

# A stack and a pop stack in series

REBECCA SMITH\*

*Department of Mathematics  
SUNY Brockport, New York  
U.S.A.*

VINCENT VATTER†

*Department of Mathematics  
University of Florida  
Gainesville, Florida  
U.S.A.  
vatter@ufl.edu*

## Abstract

We study sorting machines consisting of a stack and a pop stack in series, with or without a queue between them. While there are, a priori, four such machines, only two are essentially different: a pop stack followed directly by a stack, and a pop stack followed by a queue and then by a stack. In the former case, we obtain complete answers for the basis and enumeration of the sortable permutations. In the latter case, we present a characterization in terms of divided permutations and several conjectures.

## 1 Introduction

A stack is a last-in first-out sorting device with push and pop operations. In Volume 1 of *The Art of Computer Programming* [17, Section 2.2.1], Knuth showed that the permutation  $\pi$  can be sorted (meaning that by applying push and pop operations to the sequence of entries  $\pi(1), \dots, \pi(n)$  one can output the sequence  $1, \dots, n$ ) if and only if  $\pi$  avoids the permutation 231, i.e., if and only if there do not exist three indices  $1 \leq i_1 < i_2 < i_3 \leq n$  such that  $\pi(i_1), \pi(i_2), \pi(i_3)$  are in the same relative order as 231. Shortly thereafter Tarjan [22], Even and Itai [16], Pratt [20], and Knuth himself in *Volume 3* [18, Section 5.2.4] studied networks with multiple stacks in series or in parallel. The questions typically studied for such networks include:

---

\* First author partially supported by the NSA Young Investigator Grant H98230-08-1-0100.

† Second author partially supported by the NSA Young Investigator Grant H98230-12-1-0207.

- Can the set of sortable permutations be characterized by a finite set of forbidden patterns (e.g.,  $\{231\}$  in the case of a single stack)?
- How many permutations of each length can be sorted?

For  $k \geq 2$  stacks in series or in parallel the answer to the first question is no, due to Murphy [19] and Tarjan [22], respectively. The exact enumeration question appears to be much less tractable, and here only relatively crude bounds are known; see Albert, Atkinson, and Linton [4]. For a general overview of stack sorting, we refer the reader to Bóna [13].

Given how difficult the two stacks in series problem appears to be, numerous researchers have considered weaker machines. Atkinson, Murphy, and Ruškuc [9] considered sorting with two *increasing* stacks in series, i.e., two stacks whose entries must be in increasing order when read from top to bottom<sup>1</sup>. They characterized the permutations this machine can sort with an infinite list of forbidden patterns, and also found the enumeration of these permutations. (Interesting, these permutations are in bijection with the 1342-avoiding permutations previously counted by Bóna [12].) Another weakening, sorting with a stack of depth 2 followed by a standard stack (of infinite depth), was studied by Elder [15]. He characterized the sortable permutations with a finite list of forbidden patterns, but did not enumerate these permutations.

The objects we study, pop stacks, were introduced by Avis and Newborn [11]. A *pop stack* is a handicapped form of a stack in which the only way to output an entry in the stack is to pop the entire stack (in last-in first-out order as usual). Avis and Newborn considered placing pop stacks in series, which by their interpretation means that when the entire set of items currently in the  $i$ th pop stack is popped, they are pushed immediately onto the  $(i + 1)$ st pop stack. They proved that the set of permutations sortable by  $k$  pop stacks in series can be characterized by a finite set of forbidden patterns and provided the enumeration of these permutations for every  $k$ .

There is another way to view sorting with pop stacks in series, where one is allowed to save the output of one pop stack and pass it into the next an entry at a time. Serially linking pop stacks in this manner corresponds to placing a queue between the pop stacks, and is much more powerful than the Avis-Newborn interpretation. Atkinson and Stitt [10], who also gave a simpler derivation of Avis and Newborn's enumerative results using what is now known as the substitution decomposition, found the (rational) generating function for the permutations that can be sorted by a pop stack followed by a queue followed by another pop stack.

Here we consider sorting with a stack and a pop stack in series. A priori, there are four different ways by which they may be connected:

---

<sup>1</sup>Even without this restriction, the final stack must be increasing if the sorting is to be successful.

- PS: A pop stack followed by a stack, connected in Avis and Newborn’s manner.
- PQS: A pop stack followed by a queue followed by a stack.
- SP: A stack followed by a pop stack, connected in Avis and Newborn’s manner.
- SQP: A stack followed by a queue followed by a pop stack.

Clearly the permutations sortable by PS are a subset of those sortable by PQS. In the next section, we prove that SP and SQP are equivalent, and that PQS is a symmetry of these two machines. Then, in Section 3 we characterize and enumerate the permutations sortable by PS. In Section 4 we consider the class PQS, but are able to establish few concrete results.

It will be helpful to give names to the operations involved in these sorting machines. Given a system of two stacks (of any type) in series, we refer to moving an entry from the input to the first stack as an *input*, moving an entry from the first stack to the second stack as a *transfer*, and moving an entry from the second stack to the output as an *output*. When there is a queue between the two stacks, we use the term *transfer* to describe both moving an entry from the first stack to the queue and moving it from the queue to the second stack.

We conclude the introduction with a bit of terminology which will be useful. A *permutation class* is a downset of permutations under the containment order. Every permutation class can be specified by the set of minimal permutations which are *not* in the class, which we call its *basis*. Finally, for a set  $B$  of permutations, we denote by  $\text{Av}(B)$  the class of permutations which do not contain any element of  $B$ . For example, Knuth’s result says that the stack-sortable permutations are precisely  $\text{Av}(231)$ , i.e., that they have the singleton basis  $\{231\}$ . Given any naturally defined sorting machine, the set of sortable permutations forms a class (a subpermutation of a sortable permutation can be sorted by ignoring the operations corresponding to absent entries)<sup>2</sup>.

Finally, it is frequently helpful to remember that the permutation containment order has eight symmetries which form the dihedral group of the square. These are generated by two symmetries *inverse* and *reverse*, defined, respectively, by

$$\begin{aligned}\pi^{-1}(\pi(i)) &= i, \\ \pi^r(i) &= \pi(n+1-i),\end{aligned}$$

for all  $i$ . We will also make use of the *complement* symmetry, defined by

$$\pi^c(i) = n+1-\pi(i),$$

also for all  $i$ .

---

<sup>2</sup>The exception that proves this rule is West’s notion of 2-stack-sortability [24], which has an unusual defect due to restrictions on how the machine can use its two stacks. For example, this machine can sort 35241, but not its subpermutation 3241.

## 2 The Equivalence of PQS, SP, & SQP

We begin with the easier equivalence, between the machines SP and SQP.

**Proposition 2.1.** *The machines SP and SQP are equivalent.*

*Proof.* More generally, adding a queue after a normal stack never alters the sorting capabilities of a permutation machine. The machine SM (a stack followed by M) can sort the entries of  $\pi$  if and only if M can sort  $\pi(i_1), \dots, \pi(i_n)$  where  $i_1 \cdots i_n$  is a permutation which can be *generated* by a stack starting with the identity permutation as input. Clearly, the same is true for the machine SQM, because the queue cannot alter the order in which entries pass through it, and consequently through M. (While it is not important to this proof, note that a stack can reorder the entries of  $\pi$  as  $\pi(i_1), \dots, \pi(i_n)$  if and only if  $i_1 \cdots i_n$  avoids  $231^{-1} = 312$ .)  $\square$

Now we show that the permutations sortable by PQS and SQP are symmetries of each other. To do so we consider a construction due to Atkinson (see [7], for example). The *two-stack dual* of the permutation  $\pi$  of length  $n$  is defined by

$$\pi^d = ((\pi^r)^{-1})^r,$$

or more concretely by

$$\pi^d(i) = n + 1 - \pi^{-1}(n + 1 - i)$$

for all  $i$ .

**Proposition 2.2.** *The permutation  $\pi$  can be sorted by two stacks in series if and only if the permutation  $\pi^d$  can be sorted by two stacks in series.*

*Proof.* Consider any sequence of inputs, transfers, and outputs which sorts the permutation  $\pi$ . We think of these operations as generating  $12 \cdots n$  from the input  $\pi$ . By performing these operations in reverse, we obtain a procedure to generate  $\pi^r$  from  $n \cdots 21$  — in this new procedure, the last entry popped from the second stack becomes the first entry pushed onto the first stack. By applying  $(\pi^r)^{-1}$  to the input symbols, we obtain a procedure to generate the identity from  $n \cdots 21 \circ (\pi^r)^{-1}$ , i.e., sort the two-stack dual of  $\pi$ .  $\square$

We now consider the analogue of the proof of Proposition 2.2 for the machine PQS. The first step, of viewing the machine working backwards, requires us to introduce a *push stack*, that is a stack which can be pushed to only when empty, but can accept any length of segment pushed to it, and then can output entries one at a time. Denoting a push stack by P', we see from following the proof of Proposition 2.2 that if the machine PQS can sort  $\pi$  then the machine SQP' can generate  $\pi^r$  from  $n \cdots 21$ . By applying  $(\pi^r)^{-1}$  to the input symbols, we see that SQP' can sort  $\pi^d$ . We now argue that SQP' = SQP. In sorting a permutation with SQP', we can assume that all entries of the input are passed through the stack and placed in the queue before the push stack performs any operations. Once the push stack begins operating, it

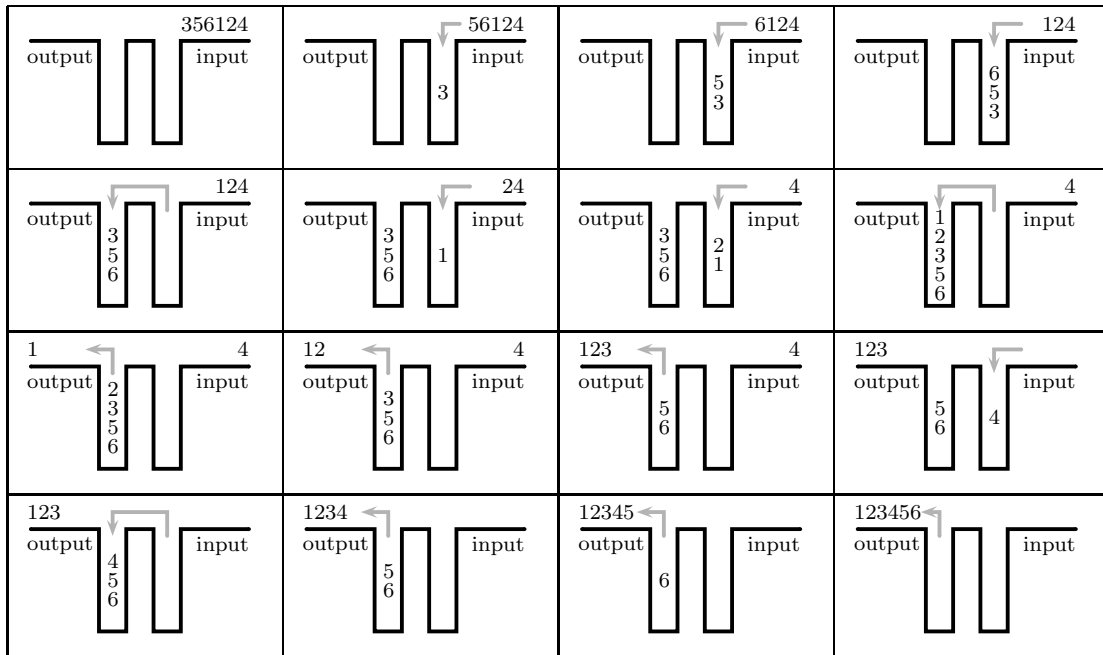


Figure 1: Sorting the permutation 24513 with the PS machine.

will then take initial segments of the entries from the queue, but then must output them all (exactly like a pop stack) before accepting any more input. Therefore, when operated in this manner, the push stack is functionally equivalent to a pop stack. This shows that if  $\pi$  can be sorted by PQS then  $\pi^d$  can be sorted by SQP. The other direction follows in the same manner, establishing the following result.

**Proposition 2.3.** *The permutation  $\pi$  can be sorted by PQS if and only if  $\pi^d$  can be sorted by  $SQP = SP$ .*

### 3 Without a Queue Between — PS

We begin by finding three permutations that the PS machine *cannot* sort. We will later show that these are the only minimal permutations which cannot be sorted by PS.

**Proposition 3.1.** *The permutations 2431, 3142, and 3241 are not PS-sortable.*

*Proof.* The proof consists of three separate case analyses. As the cases are similar, we give the details for the first only. The other two follow from the work of Smith [21] mentioned in the conclusion or can be viewed as exercises for the reader.

Consider attempting to sort the permutation 2431 with PS. First the 2 must be pushed into the pop stack. Suppose the 2 is not transferred to the next stack before the 4 enters the pop stack. If the entries of the pop stack ever contain an increase when read from top to bottom, then the sorting will clearly fail, so the 4 and thus

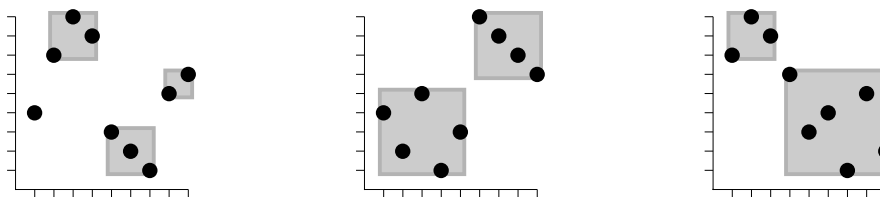


Figure 2: An inflation of 2413, a sum, and a skew sum.

also (by the pop property) the 2 must be transferred to the stack at this point. However, at that point there would be no way to output the 1 before the 3 is forced to be transferred to the stack above the 2.

Alternatively, if the 2 is transferred from the pop stack to the stack before the 4 enters the pop stack, again there is no way to output the 1 before the 3 and 4 are transferred to the stack above the 2.  $\square$

Thus the PS-sortable permutations are a subclass of  $\text{Av}(2431, 3142)$ . The structure of the reverse-complement of this class,  $\text{Av}(3142, 4213)$ , was described by Albert, Atkinson, and Vatter [6]. This structural description rests on the notion of *simple permutations*; the only intervals that are mapped to intervals by such permutations are singletons and their entire domains. For example, 31542 is not simple because it maps  $\{3, 4\}$  to  $\{4, 5\}$ , but 25314 is simple. Simple permutations are precisely those that do not arise from a non-trivial inflation, in the following sense. Given a permutation  $\sigma$  of length  $m$  and nonempty permutations  $\alpha_1, \dots, \alpha_m$ , the *inflation* of  $\sigma$  by  $\alpha_1, \dots, \alpha_m$ , denoted  $\sigma[\alpha_1, \dots, \alpha_m]$ , is the permutation of length  $|\alpha_1| + \dots + |\alpha_m|$  obtained by replacing each entry  $\sigma(i)$  by an interval that is order isomorphic to  $\alpha_i$  in such a way that the intervals themselves are order isomorphic to  $\sigma$ . For example,

$$2413[1, 132, 321, 12] = 4\ 798\ 321\ 56.$$

In particular, the inflation  $12[\alpha_1, \alpha_2]$  is called a (*direct*) *sum* and denoted by  $\alpha_1 \oplus \alpha_2$ . A permutation is *sum decomposable* if it can be expressed as a nontrivial sum, and *sum indecomposable* otherwise. A sum decomposable permutation can always be expressed as  $\alpha_1 \oplus \alpha_2$  where  $\alpha_1$  is sum indecomposable. The inflation  $21[\alpha_1, \alpha_2]$  is similarly called a *skew sum* and denoted  $\alpha_1 \ominus \alpha_2$ ; we define the terms *skew decomposable* and *skew indecomposable* analogously.

Every permutation  $\pi$  is the inflation of a unique simple permutation, called its *simple quotient*. If the quotient has length greater than 2 then the intervals inflating the quotient are uniquely determined by  $\pi$  (as Albert and Atkinson [5]). If the quotient has length 2 (i.e.,  $\pi$  is sum or skew decomposable), then we can enforce uniqueness by insisting that the first interval be sum or skew indecomposable.

A permutation class is *sum closed* if it contains the sum of any two of its members. Clearly the PS-sortable permutations form a sum closed class because if PS can sort  $\pi$  and  $\sigma$  then it can sort  $\pi \oplus \sigma$  (simply sort the interval corresponding to  $\pi$ , then sort the interval corresponding to  $\sigma$ ).

**Proposition 3.2** (Albert, Atkinson, and Vatter [6]). *The simple permutations in  $\text{Av}(2431, 3142)$  are 1 (trivially), 12, 21, and the parallel alternations of the form*

$$246 \cdots (2m)135 \cdots (2m-1)$$

for  $m \geq 2$ .

We aim to produce a structural description of  $\text{Av}(2431, 3142, 3241)$  which will allow us to show that every permutation in the class is PS-sortable. Because each of these basis elements is sum indecomposable, this class is sum closed (as is the class of PS-sortable permutations). For skew sums  $\pi \ominus \sigma$ , notice that to avoid 2431 and 3241,  $\pi$  must avoid 132 and 213, but there are no further restrictions (except that  $\sigma$  must lie in the larger class, obviously). The class  $\text{Av}(132, 213)$  is easily seen to be the *reverse layered permutations*, i.e., those of the form

$$\iota_1 \ominus \cdots \ominus \iota_m$$

where each  $\iota_1, \dots, \iota_m$  is increasing.

It remains to consider inflations of parallel alternations of the form  $246 \cdots (2m)135 \cdots (2m-1)$ . In order to avoid 3241, all even entries in such a parallel alternation except the greatest may only be inflated by increasing intervals. Furthermore, in order to avoid 2431, the greatest even entry also may only be inflated by an increasing interval. Inflations of the odd entries, however, have no further restrictions. We have therefore obtained the following structural decomposition of this class.

**Proposition 3.3.** *The class  $\text{Av}(2431, 3142, 3241)$  consists precisely of the trivial permutation 1 and all permutations of the form*

- (a)  $\pi \oplus \sigma$  where  $\pi, \sigma \in \text{Av}(2431, 3142, 3241)$ ,
- (b)  $\pi \ominus \sigma$  where  $\pi \in \text{Av}(132, 213)$  and  $\sigma \in \text{Av}(2431, 3142, 3241)$ , and
- (c) inflations of parallel alternations  $246 \cdots (2m)135 \cdots (2m-1)$  for  $m \geq 2$  where the even entries are inflated by increasing intervals and the odd entries are inflated by intervals in  $\text{Av}(2431, 3142, 3241)$ .

We now describe how to sort all of the permutations in  $\text{Av}(2431, 3142, 3241)$ , verifying that this is indeed the class of PS-sortable permutations.

**Theorem 3.4.** *The PS-sortable permutations are precisely  $\text{Av}(2431, 3142, 3241)$ .*

*Proof.* Choose an arbitrary  $\pi \in \text{Av}(2431, 3142, 3241)$ . Using induction on the length of  $\pi$ , we show that  $\pi$  is PS-sortable. The base case is trivial, as PS can sort 1. If  $\pi = \sigma \oplus \tau$  for shorter permutations  $\sigma$  and  $\tau$  in this class, then we are done by induction. If  $\pi$  is skew decomposable, then we know from Proposition 3.3 that

$$\pi = \iota_1 \ominus \cdots \ominus \iota_m \ominus \sigma$$

for increasing permutations  $\iota_1, \dots, \iota_m$  and an arbitrary  $\sigma \in \text{Av}(2431, 3142, 3241)$ . To sort permutations of this form, we push each  $\iota_k$ , in turn, onto the pop stack and then pop it into the stack. After having performed this operation on  $\iota_1, \dots, \iota_m$ , we are left with their entries sitting in increasing order in the stack, and thus can (by induction) sort and output the entries of  $\sigma$  and then output the entries of  $\iota_1, \dots, \iota_m$ .

Finally suppose  $\pi$  is of the form (c) in Proposition 3.3, so

$$\pi = 246 \cdots (2m)135 \cdots (2m - 1)[\iota_1, \dots, \iota_m, \sigma_1, \dots, \sigma_m]$$

where  $\iota_1, \dots, \iota_m$  are increasing and  $\sigma_1, \dots, \sigma_m \in \text{Av}(2431, 3142, 3241)$ . To sort these permutations, we first push all entries of  $\iota_1, \dots, \iota_m$  onto the pop stack and then pop them all into the stack. This leaves the entries of  $\iota_1, \dots, \iota_m$  in increasing order on the stack. We then sort and output the entries of  $\sigma_1$  (which can be done by induction), then output the entries of  $\iota_1$ , then sort and output the entries of  $\sigma_2$ , then output the entries of  $\iota_2$ , and so on. □

Proposition 3.3 also leads almost immediately to the enumeration of this class.

**Theorem 3.5.** *The PS-sortable permutations are enumerated by the generating function*

$$\frac{1 - 3x + 2x^2 - \sqrt{1 - 6x + 5x^2}}{2x(2 - x)},$$

sequence A033321 in the OEIS [23].

*Proof.* Let  $f$  denote the generating function for the PS-sortable permutations and  $f_{\oplus}$  (resp.,  $f_{\ominus}$ ) denote the generating function for the sum (resp., skew) decomposable PS-sortable permutations.

The sum indecomposable PS-sortable permutations are therefore counted by  $f - f_{\oplus}$ . Because every sum decomposable permutation can be expressed uniquely as the sum of a sum indecomposable permutation with an arbitrary permutation, we see that  $f_{\oplus} = (f - f_{\oplus})f$ . Solving this for  $f_{\oplus}$  yields

$$f_{\oplus} = \frac{f^2}{1 + f}.$$

Now consider skew decomposable permutations. The reverse layered permutations contain a unique skew indecomposable permutation of each length (the increasing permutation), so by Proposition 3.3, the contribution of skew decomposable permutations is

$$f_{\ominus} = \frac{xf}{1 - x}.$$

Finally, the contribution of permutations of the form (c) in Proposition 3.3 is given by

$$\sum_{m \geq 2} \left( \frac{xf}{1 - x} \right)^m = \frac{(xf)^2}{(1 - x)(1 - x - xf)}.$$



Combining these quantities (and the contribution of the permutation 1) we have

$$f = x + \frac{f^2}{1+f} + \frac{xf}{1-x} + \frac{(xf)^2}{(1-x)(1-x-xf)},$$

and solving this for  $f$  completes the proof of the theorem.  $\square$

The generating function from Theorem 3.5 has arisen at least twice before in the study of permutation patterns. The Theory of Computing Research Group at the University of Otago [2] showed that it counts  $\text{Av}(2431, 4231, 4321)$ , while Brignall, Huczynska, and Vatter [14] showed that it enumerates  $\text{Av}(2143, 2413, 3142)$ . None of these three classes are symmetries of each other so this is an example of a nontrivial “Wilf-equivalence”.

## 4 With a Queue Between — PQS

There is another way to characterize the permutations which are PS-sortable, which we introduce now because we will use it to characterize the PQS-sortable permutations. A *divided permutation* is a permutation equipped with one or more dividers  $|$ , i.e.,  $\pi_1|\pi_2|\cdots|\pi_t$ . We refer to  $\pi_1|\pi_2|\cdots|\pi_t$  as a *division* of the concatenated permutation  $\pi_1\pi_2\cdots\pi_t$ , and we refer to each  $\pi_i$  as a *block* of this division. We say that the divided permutation  $\sigma_1|\sigma_2|\cdots|\sigma_s$  is contained in the divided permutation  $\pi_1|\pi_2|\cdots|\pi_t$  if  $\pi_1\pi_2\cdots\pi_t$  contains a subsequence order isomorphic to  $\sigma_1\sigma_2\cdots\sigma_s$  in which the entries corresponding to each  $\sigma_i$  come from the same block, and no other entries of this subsequence come from that block. For example:

- $513|4|2$  contains  $32|1$  because of the subsequence  $532$ , but
- $32|1$  is not contained in  $51|34|2$  despite the subsequence  $532$ .

In particular, if  $\sigma$  contains no dividers, then  $\sigma$  is contained in  $\pi_1|\pi_2|\cdots|\pi_t$  if and only if  $\sigma$  is contained in a single block  $\pi_i$ . The notion of divided permutations is due originally to Brignall, Huczynska, and Vatter [14], who used it to study cyclic closures of permutation classes.

Given a set  $D$  of divided permutations, we are interested in those permutations which can be divided to avoid all elements of  $D$ . Note that all such sets of permutations will form permutation classes.

**Proposition 4.1.** *The permutation  $\pi$  can be sorted by PS if and only if divisions can be added to  $\pi$  to obtain a divided permutation which avoids  $21$ ,  $2|13$ , and  $2|3|1$ .*

*Proof.* We view the divisions as marking the moments when we transfer all contents of the pop stack to the stack (with a final transfer occurring at the end of reading the permutation). Thus if there is no such division of  $\pi$ , in any sorting of this permutation there will come a time when either:

- the pop stack contains an increase when read from top to bottom (21),
- the stack contains an entry which lies between two entries of the pop stack in value (2|13), or
- the stack contains a decrease when read from top to bottom (2|3|1).

Any of these three situations will cause the sorting to fail.

Conversely, suppose that divisions can be added to  $\pi$  to obtain a divided permutation which avoids 21, 2|13, and 2|3|1. To show  $\pi$  is sortable, we need to show that none of the transfers dictated by these divisions forces an inversion in the stack. We know a set of entries moved by a single transfer will not form an inversion in the stack since the divided permutation avoids 21. The only other way an inversion could be forced within the stack is if there is an entry 2 that was previously transferred to the stack, an entry 3 that is transferred to the stack later, and an entry 1 that is transferred at the same time as or later than the 3 (and thus not allowing the 2 to be output before the 3 enters the stack). Notice these PS movements imply that there is a 2|13, a 2|31 (which means there is a 21), or a 2|3|1. This shows that  $\pi$  is sortable by PS.  $\square$

The analogue of Proposition 4.1 for the PQS machine is the following.

**Proposition 4.2.** *The permutation  $\pi$  can be sorted by PQS if and only if divisions can be added to  $\pi$  to obtain a divided permutation which avoids 132, 2|13, 32|1, and 2|3|1.*

*Proof.* A permutation can be sorted with a stack if and only if it avoids 231, so we need to show that we can fill the queue between the pop stack and the stack with a 231-avoiding permutation if and only if  $\pi$  can be divided in the manner specified.

First suppose that  $\pi$  can be divided as  $\pi_1|\pi_2|\cdots|\pi_t$  so that this division avoids the four divided permutations 132, 2|13, 32|1, and 2|3|1. Consider pushing each element of  $\pi_1$  into the pop stack and then popping the entire pop stack into the queue, then pushing each element of  $\pi_2$  onto the pop stack and then popping the entire pop stack into the queue, and so on. This fills the queue with the permutation  $\pi_1^r\pi_2^r\cdots\pi_t^r$ . Now consider the four different ways this permutation could contain 231: if all three entries are in  $\pi_i^r$  then  $\pi_i^r$  contains 132, if the first entry is in  $\pi_i^r$  and the other two are in  $\pi_j^r$  for  $i < j$  then  $\pi_i|\pi_j$  contains 2|13, if the first two entries are in  $\pi_i^r$  and the last is in  $\pi_j^r$  for  $i < j$  then  $\pi_i|\pi_j$  contains 32|1, and finally, if all three entries are in different blocks, then  $\pi_1|\pi_2|\cdots|\pi_t$  contains 2|3|1. The other direction follows immediately.  $\square$

The PS-sortable permutations can be characterized by a finite number of forbidden divided patterns (Proposition 4.1) and by a finite basis (Theorem 3.4). However, it does *not* follow that every class defined by finitely many forbidden divided patterns has a finite basis, as we show in the next section. Nor is it apparent how to convert such a list of divided patterns into a basis. In the case of the PQS machine, we have

not been able to verify that it is finitely based, although computations performed by Michael Albert strongly suggest this to be the case.

**Conjecture 4.3.** *The basis of the class of PQS-sorting permutations consists of 108 permutations, all of length at most 9.*

The enumeration of this class is

$$1, 2, 6, 24, 120, 685, 4148, 25661, 159829, 997870, \dots,$$

sequence [A214611](#) in the [OEIS](#) [23].

## 5 Avoiding Divided Permutations in General

As remarked in the previous section, classes defined by avoiding finitely many divided permutations need not be finitely based. Here we give an example. Our basis will contain the infinite antichain of permutations referred to as  $U$ :

$$\begin{aligned} u_1 &= 2, 3, 5, 1, 6, 7, 4 \\ u_2 &= 2, 3, 5, 1, 7, 4, 8, 9, 6 \\ u_3 &= 2, 3, 5, 1, 7, 4, 9, 6, 10, 11, 8 \\ &\vdots \\ u_k &= 2, 3, 5, 1, 7, 4, 9, 6, 11, 8, \dots, 2k+3, 2k, 2k+4, 2k+5, 2k+2. \end{aligned}$$

Each member of  $U$  has precisely two copies of 2341: its first four entries, and the first, second, third, and fifth entries from the right, and this observation can be used to prove that  $U$  is indeed an antichain (see Atkinson, Murphy, and Ruškuc [8] for such a proof). We call the first four entries in  $u_k$  the *head*, the last five entries the *tail*, and the entries between the *midsection*.

**Proposition 5.1.** *Every member of the infinite antichain  $U$  is a basis element for the permutation class defined by avoiding the divided permutations*

- (a) 2341.
- (b) 234|1, 23|4|1, 2|34|1, 2|3|4|1,
- (c) 314|2, 31|42, and 31|4|2.

*Proof.* Consider any member of the antichain  $U$ . In order to avoid 2341, the head — 2351 — must be divided. However, to avoid 234|1, 23|4|1, 2|34|1, and 2|3|4|1, this division cannot occur between the 5 and the 1. Therefore there is a block containing 51. Now consider the four entries starting with 51, which in  $u_k$  for  $k \geq 2$  consist of 5174. As these entries are order isomorphic to 3142 and the 51 block is not divided, in order to avoid 314|2, 31|42, and 31|4|2, they cannot be divided. This propagates throughout the midsection of  $u_k$ , and at the end of the process, we see that the

entries  $5, 1, 7, 4, 9, 6, 11, 8, \dots, 2k + 1, 2k - 2$  all lie in the same block. Now consider the tail. Because the entries  $2k + 1, 2k - 2, 2k + 3, 2k$  are order isomorphic to 3142 and the  $2k + 1$  and  $2k - 2$  are not divided, there cannot be a division between them. The same argument shows that the entries  $2k + 3, 2k, 2k + 5, 2k + 2$  lie in the same block. However, this implies that this block contains the entire tail, which is order isomorphic to 2341.

We must now argue that by removing any entry of an element of  $U$  we obtain a permutation in the class, which follows from a case analysis. Typically, when an entry is removed one would relabel the remaining entries so that they consist of the numbers 1 through  $n - 1$ , but it is easier to make this argument without relabeling, so, for example, if the 1 is removed, we talk about the permutation beginning with 235.

Suppose an entry of the head is removed. In this case, the remaining elements of the head do not form a copy of 2341, so we are able to insert a division immediately preceding the 1 (assuming the 1 was not removed), or immediately following the 5 (assuming the 5 was not removed). Because of this division, we can then insert divisions between every two entries in the midsection. This then allows us to add a division in the tail between  $2k + 3$  and  $2k$ , and the resulting division avoids the divided permutations desired. For example, if the 3 is removed from  $u_5$  we have the division

$$2 \ 3 \ 5 \mid 1 \mid 7 \mid 4 \mid 9 \mid 6 \mid 11 \mid 8 \mid 13 \mid 10 \ 14 \ 15 \ 12.$$

Suppose now that an entry from the midsection is removed. In this case the permutation splits into a direct sum of two shorter permutations, one consisting of the head and forward midsection, and the other consisting of the rear midsection and the tail. We can then add a division in the first sum component between the 2 and the 3. In the second component, we add divisions between all entries in the rear midsection, and then between the  $2k + 3$  and  $2k$  in the tail. It is straight-forward to check that the resulting division avoids the desired divided permutations. For example, if the 9 is removed from  $u_5$  we have the division

$$2 \mid 3 \ 5 \ 1 \ \underline{7 \ 4} \ \overline{9 \ 6} \mid 11 \mid 8 \mid 13 \mid 10 \ 14 \ 15 \ 12,$$

where forward and rear midsections are denoted by under- and over-lining, respectively.

To complete the proof, suppose that an entry from the tail is removed. If this entry is part of the copy of 2341, then we can simply add a division between the 2 and 3 at the beginning of the permutation. For example, if the 14 is removed from  $u_5$ , we obtain

$$2 \mid 3 \ 5 \ 1 \ 7 \ 4 \ 9 \ 6 \ 11 \ 8 \ 13 \ 10 \ \cancel{14} \ 15 \ 12.$$

Otherwise, the entry  $2k$  was removed. In this case, the  $2k + 3$  and  $2k + 4$  are not involved in a copy of 3142, and so we can add a division between them, and also between the 2 and the 3. For example, if the 10 is removed from  $u_5$ , we obtain

$$2 \mid 3 \ 5 \ 1 \ 7 \ 4 \ 9 \ 6 \ 11 \ 8 \ 13 \mid \cancel{10} \ 14 \ 15 \ 12.$$

This final case completes the proof. □

## 6 Concluding Remarks

Note that the pop stack can never contain a noninversion (when read from top to bottom) in the PS machine. Thus PS sorting is a special case of sorting with a decreasing stack followed by an increasing stack, the DI machine, which has been studied by Smith [21]. However, there is no relation between DI-sortable and PQS-sortable permutations — 3142 can be sorted by PQS but not DI, while 465132 can be sorted by DI but not PQS.

We have demonstrated with Proposition 5.1 that even though the PQS sortable permutations can be characterized by finitely many divided patterns, this alone does not imply that the class has a finite basis. However, there is another possible generalization of this problem. Let us say that the permutation  $\sigma$  can be obtained from  $\pi$  by *local reversals* if  $\pi = \pi_1\pi_2\cdots\pi_t$  and  $\sigma = \pi_1^r\pi_2^r\cdots\pi_t^r$ . Thus the PQS-sortable permutations are those that can be obtained by local reversals from the 231-avoiding permutations.

**Question 6.1.** *Let  $\mathcal{C}$  be a permutation class and  $\mathcal{D}$  the class of all permutations that can be obtained by local reversals from members of  $\mathcal{C}$ . If  $\mathcal{C}$  is finitely based, must  $\mathcal{D}$  also be finitely based?*

Our suspicion is that the answer to Question 6.1 is “no”. This suspicion is based on an example found by the Theory of Computing Research Group at the University of Otago [3]. They considered a machine denoted  $\mathbb{T}$ , called a *transposition switch*. Given a permutation  $\pi$ ,  $\mathbb{T}$  returns the set of permutations which can be generated from  $\pi$  by disjoint adjacent transpositions of entries of  $\pi$ . This operation is then extended to sets (and thus, classes) of permutations in the natural way. The Otago group showed that while  $\mathbb{T}^4(\text{Av}(21))$  is finitely based,  $\mathbb{T}^k(\text{Av}(21))$  is infinitely based for all  $k \geq 5$ . Thus  $\mathbb{T}$  does not preserve finite bases.

## Acknowledgments

The analysis of the PS machine in Section 3 was aided by Michael Albert’s *PermLab* program [1]. The authors would also like to thank Albert for computing the basis in Conjecture 4.3 and drawing our attention to [3], and Daniel Rose for his L<sup>A</sup>T<sub>E</sub>X macros for drawing two stacks in series. Finally, the authors would like to thank the referees for catching an embarrassing amount of typos, and providing many valuable comments.

## References

- [1] M. H. Albert, PermLab: Software for permutation patterns, available online at <http://www.cs.otago.ac.nz/PermLab>, 2012.
- [2] M. H. Albert, R. E. L. Aldred, M. D. Atkinson, C. C. Handley, D. A. Holton and D. J. McCaughan, Sorting classes, *Electron. J. Combin.* 12 (2005), # 31, 25 pp.

- [3] M. H. Albert, R. E. L. Aldred, M. D. Atkinson, H. P. van Ditmarsch, C. C. Handley, D. A. Holton and D. J. McCaughan, Compositions of pattern restricted sets of permutations. *Australas. J. Combin.* (2007) 37, 43–56.
- [4] M. H. Albert, M. Atkinson and S. Linton, Permutations generated by stacks and dequeues, *Ann. Combin.* 14(1) (2010), 3–16.
- [5] M. H. Albert and M. D. Atkinson, Simple permutations and pattern restricted permutations, *Discrete Math.* 300(1-3) (2005), 1–15.
- [6] M. H. Albert, M. D. Atkinson and V. Vatter, Inflations of geometric grid classes of permutations: three case studies, arXiv:1209.0425 [math.CO].
- [7] M. D. Atkinson, Generalized stack permutations, *Combin. Probab. Comput.* 7 (1998), 239–246.
- [8] M. D. Atkinson, M. M. Murphy and N. Ruškuc, Partially well-ordered closed sets of permutations, *Order* 19(2) (2002), 101–113.
- [9] M. D. Atkinson, M. M. Murphy and N. Ruškuc, Sorting with two ordered stacks in series, *Theoret. Comput. Sci.* 289(1) (2002), 205–223.
- [10] M. D. Atkinson and T. Stitt, Restricted permutations and the wreath product, *Discrete Math.* 259(1-3) (2002), 19–36.
- [11] D. Avis and M. Newborn, On pop-stacks in series, *Utilitas Math.* 19 (1981), 129–140.
- [12] M. Bóna, Exact enumeration of 1342-avoiding permutations: a close link with labeled trees and planar maps, *J. Combin. Theory Ser. A* 80(2) (1997), 257–272.
- [13] M. Bóna, A survey of stack-sorting disciplines, *Electron. J. Combin.* 9(2) (2003), # 1, 16 pp.
- [14] R. Brignall, S. Huczynska and V. Vatter, Simple permutations and algebraic generating functions, *J. Combin. Theory Ser. A* 115(3) (2008), 423–441.
- [15] M. Elder, Permutations generated by a stack of depth 2 and an infinite stack in series, *Electron. J. Combin.* 13 (2006), # 68, 12 pp.
- [16] S. Even and A. Itai, Queues, stacks, and graphs, In *Theory of machines and computations (Proc. Internat. Sympos., Technion, Haifa, 1971)*, pp. 71–86, Academic Press, New York, 1971.
- [17] D. E. Knuth, *The art of computer programming, Volume 1*, Addison-Wesley Publishing Co., Reading, Mass., 1968. Fundamental Algorithms.
- [18] D. E. Knuth, *The art of computer programming, Volume 3*, Addison-Wesley Publishing Co., Reading, Mass., 1973. Sorting and searching.

- [19] M. M. Murphy, *Restricted Permutations, Antichains, Atomic Classes, and Stack Sorting*, PhD Thesis, Univ. St Andrews, 2002.
- [20] V. R. Pratt, Computing permutations with double-ended queues, parallel stacks and parallel queues, In *STOC '73: Proc. Fifth Annual ACM Symp. on Theory of Computing*, pp. 268–277, New York, NY, USA, 1973. ACM Press.
- [21] R. Smith, A decreasing stack and an increasing stack in series, arXiv:1301.6801 [math.CO].
- [22] R. Tarjan, Sorting using networks of queues and stacks, *J. Assoc. Comput. Mach.* 19 (1972), 341–346.
- [23] The On-line Encyclopedia of Integer Sequences, <http://oeis.org/>.
- [24] J. West, Sorting twice through a stack, *Theoret. Comput. Sci.* 117(1-2) (1993), 303–313.

(Received 7 Mar 2013)