\geq\{1,3,5\}}^{-1}(011223341506) = 010203041506$
p occurs at $\{1, 3, 5\}$, so we are done.
010203041506
 $\varphi_{\geq\{3,5\}}(010203041506) = 010223341506$
q occurs at $\{3, 5\}$, so we are done.
010223341506
 $\varphi_{\geq\{3\}}^{-1}(010223341506) = 010203341506$
p occurs at $\{1, 3\}$, so apply $\varphi_{=\{1,3\}}$
 $\varphi_{=\{1,3\}}(010203341506)$:

$\varphi_{\geq\{1,3\}}(010203341506) = 011223341506$
 q occurs at $\{1, 3, 5\}$, so apply $\varphi_{=\{1,3,5\}}^{-1}$
 $\varphi_{=\{1,3,5\}}^{-1}(011223341506)$:
 $\varphi_{\geq\{1,3,5\}}^{-1}(011223341506) = 010203041506$
 p occurs at $\{1, 3, 5\}$, so we are done.
 010203041506
 $\varphi_{\geq\{1,3\}}(010203041506) = 011223041506$
 q occurs at $\{1, 3\}$, so we are done.
 011223041506
 $\varphi_{\geq\{3\}}^{-1}(011223041506) = 011213041506$
 p occurs at $\{3\}$, so we are done.
 011213041506
 $\varphi_{\geq\{1\}}(011213041506) = 011213041506$
 q occurs at $\{1\}$, so apply $\varphi_{=\{1\}}^{-1}$
 $\varphi_{=\{1\}}^{-1}(011213041506)$:
 $\varphi_{\geq\{1\}}^{-1}(011213041506) = 010213041506$
 p occurs at $\{1\}$, so we are done.
 010213041506
 $\varphi_{\geq\{1\}}(010213041506) = 010213041506$
 q occurs at $\{1\}$, so we are done.
 010213041506
 We finally obtain $011203140516 \mapsto 010213041506$.

Acknowledgements

The author would like to thank Peter Doyle for numerous helpful discussions, insights, and suggestions. Many thanks as well to Sergi Elizalde, Peter Winkler, and Yan Zhuang for their valuable feedback regarding this work, and additionally, to the anonymous referee for constructive comments.

References

- [1] J. S. Auli and S. Elizalde, *Consecutive Patterns in Inversion Sequences*, Discrete Math. Theor. Comput. Sci. 21.2 (2019).
- [2] J. S. Auli and S. Elizalde, *Consecutive Patterns in Inversion Sequences II: Avoiding Patterns of Relations*, J. Integer Seq. 22 (2019), 19.7.5.
- [3] S. Corteel et al., *Patterns in Inversion Sequences I*, Discrete Math. Theor. Comput. Sci. 18.2 (2016).
- [4] P. G. Doyle. *Let's reinvent subtraction*, arXiv:2203.13836, 2022.
- [5] M. Ferreri, *On generating bijections for permutations and inversion sequences*, Dartmouth College Ph.D. Dissertations, 2024.
- [6] T. Mansour and M. Shattuck, *Pattern avoidance in inversion sequences*. Pure Math. Appl. 25 (2015), 157–176.