Theoretical Computer Science 410 (2009) 3372–3390

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

An $(18/11)n$ upper bound for sorting by prefix reversals

B. Chitturi, W. Fahle, Z. Meng, L. Morales, C.O. Shields, I.H. Sudborough*, W. Voit

Computer Science Department, Erik Jonsson School of Engineering and Computer Science, University of Texas at Dallas, Richardson, TX 75080, United States

ARTICLE INFO

Keywords:
Pancake problem
Pancake network
Sorting by prefix reversals
Permutations
Upper bounds

ABSTRACT

The pancake problem asks for the minimum number of prefix reversals sufficient for sorting any permutation of length n. We improve the upper bound for the pancake problem to $(18/11)n + O(1) \approx (1.6363)n$.

1. Introduction

The problem of sorting with the minimum number of prefix reversals, known as the *Pancake Problem*, has applications in the pancake network in parallel processing [1]. Improving the upper bound for the pancake problem has been a subject of some interest. Gates and Papadimitriou [3] proved that all permutations of length n can be sorted using at most $(5n + 5)/3 \approx (1.6666)n$ prefix reversal operations. We improve the upper bound for the pancake problem to $(18/11)n + O(1) \approx (1.6363)n$. The current lower bound is $(15/14)n \approx (1.071)n$ [4]. The exact number of prefix reversals needed to sort unsigned permutations of length n (for $n < 18$), and signed permutations (for $n < 11$) has also been determined [2,4–7].

We denote a permutation π on $Z_n = \{0, 1, \ldots, n-1\}$ by a list $\pi = \pi_1, \pi_2, \ldots, \pi_n$, where each element of Z_n occurs once and only once. Symbols π_i and π_j in Z_n are *consecutive* if $\pi_i - \pi_j \equiv \pm 1 (\text{mod } n)$. Note that this means that symbols $n - 1$ and 0 are consecutive symbols. An *adjacency* occurs between π_i and π_{i+1}, whenever π_i and π_{i+1} are consecutive symbols. For example, in the permutation 4, 5, 6, 0, 2, 1, 3, there are adjacencies between 4 and 5, between 5 and 6, between 6 and 0, and between 2 and 1.

A *block* in a permutation $\pi = \pi_1, \pi_2, \ldots, \pi_n$ is a maximal length sublist $\pi_r, \pi_{r+1}, \ldots, \pi_s$ such that there is an adjacency between π_i and π_{i+1} for all i ($r \le i < s$). A block has a distinct initial and final element, which we call its *endpoints*. An element not occurring in a block is called a *singleton*. Note that unless the permutation is a single block, there are two unique consecutive elements outside a block, one for each of its endpoints. The consecutive element for each endpoint can either be a singleton or an endpoint of another block.

A singleton also has two unique consecutive elements elsewhere in the permutation which can be singletons, endpoints of different blocks, or one of each. For example, the permutation 4, 5, 6, 0, 2, 1, 3, has two blocks, namely 4, 5, 6, 0 and 2, 1, and one singleton, namely 3. The block 4, 5, 6, 0 has endpoints 4 and 0. The endpoint 4 is consecutive to the singleton 3, and the other endpoint 0 is consecutive to 1 which is an endpoint of another block.

For a permutation π, we denote the number of blocks by $b(\pi)$ and the number of singletons by $s(\pi)$. A *prefix reversal* of size j transforms $\pi = \pi_1, \pi_2, \ldots, \pi_{j-1}, \pi_j, \pi_{j+1}, \ldots, \pi_n$ into $\pi' = \pi_j, \pi_{j-1}, \ldots, \pi_2, \pi_1, \pi_{j+1}, \ldots, \pi_n$. We use the terms *prefix reversal*, *step*, *move* and *flip* interchangeably.

For any permutation π, we define a potential function $\Phi(\pi) = (18/11)s(\pi) + (24/11)b(\pi)$ and use this function to prove our new upper bound. Our proof contains 2220 cases and a complete solution with all flip sequences is given in [8]. We show that $\Phi(\pi)$ is an upper bound on the number of prefix reversal operations to transform π into a single block by

* Corresponding author.
E-mail address: hal@utdallas.edu (I.H. Sudborough).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2008.04.045

Table 1
The nine Gates and Papadimitriou cases

Case	Flip sequence	Description	Number of flips	Δ(S)	Excess (+) or Deficiency (−)
(1)	B _ A _ → _ BA _	Singleton B at the beginning of the permutation is consecutive with a singleton A. Two singletons are eliminated and one block is created in one step.	1	12/11	+1/11
(2)	B _ A∼ _ → _ BA∼	Singleton B at the beginning of the permutation is consecutive with the left endpoint A of a block A∼. One singleton is eliminated in one step.	1	18/11	+7/11
(3)	B _ ∼A _ ∼C _ →A∼ _ B _ ∼C _ → _ ∼AB _ ∼C _ →C∼ _ BA∼ _ → _ ∼CBA∼ _	Singleton B at the beginning of the permutation is consecutive with the last elements (A and C) of 2 separate blocks ∼A and ∼C. One singleton and one block are eliminated in 4 steps.	4	42/11	−2/11
(4)	B∼ _ A _ → _ ∼BA _	Left endpoint B of block B∼ at the beginning of the permutation is consecutive with a singleton A. One singleton is eliminated in one step.	1	18/11	+7/11
(5)	B∼ _ A∼ _ → _ ∼BA∼ _	Left endpoint B in block B∼ at the beginning of the permutation is consecutive with A in block A∼. One block is eliminated in one step.	1	24/11	+13/11
(6)	B∼C _ D∼ _ →C∼B _ D∼ _ → _ B∼CD∼ _	Right endpoint C in block ∼C at the beginning is consecutive with left endpoint D in block D∼. One block is eliminated in two steps.	2	24/11	+2/11
(7)	B∼C _ ∼D _ →D∼ _ C∼B _ → _ ∼DC∼B _	Right endpoint C in block ∼C at the beginning is consecutive with right endpoint D in block ∼D. One block is eliminated in two steps.	2	24/11	+2/11
(8)	B∼C _ ∼A _ D _ →D _ A∼ _ C∼B _ → _ ∼A _ D∼B _ →B∼D _ A∼ _ → _ D∼A∼ _	The block B∼C is at the beginning, left endpoint B is consecutive with right endpoint A in block ∼A. The endpoint C of B∼C is consecutive with a singleton D occurring to the right of ∼A. One singleton and one block are eliminated in 4 steps.	4	42/11	−2/11
(9)	B∼C _ D _ ∼A _ →D _ C∼B _ ∼A _ →D∼B _ ∼A _ →A∼ _ B∼D _ → _ ∼A∼D _	The block B∼C is at the beginning, left endpoint B is consecutive with right endpoint A in block ∼A. The endpoint C of B∼C is consecutive with a singleton D occurring between the block B∼C and the block ∼A. One singleton and one block are eliminated in 4 steps.	4	42/11	−2/11

demonstrating that, for all of our flipping sequences that reduce the number of singletons and/or blocks, the reduction in the value of the potential function is greater than or equal to the number of flips. The upper bound $(18/11)n$ for permutations of length n follows, as the function $\Phi(\pi)$ is maximized over all permutations of length n when every element is a singleton, as it takes two or more elements to make a block.

Gates and Papadimitriou [3] define an algorithm based on nine cases that describe the relative position and orientation of certain blocks and singletons in permutations. We now give an overview of the nine cases. We apply our potential function Φ to the nine cases.

The nine cases are categorized by whether the initial object is a singleton or a block and whether elements consecutive with the initial object (or its endpoints) are singletons or blocks. Every permutation belongs to at least one of these nine cases. To use the algorithm, one first determines which case(s) a given permutation belongs to and then performs the prefix reversals indicated for the case. After performing the specified prefix reversals, one again has a permutation and the process is repeated, although the new permutation may not belong to the same case(s) as before.

To describe the nine cases, we adopt the notation that consecutive letters are variables denoting consecutive symbols and a block with indeterminate elements is represented by the symbol ∼. A block with left (right) endpoint B is denoted by B∼ (respectively, ∼B), and a block with left and right elements B and C, respectively, is denoted by B∼C, although here the consecutive letters B and C do not necessarily denote consecutive symbols in the permutation. An underscore denotes an arbitrary sub-list of elements in the permutation that do not play a role in the flip sequence.

As noted earlier, if the initial object in a permutation, say B, is a singleton, then there are two elements somewhere in the permutation, say A and C, that are consecutive with B. The first three cases of the Gates and Papadimitriou algorithm are for permutations in which the initial object is such a singleton B. Case (1) applies when at least one of B's consecutive elements is a singleton, which we denote by A. Case (2) applies when at least one of B's consecutive elements is the beginning endpoint of a block, which we denote by A∼, and Case (3) applies when both of B's consecutive elements are terminating endpoints of distinct blocks, which we denote by ∼A and ∼C, respectively. Cases (1)–(3) are shown in Table 1. Every permutation with an initial singleton element belongs to one of these three cases.

Now consider a permutation whose initial element is a block, which we denote by B∼C. This gives rise to the remaining six cases of the Gates and Papadimitriou algorithm. These cases are differentiated by the type of elements (*i.e.*, singletons or endpoints of blocks) that are consecutive with the endpoints B and C of the initial block. Case (4) applies when the element consecutive with the beginning endpoint B, which we denote by A, is a singleton. Case (5) applies when the element consecutive with B is the beginning endpoint of another block, which we denote by A∼. That leaves permutations where the element consecutive with B is the terminating endpoint of another block, which we denote by ∼A. However, this situation by itself is not good for constructing an efficient algorithm, because too many steps are required to make the beginning endpoint B adjacent to its consecutive element A.

To resolve this issue, Gates and Papadimitriou consider the element, say D, consecutive with the terminating endpoint C of the initial block. This gives rise to Cases (6) and (7). Case (6) applies when D is the beginning endpoint of another block, which we denote by D∼. Case (7) applies when D is the terminating endpoint of another block, which we denote by ∼D. Both of these cases can be resolved in an acceptable number of steps. The only remaining possibility is that D is a singleton. That is, the only permutations not included in Cases (1) through (7) are those whose initial element is a block B∼C, the element, A, consecutive with B is the terminating endpoint of another block, denoted by ∼A, and the element, D, consecutive with C is a singleton. Case (8) applies when ∼A occurs to the left of D in the permutation. Case (9) applies when ∼A occurs to the right of D. These cases can also be resolved in an acceptable number of steps [3]. Cases (4)–(9) are shown in Table 1.

It follows that, for every permutation π, at least one of the nine cases applies. The Gates and Papadimitriou algorithm describes sequences of prefix reversals for each case that are sufficient to prove the indicated $(5n + 5)/3$ upper bound [3].

Observe that, in the preceding discussion, Cases (6)–(9) deal with the situation where the initial object is a block B∼C and B's consecutive element is the terminating endpoint of another block, denoted by ∼A. This single configuration was expanded in [3] to the four cases (6)–(9). Through this expansion, more information about the structure of the permutation is available. In particular, by considering the position of the singleton D consecutive with C, a good sequence of prefix reversals can be described.

For our improved upper bound, a similar strategy is used. We design an improved algorithm by examining cases in more detail, which results in further division of the nine cases. Specifically, we start with the nine cases described above, which provide a complete solution for the previous upper bound. From these nine cases, we construct a refined set of cases by *compounding* which we now describe.

Suppose that some Case (i), ($1 \le i \le 9$), applies to a permutation π. Then by applying the prefix reversals defined by the Gates and Papadimitriou algorithm for Case (i), one obtains a new permutation π' and some Case (j) applies to π'. We call this two-case sequence a *compound case*, and we say compound Case (i)–(j) applies to π. As every permutation belongs to such a compound case, this perspective also yields a complete set of cases. (The only exception would be when π becomes a single block by the steps of the initial Case (i).) Furthermore, as we shall see, by considering such compound cases we can design more efficient prefix reversal sequences. For example, there is a four-step flip sequence described in [3] for Case (3). Therefore, a permutation in compound Case (3-3) would take eight steps following the algorithm of [3]. Note that there are several compound (3-3) cases, because the objects that form the subsequent Case (3) can be in different locations relative to the objects of the initial Case (3). Table 5 shows that seven steps are sufficient in all but two of the compound (3-3) cases. This is our motivation to consider compound cases and the foundation of our improved algorithm.

In fact, we have found shorter flip sequences for most, but not all, compound cases. There are compound cases for which shorter sequences do not exist. We resolve these by additional expansion. Specifically, we consider additional elements in positions adjacent to other already identified elements. For these additional elements, we then consider all possible positions of objects consecutive with the new elements. We call this *expansion by breadth*.

For example, one of the difficult sub-cases of the compound case (3-3) is any permutation of the form B _ H∼ _ J∼ _ ∼A I _ ∼C _, shown as (3-3a) in Table 5. This denotes a sub-case involving an initial Case (3), with the initial singleton denoted by B, and B's two consecutive elements are the terminal endpoints of blocks, denoted by ∼A and ∼C, the subsequent Case (3) involves a singleton I next to ∼A and somewhere between the singleton B and the block ∼A are the blocks H∼ and J∼ containing the elements, namely H and J, consecutive with I.

To find an efficient sequence of flips for this case we consider an expansion of the permutation into several sub-cases, represented by the *generating string* B _ H∼ _ J∼ _ ∼A I _ S∼C _ T*. The generating string denotes all sub-cases, where the beginning endpoint of the block ∼C is denoted by the symbol S and the element consecutive to S is denoted by the symbol T, where all possible positions of T, both as a singleton or endpoint of a block, are considered. That is, the compact notation T* indicates all sub-cases based on possible positions of the element T.

Note that there must be an initial element S of such a block and its consecutive element T must be somewhere in the permutation, so the indicated expansion gives again a complete set of sub-cases. That is, *every* permutation of the compound (3-3a) case B _ H∼ _ J∼ _ ∼A I _ ∼C _ is covered by one of the indicated sub-cases.

Although the flip sequences are not shown here, Table 15 in the Appendix indicates that there is an efficient flip sequence for all but one of these sub-cases. The exception is the sub-case denoted by B _ H∼ _ J∼ _ ∼A I _ S∼C _ ∼T _, where the element T is the terminating endpoint of a block that appears after the block S∼C. To find a good flip sequence for this exceptional sub-case, we consider a further expansion into B _ H∼ _ J∼ _ U∼A I _ S∼C _ ∼T _ V*.

Table 2
(3-1) compound cases

Case and excess (+) or Deficiency (−) of Δ(S)	Description	Flip sequence (not given for deficiencies)
(3-1a) Deficiency: −1/11	B _ H _ ∼A I _ ∼C _	(Deficiency)
(3-1b) Deficiency: −1/11	B _ ∼A I _ H _ ∼C _	(Deficiency)
(3-1c) Deficiency: −1/11	B _ ∼A I _ ∼C _ H _	(Deficiency)

Table 3
(8-1) compound cases

Case and excess (+) or Deficiency (−) of Δ (S)	Description	Flip sequence (not given for deficiencies)
(8-1a) Excess: +3/11	B∼C _ H _ ∼A I _ D _	B∼C _ H|_ ∼A I _ D _ →H _ C∼B _ ∼A|I _ D _ →A∼ _|B∼C _ H∼I _ D _ →_ ∼C _ H∼I _ D _
(8-1b) Deficiency: −1/11	B∼C _ ∼A I _ H _ D _	(Deficiency)
(8-1c) Deficiency: −1/11	B∼C _ ∼A I _ D _ H _	(Deficiency)

Table 4
(9-1) compound cases

Case and excess (+) or Deficiency (−) of Δ(S)	Description	Flip sequence (not given for deficiencies)
(9-1a) Deficiency: −1/11	B∼C _ G _ D F _ ∼A _	(Deficiency)
(9-1b) Deficiency: −1/11	B∼C _ D F _ G _∼A _	(Deficiency)
(9-1c) Deficiency: −1/11	B∼C _ D F _ ∼A _ G _	(Deficiency)

This is another example of a generating string, and it is interpreted as follows. The beginning endpoint of the block ∼A is identified as the symbol U. V denotes the element outside the block that is consecutive with U. We have sub-cases based on all possible positions of the element V in the permutation. Note that, once again, there must be an initial element U of the ∼A block and U's consecutive element must be somewhere in the permutation, so this expansion again gives a complete set of cases. As indicated in Table 15 there are efficient flip sequences for each of these sub-cases, so no further expansion is required.

For a given sequence of steps S that transforms a permutation π into a permutation π', let Δ(S) denote $\Phi(\pi) - \Phi(\pi')$. For a sequence S of k prefix reversals, for some $k \geq 1$, let the *excess of S*, denoted by excess*(S)*, be $\Delta(S) - k$. If excess(S)≥0, then S is called a *good* sequence. If excess(S)<0, then the excess of S will be called the *deficiency of S* and the sequence S is *bad*. Column 6 of Table 1 shows this value for each of the nine Gates and Papadimitriou cases.

The preceding paragraphs describe our expansions in the number of cases. Note that only three of the nine cases of Gates and Papadimitriou need expansion when using our potential function $\Phi(\pi) = (18/11)s(\pi) + (24/11)b(\pi)$. This is recorded in Table 1, where a deficiency (indicated by a negative Δ(S)) shows when an expansion is necessary. In particular, expansions are only necessary for Cases (3), (8), and (9). So, we only need to consider compound cases of the form (3-y), (8-y), (9-y), where y is a case number between 1 and 9. Moreover, we do not need to look for good sequences for compound cases (3-y), (8-y), or (9-y), where y∈{2, 4, 5, 6, 7}, as the flip sequences of [3] are sufficient for our potential function. This can be seen by observing that the deficiency for the initial cases (3), (8), and (9) is offset by the excess (indicated by a positive Δ(S)) for the subsequent cases y∈{2, 4, 5, 6, 7}, so that the cumulative Δ(S) for these compound cases is positive. Hence, we only need to find good flip sequences for compound cases (3-y), (8-y), and (9-y), where y∈{1, 3, 8, 9}.

We have identified 93 compound cases which require expansion. For example, for the (3-3) compound cases there are nine ways that the objects in a succeeding case (3) can be arranged relative to the initial case (3). These are labeled as (3-3a), (3-3b), …, (3-3i) and are shown in Table 5. Tables 2–4 and Tables 6–13 show the other compound cases which require expansion.

Among the 93 compound cases where good sequences must be found, and which are described in Table 2 through Table 13, eleven fail. That is, for eleven of the 93 sub-cases, no flip sequence exists that is sufficient to satisfy our potential function. These are indicated by the word *deficiency* in the tables. In particular, the compound cases denoted by (3-1a), (3-1b),

Table 5
(3-3) compound cases

<table>
<tr><th>Case and excess (+) or Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(3-3a)
Deficiency: −2/11</td><td>B _ H∼ _ J∼ _ ∼A I _ ∼C _</td><td>(Deficiency)</td></tr>
<tr><td>(3-3b)
Excess = +7/11</td><td>B _ J∼ _ ∼A I _ ∼C _ ∼H _</td><td>B _ J∼ _ ∼A|I _ ∼C _ ∼H _
→A∼ _ ∼J _|B I _ ∼C _ ∼H _
→_ J∼ _ ∼B I _ ∼C _ ∼H|_
→H∼ _ C∼ _|I B∼ _ ∼J _ _
→_ ∼C _ ∼I B∼ _ ∼J|_ _
→J∼ _ ∼B|I∼ _ C∼ _ _ _
→B∼ _ ∼ _|C∼ _ _ _
→_ ∼ _ ∼ _ _ _</td></tr>
<tr><td>(3-3c)
Excess = +7/11</td><td>B _ H∼ _ ∼A I _ ∼J _ ∼C _</td><td>B _ H∼ _ ∼A I _ ∼J|_ ∼C _
→J∼ _|I A∼ _ ∼H _ B _ ∼C _
→_ ∼I A∼ _ ∼H|_ B _ ∼C _
→H∼ _ ∼A|I∼ _ _ B _ ∼C _
→A∼ _ ∼ _ _|B _ ∼C _
→_ _ ∼ _ ∼B _ ∼C|_
→C∼ _|B∼ _ ∼ _ _ _
→_ ∼ _ ∼ _ _ _</td></tr>
<tr><td>(3-3d)
Excess = +7/11</td><td>B _ ∼A I _ ∼H _ ∼J _ ∼C _</td><td>B _ ∼A I _ ∼H|_ ∼J _ ∼C _
→H∼ _|I A∼ _ B _ ∼J _ ∼C _
→_ ∼I A∼ _ B _ ∼J|_ ∼C _
→J∼ _ B _ ∼A|I∼ _ _ ∼C _
→A∼ _|B _ ∼ _ _ ∼C _
→_ ∼B _ ∼ _ _ ∼C|_
→C∼ _ _ ∼ _|B∼ _ _
→_ ∼ _ _ ∼ _ _</td></tr>
<tr><td>(3-3e)
Excess = +7/11</td><td>B _ ∼A I _ ∼J _ ∼C _ ∼H _</td><td>B _ ∼A I _ ∼J|_ ∼C _ ∼H _
→J∼ _|I A∼ _ B _ ∼C _ ∼H _
→_ ∼I A∼ _ B _ ∼C _ ∼H|_
→H∼ _ C∼ _ B _ ∼A|I∼ _ _
→A∼ _|B _ ∼C _ ∼ _ _
→_ ∼B _ ∼C|_ ∼ _ _
→C∼ _|B∼ _ _ ∼ _ _
→_ ∼ _ _ ∼ _ _</td></tr>
<tr><td>(3-3f)
Excess = +7/11</td><td>B _ ∼A I _ ∼C _ ∼H _ ∼J _</td><td>B _ ∼A|I _ ∼C _ ∼H _ ∼J _
→A∼ _|B I _ ∼C _ ∼H _ ∼J _
→_ ∼B I _ ∼C _ ∼H|_ ∼J _
→H∼ _ C∼ _|I B∼ _ _ ∼J _
→_ ∼C _ ∼I B∼ _ _ ∼J|_
→J∼ _ _ ∼B|I∼ _ C∼ _ _
→B∼ _ _ ∼ _|C∼ _ _
→_ ∼ _ _ ∼ _ _</td></tr>
<tr><td>(3-3g)
Excess = +7/11</td><td>B _ H∼A I _ ∼J _ ∼C _</td><td>B _ H∼A I _ ∼J|_ ∼C _
→J∼ _|I A∼H _ B _ ∼C _
→_ ∼I A∼H|_ B _ ∼C _
→H∼A|I∼ _ _ B _ ∼C _
→A∼ _ _|B _ ∼C _
→_ _ ∼B _ ∼C|_
→C∼ _|B∼ _ _ _
→_ ∼ _ _ _</td></tr>
<tr><td>(3-3h)
Deficiency: −2/11</td><td>B _ H∼ _ J∼A I _ ∼C _</td><td>(Deficiency)</td></tr>
<tr><td>(3-3i)
Excess = +7/11</td><td>B _ J∼A I _ ∼C _ ∼H _</td><td>B _ J∼A|I _ ∼C _ ∼H _
→A∼J _|B I _ ∼C _ ∼H _
→_ J∼B I _ ∼C _ ∼H|_
→H∼ _ C∼ _|I B∼J _ _
→_ ∼C _ ∼I B∼J|_ _
→J∼B|I∼ _ C∼ _ _ _
→B∼ _|C∼ _ _ _
→_ ∼ _ _ _</td></tr>
</table>

Table 6
(3-8) compound cases

<table>
<tr><th>Case and excess (+) or Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(3-8a)
Excess = +7/11</td><td>B _ K _ H~ _ ~A I~J _ ~C _</td><td>B _ K _ H~ _ ~A I~J|_ ~C _
→J~I A~ _ ~H _|K _ B _ ~C _
→_ H~ _ ~A I~K _ B _ ~C|_
→C~ _|B _ K~I A~ _ ~H _ _
→_ ~B _ K~I A~ _ ~H|_ _
→H~ _ ~A|I~K _ B~ _ _ _
→A~ _ ~K _|B~ _ _ _ → _ K~ _ ~ _ _ _</td></tr>
<tr><td>(3-8b)
Excess = +7/11</td><td>B _ H~ _ ~A I~J _ ~C _ K _</td><td>B _ H~ _ ~A|I~J _ ~C _ K _
→A~ _ ~H _|B I~J _ ~C _ K _
→_ H~ _ ~B I~J _ ~C _ K|_
→K _ C~ _|J~I B~ _ ~H _ _
→_ ~C _ K~I B~ _ ~H|_ _
→H~ _ ~B|I~K _ C~ _ _ _
→B~ _ ~K _|C~ _ _ _ → _ K~ _ ~ _ _ _</td></tr>
<tr><td>(3-8c)
Excess = 0</td><td>B _ K _ ~A I~J _ ~H _ ~C _</td><td>B _ K _ ~A I~J|_ ~H _ ~C _
→J~I A~ _ K _ B _ ~H|_ ~C _
→H~ _ B _ K _ ~A|I~J _ ~C _
→A~ _ K _|B _ ~J _ ~C _
→_ K _ ~B _ ~J _ ~C|_
→C~ _ J~ _|B~ _ K _ _ → _ ~J _ ~ _ K _ _</td></tr>
<tr><td>(3-8d)
Excess = 0</td><td>B _ ~A I~J _ ~H _ K _ ~C _</td><td>B _ ~A I~J|_ ~H _ K _ ~C _
→J~I A~ _ B _ ~H|_ K _ ~C _
→H~ _ B _ ~A|I~J _ K _ ~C _
→A~ _|B _ ~J _ K _ ~C _
→_ ~B _ ~J _ K _ ~C|_
→C~ _ K _ J~ _|B~ _ _ → _ ~J _ K _ ~ _ _</td></tr>
<tr><td>(3-8e)
Excess = 0</td><td>B _ ~A I~J _ ~H _ ~C _ K _</td><td>B _ ~A I~J|_ ~H _ ~C _ K _
→J~I A~ _ B _ ~H|_ ~C _ K _
→H~ _ B _ ~A|I~J _ ~C _ K _
→A~ _|B _ ~J _ ~C _ K _
→_ ~B _ ~J _ ~C|_ K _
→C~ _ J~ _|B~ _ _ K _ → _ ~J _ ~ _ _ K _</td></tr>
<tr><td>(3-8f)
Excess = +7/11</td><td>B _ ~A I~J _ ~C _ ~H _ K _</td><td>B _ ~A|I~J _ ~C _ ~H _ K _
→A~ _|B I~J _ ~C _ ~H _ K _
→_ ~B I~J _ ~C _ ~H _ K|_
→K _ H~ _ C~ _|J~I B~ _ _
→_ ~C|_ ~H _ K~I B~ _ _
→C~ _ _ ~H _ K~I|B~ _ _
→I~K _|H~ _ _ ~ _ _ → _ K~ _ _ ~ _ _</td></tr>
<tr><td>(3-8g)
Excess = +7/11</td><td>B _ M _ H~ _ L~A I~C _</td><td>B _|M _ H~ _ L~A I~C _
→_ B M _ H~ _ L~A I~C|_
→C~I A~L _ ~H _ M|B _ _
→M _ H~ _|L~A I~B _ _
→_ ~H _ M~A I~B|_ _
→B~I|A~M _ H~ _ _ _
→I~M _|H~ _ _ _ → _ M~ _ _ _</td></tr>
<tr><td>(3-8h)
Excess = 0</td><td>B _ H~ _ L~A I~C _ M _</td><td>B _ H~ |_ L~A I~C _ M _
→~H|_ B _ L~A I~C _ M _
→H~ _ B _ L~A|I~C _ M _
→A~L _|B _ ~C _ M _
→_ L~B _ ~C|_ M _
→C~ _|B~L _ _ M _ → _ ~L _ _ M _</td></tr>
<tr><td>(3-8i)
Excess = 0</td><td>B _ L~A I~C _ ~H _ M _</td><td>B _ L~A I~C|_ ~H _ M _
→C~I A~L _ B _ ~H|_ M _
→H~ _ B _ L~A|I~C _ M _
→A~L _|B _ ~C _ M _
→_ L~B _ ~C|_ M _
→C~ _|B~L _ _ M _ → _ ~L _ _ M _</td></tr>
</table>

(*continued on next page*)

Table 6
(*continued*)

Case and excess (+) or Deficiency (−) of Δ(S)	Description	Flip sequence (not given for deficiencies)
(3-8j) Excess = +7/11	B _ K _ H∼A I∼J _ ∼C _	B _ K _ H∼A I∼J|_ ∼C _ →J∼I A∼H _|K _ B _ ∼C _ →_ H∼A I∼K _ B _ ∼C|_ →C∼ _|B _ K∼I A∼H _ _ →_ ∼B _ K∼I A∼H|_ _ → H∼A|I∼K _ B∼ _ _ _ →A∼K _|B∼ _ _ _ → _ K∼ _ _ _
(3-8k) Excess = +7/11	B _ H∼A I∼J _ ∼C _ K _	B _ H∼A|I∼J _ ∼C _ K _ →A∼H _|B I∼J _ ∼C _ K _ →_ H∼B I∼J _ ∼C _ K|_ →K _ C∼ _|J∼I B∼H _ _ →_ ∼C _ K∼I B∼H|_ _ →H∼B|I∼K _ C∼ _ _ _ →B∼K _|C∼ _ _ _ → _ K∼ _ _ _

Table 7
(3-9) compound cases

Case and excess (+) or Deficiency (−) of Δ(S)	Description	Flip sequence (not given for deficiencies)
(3-9a) Excess = 0	B _ H∼ _ K _ ∼A I∼J _ ∼C _	B _ H∼ |_ K _ ∼A I∼J _ ∼C _ →∼H|_ B _ K _ ∼A I∼J _ ∼C _ →H∼ _ B _ K _ ∼A|I∼J _ ∼C _ →A∼ _ K _|B _ ∼J _ ∼C _ →_ K _ ∼B _ ∼J _ ∼C|_ →C∼ _ J∼ _|B∼ _ K _ _ → _ ∼J _ ∼ _ K _ _
(3-9b) Excess = +7/11	B _ K _ ∼A I∼J _ ∼C _ ∼H _	B _ K _ ∼A I∼J|_ ∼C _ ∼H _ →J∼I A∼ _|K _ B _ ∼C _ ∼H _ →_ ∼A I∼K _ B _ ∼C|_ ∼H _ →C∼ _|B _ K∼I A∼ _ _ ∼H _ →_ ∼B _ K∼I A∼ _ _ ∼H|_ →H∼ _ _ ∼A|I∼K _ B∼ _ _ →A∼ _ _ ∼K _|B∼ _ _ → _ K∼ _ _ ∼ _ _
(3-9c) Excess = +7/11	B _ H∼ _ ∼A I∼J _ K _ ∼C _	→B _ H∼ _ ∼A I∼J _ K|_ ∼C _ →K _|J∼I A∼ _ ∼H _ B _ ∼C _ →_ K∼I A∼ _ ∼H|_ B _ ∼C _ →H∼ _ ∼A|I∼K _ _ B _ ∼C _ →A∼ _ ∼K _ _|B _ ∼C _ →_ _ K∼ _ ∼B _ ∼C|_ →C∼ _|B∼ _ ∼K _ _ _ → _ ∼ _ ∼K _ _ _
(3-9d) Excess = +7/11	B _ ∼A I∼J _ K _ ∼H _ ∼C _	B _ ∼A I∼J _ K|_ ∼H _ ∼C _ →K _|J∼I A∼ _ B _ ∼H _ ∼C _ →_ K∼I A∼ _ B _ ∼H|_ ∼C _ →H∼ _ B _ ∼A|I∼K _ _ ∼C _ →A∼ _|B _ ∼K _ _ ∼C _ →_ ∼B _ ∼K _ _ ∼C|_ →C∼ _ _ K∼ _|B∼ _ _ → _ ∼K _ _ ∼ _ _
(3-9e) Excess = +7/11	B _ ∼A I∼J _ K _ ∼C _ ∼H _	B _ ∼A I∼J _ K|_ ∼C _ ∼H _ →K _|J∼I A∼ _ B _ ∼C _ ∼H _ →_ K∼I A∼ _ B _ ∼C _ ∼H|_ →H∼ _ C∼ _ B _ ∼A|I∼K →A∼ _|B _ ∼C _ ∼K _ _ →_ ∼B _ ∼C|_ ∼K _ _ →C∼ _|B∼ _ _ ∼K _ _ → _ ∼ _ _ ∼K _ _
(3-9f) Excess = +7/11	B _ ∼A I∼J _ ∼C _ K _ ∼H _	B _ ∼A|I∼J _ ∼C _ K _ ∼H _ →A∼ _|B I∼J _ ∼C _ K _ ∼H _ →_ ∼B I∼J _ ∼C _ K|_ ∼H _ →K _ C∼ _|J∼I B∼ _ _ ∼H _ →_ ∼C _ K∼I B∼ _ _ ∼H|_ →H∼ _ _ ∼B|I∼K _ C∼ _ _ →B∼ _ _ ∼K _|C∼ _ _ → _ K∼ _ _ ∼ _

Table 7
(*continued*)

<table>
<tr><th>Case and excess (+) or Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(3-9g)
Excess = 0</td><td>B _ H∼ _ M _ L∼A I∼C _</td><td>B _ H∼ |_ M _ L∼A I∼C _
→∼H|_ B _ M _ L∼A I∼C _
→H∼ _ B _ M _ L∼A|I∼C _
→A∼L _ M _|B _ ∼C _
→_ M _ L∼B _ ∼C|_
→C∼ _|B∼L _ M _ _ → _ ∼L _ M _ _</td></tr>
<tr><td>(3-9h)
Excess = 0</td><td>B _ M _ L∼A I∼C _ ∼H _</td><td>B _ M _ L∼A I∼C|_ ∼H _
→C∼I A∼L _ M _ B _ ∼H|_
→H∼ _ B _ M _ L∼A|I∼C _
→A∼L _ M _|B _ ∼C _
→_ M _ L∼B _ ∼C|_
→C∼ _|B∼L _ M _ _ → _ ∼L _ M _ _</td></tr>
<tr><td>(3-9i)
Excess = 0</td><td>B _ L∼A I∼C _ M _ ∼H _</td><td>B _ L∼A I∼C|_ M _ ∼H _
→C∼I A∼L _ B _ M _ ∼H|_
→H∼ _ M _ B _ L∼A|I∼C _
→A∼L _|B _ M _ ∼C _
→_ L∼B _ M _ ∼C|_
→C∼ _ M _|B∼L _ _ → _ M _ ∼L _ _</td></tr>
<tr><td>(3-9j)
Excess = +7/11</td><td>B _ H∼A I∼J _ K _ ∼C _</td><td>B _ H∼A I∼J _ K|_ ∼C _
→K _|J∼I A∼H _ B _ ∼C _
→_ K∼I A∼H|_ B _ ∼C _
→H∼A|I∼K _ _ B _ ∼C _
→A∼K _ _|B _ ∼C _
→_ _ K∼B _ ∼C|_
→C∼ _|B∼K _ _ _ → _ ∼K _ _ _</td></tr>
</table>

(3-1c), (8-1b), (8-1c), (9-1a), (9-1b), (9-1c), (3-3a), (3-3h), and (8-8c) need further expansion. For each of these eleven failures we investigate sub-cases based on *expansion by breadth*.

Expansion by breadth increases the number of total cases dramatically, because it introduces new elements, and we must consider all possible positions for their consecutive elements. The number of possible arrangements increases significantly. Consequently, our description in the paper does not explicitly list all of the sub-cases. Instead, we use *generating strings* in Tables 15–17 in the Appendix, to describe expansions that are sufficient to obtain a complete set of cases.

Generating strings can be used to describe a complete list of all 2220 cases more succinctly. We have verified the correctness of our potential function $\Phi(\pi) = (18/11)s(\pi) + (24/11)b(\pi)$ in all cases. A reader may wish to independently verify the correctness. The work of such an interested reader may be made easier through the use of a Java tool available at our web site [8]. The tool can be used by copying any of the generating sequences given in Tables 15–17 in the Appendix, together with an upper bound for the lengths of flip sequences to be considered. The tool will give each of the derived sub-cases and a good flip sequence when one can be found, and indicate which sub-cases fail. As indicated in Tables 15–17 in the Appendix, all failed sub-cases have been resolved. We include these tables in the paper rather than at our web site in order to provide a permanent archival record. Readers are invited to visit our website [8] to see the complete record with all flip sequences.

The remainder of the paper describes our algorithm in detail. Section 2 enumerates the original Gates and Papadimitriou cases and gives the results of our potential function on these cases. Section 3 shows how we systematically form compound cases from the original cases and gives our flip sequences for these compound cases. Section 4 describes further expansions by breadth to resolve all sub-cases remaining from Section 3. Section 5 discusses further research and open problems.

2. Applying the potential function Φ to the Gates and Papadimitriou cases

For a given sequence of steps S that transforms a permutation π into a permutation π', let Δ(S) denote $\Phi(\pi) - \Phi(\pi')$. To show that $\Phi(\pi)$ is an upper bound on the number of steps to sort a permutation, we need to prove that for every permutation π, there is a sequence S of k prefix reversals, for some $k > 0$, that reduces the number of singletons and/or blocks, such that $\Delta(S) \geq k$. That is, we show that the potential function Φ always decreases by at least as much as the number of steps taken. From this it follows by a simple inductive argument that $\Phi(\pi)$ is an upper bound for the number of prefix reversals to transform π into a single block.

Consider again the nine cases used in the upper bound proof by Gates and Papadimitriou [3]. In Table 1, we provide a description of each of the nine cases using the notation defined in the previous paragraph. The first column gives the case number. The second column gives a flip sequence, that is, a sequence of permutations starting with the initial permutation π that defines the case, followed by one or more permutations where each permutation is transformed into the next by a prefix reversal, culminating in the final permutation π' of the sequence. These are the flip sequences described in [3]. The third column gives a verbal description of the initial permutation for each case, in terms of the initial object, and the position

Table 8
(8-3) compound cases

<table>
<tr><th>Case and excess (+) / Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence</th></tr>
<tr><td>(8-3a)
Excess: +5/11</td><td>B∼C _ H∼ _ J∼ _ ∼A I _ D _</td><td>B∼C|_ H∼ _ J∼ _ ∼A I _ D _
→C∼B _ H∼ _ J∼ _ ∼A I _|D _
→_ I A∼ _ ∼J|_ ∼H _ B∼D _
→J∼ _ ∼A|I _ _ ∼H _ B∼D _
→A∼ _ ∼I _ _ ∼H _|B∼D _
→_ H∼ _ _ I∼ _ ∼D _</td></tr>
<tr><td>(8-3b)
Excess: +7/11</td><td>B∼C _ J∼ _ ∼A I _ D _ ∼H _</td><td>B∼C _ J∼ _ ∼A I _ D|_ ∼H _
→D _ I A∼ _ ∼J _|C∼B _ ∼H _
→_ J∼ _ ∼A I _ D∼B _ ∼H|_
→H∼ _ B∼D _|I A∼ _ ∼J _ _
→_ D∼B _ ∼I A∼ _ ∼J|_ _
→J∼ _ ∼A|I∼ _ B∼D _ _ _
→A∼ _ ∼ _|B∼D _ _ _
→_ ∼ _ ∼D _ _ _</td></tr>
<tr><td>(8-3c)
Excess: +5/11</td><td>B∼C _ H∼ _ ∼A I _ ∼J _ D _</td><td>B∼C|_ H∼ _ ∼A I _ ∼J _ D _
→C∼B _ H∼ _ ∼A I _ ∼J _|D _
→_ J∼ _ I A∼ _ ∼H|_ B∼D _
→H∼ _ ∼A|I _ ∼J _ _ B∼D _
→A∼ _ ∼I _ ∼J _ _|B∼D _
→_ _ J∼ _ I∼ _ ∼D _</td></tr>
<tr><td>(8-3d)
Excess: 0</td><td>B∼C _ ∼A I _ ∼H _ ∼J _ D _</td><td>→B∼C|_ ∼A I _ ∼H _ ∼J _ D _
→C∼B _ ∼A I _ ∼H|_ ∼J _ D _
→H∼ _|I A∼ _ B∼C _ ∼J _ D _
→_ ∼I A∼ _ B∼C _ ∼J|_ D _
→J∼ _ C∼B _ ∼A|I∼ _ _ D _
→A∼ _|B∼C _ ∼ _ _ D _
→_ ∼C _ ∼ _ _ D _</td></tr>
<tr><td>(8-3e)
Excess: +7/11</td><td>B∼C _ ∼A I _ ∼J _ D _ ∼H _</td><td>B∼C|_ ∼A I _ ∼J _ D _ ∼H _
→C∼B _ ∼A I _ ∼J _|D _ ∼H _
→_ J∼ _ I A∼ _ B∼D _ ∼H|_
→H∼ _ D∼B _ ∼A|I _ ∼J _ _
→A∼ _|B∼D _ ∼I _ ∼J _ _
→_ ∼D _ ∼I _ ∼J|_ _
→J∼ _|I∼ _ D∼ _ _ _
→_ ∼ _ D∼ _ _ _</td></tr>
<tr><td>(8-3f)
Excess: +7/11</td><td>B∼C _ ∼A I _ D _ ∼H _ ∼J _</td><td>B∼C|_ ∼A I _ D _ ∼H _ ∼J _
→C∼B _ ∼A I _|D _ ∼H _ ∼J _
→_ I A∼ _ B∼D _ ∼H|_ ∼J _
→H∼ _ D∼B _ ∼A|I _ _ ∼J _
→A∼ _|B∼D _ ∼I _ _ ∼J _
→_ ∼D _ ∼I _ _ ∼J|_
→J∼ _ _|I∼ _ D∼ _ _
→_ _ ∼ _ D∼ _ _</td></tr>
<tr><td>(8-3g)
Excess: +5/11</td><td>B∼C _ H∼A I _ ∼J _ D _</td><td>B∼C|_ H∼A I _ ∼J _ D
→C∼B _ H∼A I _ ∼J _|D
→_ J∼ _ I A∼H|_ B∼D
→H∼A|I _ ∼J _ _ B∼D
→A∼I _ ∼J _ _|B∼D
→_ _ J∼ _ I∼D</td></tr>
<tr><td>(8-3h)
Excess: +5/11</td><td>B∼C _ H∼ _ J∼A I _ D _</td><td>B∼C|_ H∼ _ J∼A I _ D _
→C∼B _ H∼ _ J∼A I _|D _
→_ I A∼J|_ ∼H _ B∼D _
→J∼A|I _ _ ∼H _ B∼D _
→A∼I _ _ ∼H _|B∼D _
→_ H∼ _ _ I∼D _</td></tr>
<tr><td>(8-3i)
Excess: +7/11</td><td>B∼C _ J∼A I _ D _ ∼H _</td><td>B∼C _ J∼A I _ D|_ ∼H _
→D _ I A∼J _|C∼B _ ∼H _
→_ J∼A I _ D∼B _ ∼H|_
→H∼ _ B∼D _|I A∼J _ _
→_ D∼B _ ∼I A∼J|_ _
→J∼A|I∼ _ B∼D _ _ _
→A∼ _|B∼D _ _ _
→_ ∼D _ _ _</td></tr>
</table>

Table 9
(8-8) compound cases

<table>
<tr><th>Case and excess (+) / Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(8-8a)
Excess: +11/11</td><td>B∼C _ K _ H∼ _ ∼A I∼J _ D _</td><td>B∼C|_ K _ H∼ _ ∼A I∼J _ D _
→C∼B _ K _ H∼ _ ∼A I∼J _|D _
→_ J∼I A∼ _ ∼H|_ K _ B∼D _
→H∼ _ ∼A|I∼J _ _ K _ B∼D _
→A∼ _ ∼J _ _ K _|B∼D _
→_ K _ _ J∼ _ ∼D _</td></tr>
<tr><td>(8-8b)
Excess: +11/11</td><td>B∼C _ H∼ _ ∼A I∼J _ D _ K _</td><td>B∼C|_ H∼ _ ∼A I∼J _ D _ K _
→C∼B _ H∼ _ ∼A I∼J _|D _ K _
→_ J∼I A∼ _ ∼H|_ B∼D _ K _
→H∼ _ ∼A|I∼J _ _ B∼D _ K _
→A∼ _ ∼J _ _|B∼D _ K _
→_ _ J∼ _ ∼D _ K _</td></tr>
<tr><td>(8-8c)
Deficiency: −2/11</td><td>B∼C _ K _ ∼A I∼J _ ∼H _ D _</td><td>(Deficiency)</td></tr>
<tr><td>(8-8d)
Excess: 0</td><td>B∼C _ ∼A I∼J _ ∼H _ K _ D _</td><td>B∼C _ ∼A I∼J _ ∼H _ K|_ D _
→K _ H∼ _|J∼I A∼ _ C∼B _ D _
→_ ∼H _ K∼I A∼ |_ C∼B _ D _
→∼A I∼K _ H∼ _ _ C∼B|_ D _
→B∼C _ _ ∼H _ K∼I|A∼ _ D _
→I∼K _|H∼ _ _ C∼ _ D _
→_ K∼ _ _ C∼ _ D _</td></tr>
<tr><td>(8-8e)
Excess: 0</td><td>B∼C _ ∼A I∼J _ ∼H _ D _ K _</td><td>B∼C _ ∼A I∼J _ ∼H _ D _ K|_
→K _ D _ H∼ _|J∼I A∼ _ C∼B _
→_ ∼H _ D _ K∼I A∼ |_ C∼B _
→∼A I∼K _ D _ H∼ _ _ C∼B|_
→B∼C _ _ ∼H _ D _ K∼I|A∼ _
→I∼K _ D _|H∼ _ _ C∼ _
→_ D _ K∼ _ _ C∼ _</td></tr>
<tr><td>(8-8f)
Excess: +11/11</td><td>B∼C _ ∼A I∼J _ D _ ∼H _ K _</td><td>B∼C|_ ∼A I∼J _ D _ ∼H _ K _
→C∼B _ ∼A I∼J _|D _ ∼H _ K _
→_ J∼I A∼ _ B∼D _ ∼H|_ K _
→H∼ _ D∼B _ ∼A|I∼J _ _ K _
→A∼ _|B∼D _ ∼J _ _ K _
→_ ∼D _ ∼J _ _ K _</td></tr>
<tr><td>(8-8g)
Excess: +9/11</td><td>B∼C _ K _ E∼ _ J∼A D _</td><td>B∼C|_ K _ E∼ _ J∼A D _
→C∼B _ K _ E∼ _ J∼A|D _
→A∼J _ ∼E _ K _|B∼D _
→_ K _ E∼ _ J∼D _</td></tr>
<tr><td>(8-8h)
Excess: +9/11</td><td>B∼C _ E∼ _ J∼A D _ K _</td><td>B∼C|_ E∼ _ J∼A D _ K _
→C∼B _ E∼ _ J∼A|D _ K _
→A∼J _ ∼E _|B∼D _ K _
→_ E∼ _ J∼D _ K _</td></tr>
<tr><td>(8-8i)
Excess: +9/11</td><td>B∼C _ J∼A D _ ∼E _ K _</td><td>B∼C|_ J∼A D _ ∼E _ K _
→C∼B _ J∼A|D _ ∼E _ K _
→A∼J _|B∼D _ ∼E _ K _
→_ J∼D _ ∼E _ K _</td></tr>
<tr><td>(8-8j)
Excess: +11/11</td><td>B∼C _ K _ H∼A I∼J _ D _</td><td>B∼C|_ K _ H∼A I∼J _ D _
→C∼B _ K _ H∼A I∼J _|D _
→_ J∼I A∼H|_ K _ B∼D _
→H∼A|I∼J _ _ K _ B∼D _
→A∼J _ _ K _|B∼D _
→_ K _ _ J∼D _</td></tr>
<tr><td>(8-8k)
Excess: +11/11</td><td>B∼C _ H∼A I∼J _ D _ K _</td><td>B∼C|_ H∼A I∼J _ D _ K _
→C∼B _ H∼A I∼J _|D _ K _
→_ J∼I A∼H|_ B∼D _ K _
→H∼A|I∼J _ _ B∼D _ K _
→A∼J _ _|B∼D _ K _
→_ _ J∼D _ K _</td></tr>
</table>

Table 10
(8-9) compound cases

Case and excess (+) / Deficiency (−) of Δ(S)	Description	Flip sequence (not given for deficiencies)
(8-9a) Excess: +11/11	B~C _ H~ _ K _ ~A I~J _ D _	B~C|_ H~ _ K _ ~A I~J _ D _ →C~B _ H~ _ K _ ~A I~J _|D _ →_ J~I A~ _ K _ ~H|_ B~D _ →H~ _ K _ ~A|I~J _ _ B~D _ →A~ _ K _ ~J _ _|B~D _ →_ _ J~ _ K _ ~D _
(8-9b) Excess: +11/11	B~C _ K _ ~A I~J _ D _ ~H _	B~C|_ K _ ~A I~J _ D _ ~H _ →C~B _ K _ ~A I~J _|D _ ~H _ →_ J~I A~ _ K _ B~D _ ~H|_ →H~ _ D~B _ K _ ~A|I~J _ _ →A~ _ K _|B~D _ ~J _ _ →_ K _ ~D _ ~J _ _
(8-9c) Excess: +11/11	B~C _ H~ _ ~A I~J _ K _ D _	B~C|_ H~ _ ~A I~J _ K _ D _ →C~B _ H~ _ ~A I~J _ K _|D _ →_ K _ J~I A~ _ ~H|_ B~D _ →H~ _ ~A|I~J _ K _ _ B~D _ →A~ _ ~J _ K _ _|B~D _ →_ _ K _ J~ _ ~D _
(8-9d) Excess: 0	B~C _ ~A I~J _ K _ ~H _ D _	B~C|_ ~A I~J _ K _ ~H _ D _ →C~B _ ~A I~J _ K|_ ~H _ D _ →K _|J~I A~ _ B~C _ ~H _ D _ →_ K~I A~ _ B~C _ ~H|_ D _ →H~ _ C~B _ ~A|I~K _ _ D _ →A~ _|B~C _ ~K _ _ D _ →_ ~C _ ~K _ _ D _
(8-9e) Excess: +11/11	B~C _ ~A I~J _ K _ D _ ~H _	B~C|_ ~A I~J _ K _ D _ ~H _ →C~B _ ~A I~J _ K _|D _ ~H _ →_ K _ J~I A~ _ B~D _ ~H|_ →H~ _ D~B _ ~A|I~J _ K _ _ →A~ _|B~D _ ~J _ K _ _ →_ ~D _ ~J _ K _ _
(8-9f) Excess: +11/11	B~C _ ~A I~J _ D _ K _ ~H _	B~C|_ ~A I~J _ D _ K _ ~H _ →C~B _ ~A I~J _|D _ K _ ~H _ →_ J~I A~ _ B~D _ K _ ~H|_ →H~ _ K _ D~B _ ~A|I~J _ _ →A~ _|B~D _ K _ ~J _ _ →_ ~D _ K _ ~J _ _
(8-9g) Excess: +9/11	B~C _ E~ _ K _ J~A D _	B~C|_ E~ _ K _ J~A D _ →C~B _ E~ _ K _ J~A|D _ →A~J _ K _ ~E _|B~D _ →_ E~ _ K _ J~D _
(8-9h) Excess: +9/11	B~C _ K _ J~A D _ ~E _	B~C|_ K _ J~A D _ ~E _ →C~B _ K _ J~A|D _ ~E _ →A~J _ K _|B~D _ ~E _ →_ K _ J~D _ ~E _
(8-9i) Excess: +9/11	B~C _ J~A D _ K _ ~E _	B~C|_ J~A D _ K _ ~E _ →C~B _ J~A|D _ K _ ~E _ →A~J _|B~D _ K _ ~E _ →_ J~D _ K _ ~E _
(8-9j) Excess: +11/11	B~C _ H~A I~J _ K _ D _	B~C|_ H~A I~J _ K _ D _ →C~B _ H~A I~J _ K _|D _ →_ K _ J~I A~H|_ B~D _ →H~A|I~J _ K _ _ B~D _ →A~J _ K _ _|B~D _ →_ _ K _ J~D _

of symbols which are consecutive with the endpoint symbols of the initial object. The fourth column gives the length of the flip sequence. The fifth column gives Δ(S), the change in the value of the potential function after completing the prescribed flip sequence. The final column gives the excess (or deficiency) of S, *i.e.* Δ(S) minus the length of the flip sequence.

Table 11
(9-3) compound cases

<table>
<tr><th>Case and excess (+) / Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(9-3a)
Excess: 0</td><td>B∼C _ H∼ _ J∼ _ D I _∼A _</td><td>B∼C _|H∼ _ J∼ _ D I _∼A _
→_ C∼B H∼ _ J∼ _ D I _∼A|_
→A∼_ I D _ ∼J _ ∼H|B∼C _ _
→H∼ _ J∼ _ D|I _∼C _ _
→D _ ∼J _ ∼I|_∼C _ _
→I∼ _|J∼ _ D _∼C _ _
→_ ∼ _ D _∼C _ _</td></tr>
<tr><td>(9-3b)
Excess: +7/11</td><td>B∼C _ J∼ _ D I _ ∼A _ ∼H _</td><td>→B∼C _ J∼ _ D|I _ ∼A _ ∼H _
→D _ ∼J _|C∼B I _ ∼A _ ∼H _
→_ J∼ _ D∼B I _ ∼A _ ∼H|_
→H∼ _ A∼ _|I B∼D _ ∼J _ _
→_ ∼A _ ∼I B∼D _ ∼J|_ _
→J∼ _ D∼B|I∼ _ A∼ _ _ _
→B∼D _ ∼ _|A∼ _ _ _
→_ ∼ _ D∼ _ _ _</td></tr>
<tr><td>(9-3c)
Excess: +7/11</td><td>B∼C _ H∼ _ D I _∼J _ ∼A _</td><td>B∼C _ H∼ _ D I _∼J|_ ∼A _
→J∼_|I D _ ∼H _ C∼B _ ∼A _
→_∼I D _ ∼H|_ C∼B _ ∼A _
→H∼ _ D|I∼_ _ C∼B _ ∼A _
→D _ ∼_ _|C∼B _ ∼A _
→_ _∼ _ D∼B _ ∼A|_
→A∼ _|B∼D _ ∼_ _ _
→_ ∼D _ ∼_ _ _</td></tr>
<tr><td>(9-3d)
Excess: +7/11</td><td>B∼C _ D I _∼H _ ∼J _ ∼A _</td><td>B∼C _ D I _∼H|_ ∼J _ ∼A _
→H∼_|I D _ C∼B _ ∼J _ ∼A _
→_∼I D _ C∼B _ ∼J|_ ∼A _
→J∼ _ B∼C _ D|I∼_ _ ∼A _
→D _|C∼B _ ∼_ _ ∼A _
→_ D∼B _ ∼_ _ ∼A|_
→A∼ _ _∼ _|B∼D _ _
→_ ∼_ _ ∼D _ _</td></tr>
<tr><td>(9-3e)
Excess: +7/11</td><td>B∼C _ D I _ ∼J _ ∼A _∼H _</td><td>B∼C _ D I _ ∼J|_ ∼A _∼H _
→J∼ _|I D _ C∼B _ ∼A _∼H _
→_ ∼G D _ C∼B _ ∼A _∼H|_
→H∼_ A∼ _ B∼C _ D|G∼ _ _
→D _|C∼B _ ∼A _∼ _ _
→_ D∼B _ ∼A|_∼ _ _
→A∼ _|B∼D _ _∼ _ _
→_ ∼D _ _∼ _ _</td></tr>
<tr><td>(9-3f)
Excess: +7/11</td><td>B∼C _ D I _ ∼A _ ∼H _ ∼J _</td><td>B∼C _ D|I _ ∼A _ ∼H _ ∼J _
→D _|C∼B I _ ∼A _ ∼H _ ∼J _
→_ D∼B I _ ∼A _ ∼H|_ ∼J _
→H∼ _ A∼ _|I B∼D _ _ ∼J _
→_ ∼A _ ∼I B∼D _ _ ∼J|_
→J∼ _ _ D∼B|I∼ _ A∼ _ _
→B∼D _ _ ∼ _|A∼ _ _
→_ ∼ _ _ D∼ _ _</td></tr>
</table>

Let us illustrate the information provided in the tables. For example, in Case (1), shown in Table 1, we have a permutation π of the form B _ A _, where B and A denote consecutive singletons. By a sequence S consisting of a single prefix reversal we make an adjacency by putting B next to A, thereby creating the permutation π' of the form _ BA _. This eliminates the two singletons B and A, and creates a new block, namely BA. Therefore, the change in our potential function, going from π to π', is a decrease in $s(\pi)$ of 2 and an increase in $b(\pi)$ of 1. That is, if $\Phi(\pi) = (18/11)s(\pi) + (24/11)b(\pi)$, then $\Phi(\pi') = (18/11)(s(\pi) - 2) + (24/11)(b(\pi) + 1)$. So, $\Delta(S) = \Phi(\pi) - \Phi(\pi') = 2(18/11) - (24/11) = 12/11$. As 12/11 is greater than 1, the length of the sequence S, the flip sequence S is good and, in fact, has an excess of $(12/11) - 1 = 1/11$.

In Case (3) shown in Table 1, the flip sequence S has length 4 and makes the initial singleton adjacent to its two consecutive elements that are endpoints of blocks. Thus, it reduces the number of singletons by one and reduces the number of blocks by one. That is, $\Delta(S) = \Phi(\pi) - \Phi(\pi')$, where π is the initial permutation and π' is the resulting permutation after the sequence S, is $(18/11) + (24/11) = 42/11$. As this is less than 4, the flip sequence S is not good, and the deficiency is $(42/11) - 4 = -2/11$.

Table 12
(9-8) compound cases

<table>
<tr><th>Case and excess (+) / Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(9-8a)
Excess: +7/11</td><td>B~C _ K _ H~ _ D I~J _ ~A _</td><td>B~C _ K _ H~ _ D I~J|_ ~A _
→J~I D _ ~H _|K _ C~B _ ~A _
→_ H~ _ D I~K _ C~B _ ~A|_
→A~ _|B~C _ K~I D _ ~H _ _
→_ ~C _ K~I D _ ~H|_ _
→H~ _ D|I~K _ C~ _ _ _
→D _ ~K _|C~ _ _ _
→_ K~ _ D~ _ _ _</td></tr>
<tr><td>(9-8b)
Excess: +7/11</td><td>B~C _ H~ _ D I~J _~A _ K _</td><td>B~C _ H~ _ D|I~J _~A _ K _
→D _ ~H _|C~B I~J _~A _ K _
→_ H~ _ D~B I~J _~A _ K|_
→K _ A~_ |J~I B~D _ ~H _ _
→_~A _ K~I B~D _ ~H|_ _
→H~ _ D~B|I~K _ A~_ _ _
→B~D _ ~K _|A~_ _ _
→_ K~ _ D~_ _</td></tr>
<tr><td>(9-8c)
Excess: 0</td><td>B~C _ K _ D I~J _ ~H _ ~A _</td><td>B~C _ K _ D I~J|_ ~H _ ~A _
→J~I D _|K _ C~B _ ~H _ ~A _
→_ D I~K _ C~B _ ~H _ ~A|_
→A~ _ H~ _|B~C _ K~I D _ _
→_ ~H _ ~C _ K~I|D _ _
→I~K _ C~ _|H~ _ D _ _
→_ ~C _ K~ _ D _ _</td></tr>
<tr><td>(9-8d)
Excess: 0</td><td>B~C _ D I~J _ ~H _ K _ ~A _</td><td>B~C _ D I~J|_ ~H _ K _ ~A _
→J~I D _ C~B _ ~H|_ K _ ~A _
→H~ _ B~C _ D|I~J _ K _ ~A _
→D _|C~B _ ~J _ K _ ~A _
→_ D~B _ ~J _ K _ ~A|_
→A~ _ K _ J~ _|B~D _ _
→_ ~J _ K _ ~D _ _</td></tr>
<tr><td>(9-8e)
Excess: 0</td><td>B~C _ D I~J _ ~H _ ~A _ K _</td><td>B~C _ D I~J|_ ~H _ ~A _ K _
→J~I D _ C~B _ ~H|_ ~A _ K _
→H~ _ B~C _ D|I~J _ ~A _ K _
→D _|C~B _ ~J _ ~A _ K _
→_ D~B _ ~J _ ~A|_ K _
→A~ _ J~ _|B~D _ _ K _
→_ ~J _ ~D _ _ K _</td></tr>
<tr><td>(9-8f)
Excess: +7/11</td><td>B~C _ D I~J _ ~A _ ~H _ K _</td><td>B~C _ D|I~J _ ~A _ ~H _ K _
→D _|C~B I~J _ ~A _ ~H _ K _
→_ D~B I~J _ ~A _ ~H _ K|_
→K _ H~ _ A~ _|J~I B~D _ _
→_ ~A|_ ~H _ K~I B~D _ _
→A~ _ _ ~H _ K~I|B~D _ _
→I~K _|H~ _ _ ~D _ _
→_ K~ _ _ ~D _ _</td></tr>
<tr><td>(9-8g)
Excess: +5/11</td><td>B~C _ E _ H~ _ D I~A _</td><td>B~C|_ E _ H~ _ D I~A _
→C~B _ E _ H~ _|D I~A _
→_ ~H|_ E _ B~D I~A _
→H~ _ _ E _ B~D|I~A _
→D~B _|E _ _ ~A _
→_ B~E _ _ ~A _</td></tr>
<tr><td>(9-8h)
Excess: 0</td><td>B~C _ H~ _ D I~A _ E _</td><td>B~C|_ H~ _ D I~A _ E _
→C~B _ H~ _|D I~A _ E _
→_ ~H|_ B~D I~A _ E _
→H~ _ _ B~D|I~A _ E _
→D~B _ _ ~A|_ E _
→A~ _ _|B~D _ E _
→_ _ ~D _ E _</td></tr>
<tr><td>(9-8i)
Excess: 0</td><td>B~C _ D I~A _ ~H _ E _</td><td>B~C _ D I~A|_ ~H _ E _
→A~I D _ C~B _ ~H|_ E _
→H~ _ B~C _ D|I~A _ E _
→D _|C~B _ ~A _ E _
→_ D~B _ ~A|_ E _
→A~ _|B~D _ _ E _
→_ ~D _ _ E _</td></tr>
</table>

Table 13
(9-9) compound cases

<table>
<tr><th>Case and excess (+) / Deficiency (−) of Δ(S)</th><th>Description</th><th>Flip sequence
(not given for deficiencies)</th></tr>
<tr><td>(9-9a)
Excess: +4/11</td><td>B∼C _ H∼ _ K _ D I∼J _∼A _</td><td>B∼C _|H∼ _ K _ D I∼J _∼A _
→_ C∼B H∼ _ K _ D I∼J _∼A|_
→A∼_ J∼I D _ K _ ∼H|B∼C _ _
→H∼ _ K _ D|I∼J _∼C _ _
→D _ K _ ∼J _∼C _ _</td></tr>
<tr><td>(9-9b)
Excess: +7/11</td><td>B∼C _ K _ D I∼J _ ∼A _ ∼H _</td><td>B∼C _ K _ D I∼J|_ ∼A _ ∼H _
→J∼I D _|K _ C∼B _ ∼A _ ∼H _
→_ D I∼K _ C∼B _ ∼A|_ ∼H _
→A∼ _|B∼C _ K∼I D _ _ ∼H _
→_ ∼C _ K∼I D _ _ ∼H|_
→H∼ _ _ D|I∼K _ C∼ _ _
→D _ _ ∼K _|C∼ _ _
→_ K∼ _ _ D∼ _ _</td></tr>
<tr><td>(9-9c)
Excess: +7/11</td><td>B∼C _ H∼ _ D I∼J _ K _ ∼A _</td><td>B∼C _ H∼ _ D I∼J _ K|_ ∼A _
→I _|J∼I D _ ∼H _ C∼B _ ∼A _
→_ K∼I D _ ∼H|_ C∼B _ ∼A _
→H∼ _ D|I∼K _ _ C∼B _ ∼A _
→D _ ∼K _ _|C∼B _ ∼A _
→_ _ K∼ _ D∼B _ ∼A|_
→A∼ _|B∼D _ ∼K _ _ _
→_ ∼D _ ∼K _ _ _</td></tr>
<tr><td>(9-9d)
Excess: +7/11</td><td>B∼C _ D I∼J _ K _ ∼H _ ∼A _</td><td>B∼C _ D I∼J _ K|_ ∼H _ ∼A _
→K _|J∼I D _ C∼B _ ∼H _ ∼A _
→_ K∼I D _ C∼B _ ∼H|_ ∼A _
→H∼ _ B∼C _ D|I∼K _ _ ∼A _
→D _|C∼B _ ∼K _ _ ∼A _
→_ D∼B _ ∼K _ _ ∼A|_
→A∼ _ _ K∼ _|B∼D _ _
→_ ∼K _ _ ∼D _ _</td></tr>
<tr><td>(9-9e)
Excess: +7/11</td><td>B∼C _ D I∼J _ K _ ∼A _ ∼H _</td><td>B∼C _ D I∼J _ K|_ ∼A _ ∼H _
→K _|J∼I D _ C∼B _ ∼A _ ∼H _
→_ K∼I D _ C∼B _ ∼A _ ∼H|_
→H∼ _ A∼ _ B∼C _ D|I∼K _ _
→D _|C∼B _ ∼A _ ∼K _ _
→_ D∼B _ ∼A|_ ∼K _ _
→A∼ _|B∼D _ _ ∼K _ _
→_ ∼D _ _ ∼K _ _</td></tr>
<tr><td>(9-9f)
Excess: +7/11</td><td>B∼C _ D I∼J _ ∼A _ K _ ∼H _</td><td>B∼C _ D|I∼J _ ∼A _ K _ ∼H _
→D _|C∼B I∼J _ ∼A _ K _ ∼H _
→_ D∼B I∼J _ ∼A _ K|_ ∼H _
→K _ A∼ _|J∼I B∼D _ _ ∼H _
→_ ∼A _ K∼I B∼D _ _ ∼H|_
→H∼ _ _ D∼B|I∼K _ A∼ _ _
→B∼D _ _ ∼K _|A∼ _ _
→_ K∼ _ _ D∼ _ _</td></tr>
<tr><td>(9-9g)
Excess: +5/11</td><td>B∼C _ H∼ _ E _ D I∼A _</td><td>B∼C|_ H∼ _ E _ D I∼A _
→C∼B _ H∼ _ E _|D I∼A _
→_ E _ ∼H|_ B∼D I∼A _
→H∼ _ E _ _ B∼D|I∼A _
→D∼B _ _|E _ ∼A _
→_ _ B∼E _ ∼A _</td></tr>
<tr><td>(9-9h)
Excess: 0</td><td>B∼C _ E _ D I∼A _ ∼H _</td><td>B∼C _ E _ D I∼A|_ ∼H _
→A∼I D _ E _ C∼B _ ∼H|_
→H∼ _ B∼C _ E _ D|I∼A _
→D _ E _|C∼B _ ∼A _
→_ E _ D∼B _ ∼A|_
→A∼ _|B∼D _ E _ _
→_ ∼D _ E _ _</td></tr>
<tr><td>(9-9i)
Excess: 0</td><td>B∼C _ D I∼A _ E _ ∼H _</td><td>B∼C _ D I∼A|_ E _ ∼H _
→A∼I D _ C∼B _ E _ ∼H|_
→H∼ _ E _ B∼C _ D|I∼A _
→D _|C∼B _ E _ ∼A _
→_ D∼B _ E _ ∼A|_
→A∼ _ E _|B∼D _ _
→_ E _ ∼D _ _</td></tr>
</table>

Table 14
All arrangements represented by the generating sequence B S* _ H _ ~A I _ ~C _ T*

Case	Initial permutation	Case	Initial permutation
(3-1a-a)	B S _ T _ H _ ~A I _ ~C _	(3-1a-p)	B S~ _ T~ _ H _ ~A I _ ~C _
(3-1a-b)	B S _ T~ _ H _ ~A I _ ~C _	(3-1a-q)	B S~ _ ~T _ H _ ~A I _ ~C _
(3-1a-c)	B S _ ~T _ H _ ~A I _ ~C _	(3-1a-r)	B S~ _ H _ T _ ~A I _ ~C _
(3-1a-d)	B S _ H _ T _ ~A I _ ~C _	(3-1a-s)	B S~ _ H _ T~ _ ~A I _ ~C _
(3-1a-e)	B S _ H _ T~ _ ~A I _ ~C _	(3-1a-t)	B S~ _ H _ ~T _ ~A I _ ~C _
(3-1a-f)	B S _ H _ ~T _ ~A I _ ~C _	(3-1a-u)	B S~ _ H _ T~A I _ ~C _
(3-1a-g)	B S _ H _ T~A I _ ~C _	(3-1a-v)	B S~ _ H _ ~A I _ T _ ~C _
(3-1a-h)	B S _ H _ ~A I _ T _ ~C _	(3-1a-w)	B S~ _ H _ ~A I _ T~ _ ~C _
(3-1a-i)	B S _ H _ ~A I _ T~ _ ~C _	(3-1a-x)	B S~ _ H _ ~A I _ ~T _ ~C _
(3-1a-j)	B S _ H _ ~A I _ ~T _ ~C _	(3-1a-y)	B S~ _ H _ ~A I _ T~C _
(3-1a-k)	B S _ H _ ~A I _ T~C _	(3-1a-z)	B S~ _ H _ ~A I _ ~C _ T _
(3-1a-l)	B S _ H _ ~A I _ ~C _ T _	(3-1a-aa)	B S~ _ H _ ~A I _ ~C _ T~ _
(3-1a-m)	B S _ H _ ~A I _ ~C _ T~ _	(3-1a-ab)	B S~ _ H _ ~A I _ ~C _ ~T _
(3-1a-n)	B S _ H _ ~A I _ ~C _ ~T _	(3-1a-ac)	B H _ ~A I _ ~C _
(3-1a-o)	B S~ _ T _ H _ ~A I _ ~C _		

3. Compound cases

As Table 1 indicates, the sequences of prefix reversals given by Gates and Papadimitriou [3] are good for our potential function Φ in all cases except for cases (3), (8), and (9). Specifically, for each of the sequences S in cases (3), (8), and (9) there are four prefix reversals, but since $\Delta(S) = 42/11 < 4$, resulting in a deficiency of $-2/11$, these cases are bad for Φ. So, to prove that the potential function $\Phi(\pi)$ is an upper bound for the number of steps to sort π, it is sufficient to find good sequences for cases (3), (8), and (9) or expanded replacements.

We resolve cases (3), (8), and (9) by considering compound cases and alternate flip sequences for them. Using arithmetic based on the $\Delta(S)$ values of Table 1, we see that the potential function $\Phi(\pi)$ and the Gates and Papadimitriou sequences of prefix reversals works for all compound cases, except (3-y), (8-y), and (9-y), where y∈{1, 3, 8, 9}. That is, if one adds $\Delta(S)$ for any sequence S given in Table 1 to the $-2/11$ deficiency given for cases (3), (8), and (9), one obtains a positive number, except when the second case in a compound case is either (1), (3), (8), or (9). Therefore, one needs to exhibit good sequences for these compound cases.

Tables 2 and 5–7 show all sub-cases of (3-y) compound cases, Tables 3 and 8–10 show all sub-cases of (8-y) compound cases, and Tables 4 and 11–13 show all sub-cases of (9-y) compound cases. These tables show good sequences for the sub-cases, when they exist, and also indicate when good sequences do not exist. The sub-cases for which good sequences do not exist will be handled by a breadth expansion in Section 4.

Note that each compound case (x-y) actually describes a *set* of sub-cases as shown in Tables 2–13. The reason is that in a permutation, the blocks and singletons of the y portion of the case may be positioned in several different spots with respect to the components of x portion of the case. To illustrate, consider the compound (3-9) case. Let the letters X, Y, and Z denote arbitrary sub-lists. Using this notation, the initial permutation for case (3) can be represented by the string w = B X ~A Y ~C Z. Application of the sequence of prefix reversals S for case (3) given in Table 1 yields the string w' = Y ~C B A~ X^R Z, where X^R denotes the reversal of the sub-list X.

This means two things: (a) that the object that appears at the beginning of the permutation represented by the string w' is the object which is at the beginning of the sub-list Y and, therefore, is immediately after the block ~A in the original permutation and (b) the sub-list X is reversed after the completion of the flip sequence S and, hence, any blocks within X will have reversed orientation. This tells us where to place the singleton and blocks of the subsequent case (9) of the compound case (3-9). That is, one must consider all of the sub-cases shown in Table 7. Sub-cases (3-9a) through (3-9f) are straightforward inter-leavings of the case (9) components into the case (3) portion. Sub-cases (3-9g) through (3-9j) have to be included as well, but the reason might not be so obvious. To see why these latter cases are necessary, consider, for example, sub-case (3-9j). This sub-case takes care of the situation in which the unspecified endpoint at the other end of the ~H block for the case (9) portion of the permutation is also the endpoint of the ~A block for the case (3) portion of the permutation. Sub-cases (3-9g), (3-9h), (3-9i) take care of similar situations where some of the case (9) components are merged with some of the case (3) components. Each of Tables 2–13 contain several such cases as needed for a complete enumeration of all arrangements.

In the tables, the vertical bar (|) in each permutation in the flip sequence column identifies the prefix which will be reversed to arrive at the next permutation in the sequence.

Tables 2–13 indicate good flip sequences for several compound cases, but for the cases marked as deficiencies, no good sequences exist. In particular, sub-cases (3-1a, b, c), (3-3a, h), (8-1b, c), (8-8c), and (9-1a, b, c) need further expansion. Expansion by looking deeper, *i.e.* additional compounding, such as (3-3-1), (3-3-3), (3-3-8), and (3-3-9), results in a very large number of sub-cases. Consequently, we choose to resolve these cases by expanding by breadth, as described in the next section.

4. Expanding cases by breadth

As mentioned in the introduction, the richer structure revealed by compound cases allows us to find good flip sequences for many compound cases. However, further structure is needed to resolve the remaining cases and to arrive at a complete proof. We now give a detailed analysis of the expansion of cases by breadth. We start with an example of another compound case.

Consider sub-case (3-1a), denoted by B _ H _ ∼A I _ ∼C _, described in Table 2, and for which there is no good sequence. We expand it further by considering an element, either a singleton S or a block S∼, which occurs just after the initial singleton B, and all positions where an element consecutive to S, either a singleton T, a block beginning with T, or a block ending with T, may be located. This set of sub-cases, denoted by the generating string B S* _ H _ ∼A I _ ∼C _ T*, is given in Table 14. We also need to include the case when the element immediately after the initial singleton B is a singleton H from the case (1) portion of the compound case (3-1), rather than some new element S. This specific sub-case is included as the last sub-case (3-1a-ac) in Table 14, namely B H _ ∼A I _ ∼C _.

There are good sequences for all but four of the expanded sub-cases of (3-1a) shown in Table 14. The exceptions are (3-1a-c), (3-1a-j), (3-1a-n), and (3-1a-ac). These four sub-cases, where no good sequences were found, are further expanded, as indicated in Table 15 in the Appendix. For example, sub-case (3-1a-c), namely B S _ ∼T _ H _ ∼A I _ ∼C, is resolved by considering permutations denoted by the generating string B S _ ∼T _ H _ ∼A I _ ∼C _ R*. Here, the R* represents an element consecutive to the singleton S, which may be a singleton R, or a block beginning with R or a block ending with R, placed in all possible positions within the permutation. This generates a set of sub-cases, denoted in Table 15 in the Appendix by (3-1a-c) a-r. Every one of these sub-cases has a good sequence, so no further expansion is necessary.

The example of the expansion by breadth of the compound case (3-1a-c) is indicative of the manner in which all other compound cases are resolved. That is, we expand by considering additional elements either as part of an existing block or as a singleton next to an existing element and consider all possible positions where consecutive elements may be placed. In general, we create generating strings in three ways:

(a) by placing a singleton, say S, in some position and adding a consecutive letter, either R or T, with an asterisk, at the end of the string;
(b) by placing a singleton with an asterisk, say S*, in some position and adding a consecutive letter, either R or T, with an asterisk, at the end of the string;
(c) by placing a singleton with an asterisk, say *S, in some position and adding a consecutive letter, either R or T, with an asterisk, at the end of the string.

In (a) the singleton S is fixed in the chosen position, while the consecutive symbol T is moved to every possible position either as a singleton, or as a right or left endpoint of a block. In (b) the symbol S* is replaced in the indicated position first with a singleton and second with a block in which S is the left endpoint, while the consecutive symbol T is moved to every possible position either as a singleton, or as a right or left endpoint of a block. Similarly, in (c) the symbol S* is replaced in the indicated position first with a singleton and second with a block in which S is the right endpoint, while the consecutive symbol T is moved to every possible position either as a singleton, or as a right or left endpoint of a block.

To illustrate the creation of generating strings, refer again to Table 14. The situations where S is a singleton are shown in Table 14 in sub-cases (3-1a-a) through (3-1a-n). The situations where S is the left endpoint of a block are shown in Table 14 in sub-cases (3-1a-o) through (3-1a-ab). This includes the cases, shown in Table 14 as sub-cases (3-1a-g), (3-1a-k), (3-1a-u), (3-1a-y), where T is added as an endpoint of an existing block. It also includes the sub-case (3-1a-ac), where the element immediately following the initial element is not new, but is an existing element (in this case it is a singleton from case (1) of the compound (3-1a) case).

Due to the fact that there are 2220 sub-cases each requiring flip sequences, we do not include an exhaustive list of sub-cases and good sequences. Instead, in Tables 15–17 in the Appendix, we list all generating strings used to form the sub-cases. These give the structure of our solution. A reader can verify its correctness or go to [8] where a file contains all cases and good flip sequences.

Tables 15–17 (in the Appendix) can be viewed as trees. Sub-cases that need further expansion are followed in the table by an indented list of further generating strings for additional sub-cases. In these tables, every sub-case that is not followed by an indented list represents a case where every sub-case has a good sequence.

We used programs to automate the process of verifying the large number of cases and the search for good sequences. For example, the Java program we used is available for download at our website [8]. The input to the program is a generating string of the form given in Table 15 in the Appendix. Given such a string, the program systematically generates each implied permutation and searches for a good sequence for it. The website has a *readme* file with instructions for using the program. The program tries all possible flip sequences up to a user specified maximum sequence length, with branch-and-bound pruning of the search tree. That is, it deletes from consideration sequences that can not possibly create enough new adjacencies to satisfy the potential function within the remaining number of steps. For each permutation generated by the generating string, the program output consists of a good flip sequence S (if one is found), the value of $\Delta(S)$, the length of the flip sequence and the excess or deficiency of $\Delta(S)$. The complete solution [8] is arranged in the order of cases as indicated in Tables 15–17 in the Appendix.

5. Conclusion

We have improved the upper bound for the pancake problem to $(18/11)n \approx (1.6363)n$. Our proof required a subdivision of the nine basic cases defined in [3] into several additional cases. We conjecture that a further improvement on the upper bound is possible by further subdivisions and several additional cases. This could be done, for example, by defining a different potential function, with different constants as multipliers of the number of singletons and blocks, and then developing cases with good flip sequences to justify it. Of course, the number of cases may be much larger than what we have used. Evidence that the actual upper bound may be lower is suggested by the computed values for the worst case number of flips for all permutations of length 14, 15, 16, etc. [5,6]. The computed numbers are 16, 17, and 18 steps for permutations of length 14, 15, and 16, respectively, which is smaller than the current asymptotic upper bound values. Although we believe that the upper bound can be further improved, we currently only see possibilities of a proof based on further case subdivision or proving a conjecture [2] for which we, unfortunately, currently have no special insight.

The lower bound is currently $(15/14)n$ [4]. This was proved by a small change in the technique used in [3] to prove a $(17/16)n$ lower bound. Specifically, consider the permutation $P(0) = 1, 7, 5, 3, 6, 4, 2$ and, for all i $(i \geq 1)$, the permutations $P(\text{i}) = 1 + 7\text{i}, 7 + 7\text{i}, 5 + 7\text{i}, 3 + 7\text{i}, 6 + 7\text{i}, 4 + 7\text{i}, 2 + 7\text{i}$. Define a permutation π_n on the symbols in $\{1, 2, \ldots, 7n\}$ by $P(0), P(1), P(2), \ldots, P(n-1)$. A *waste* is a move that does not create a new adjacency. The lower bound argument is that there must be at least one waste for every *two* of the P(i)'s. We conjecture that a larger number of wastes must occur, perhaps even nearly as much as one waste for every P(i), and that the lower bound can be improved by proving the conjecture.

In this paper, we have restricted our discussion to the problem of sorting *unsigned* permutations. The problem of sorting signed permutations (or "burnt pancakes") has also been studied [2–4]. Cohen and Blum [2] showed a $(23/14)n \approx (1.6429)n$ upper bound for the signed permutation $-I_n = -1, -2, \ldots, -n$, which they conjectured to be the hardest signed permutation to sort using prefix reversals. The upper bound for $-I_n$ was improved to $3(n+1)/2 \approx (1.5)n + O(1)$ by Heydari and Sudborough [4]. Since (unsigned) permutations are no harder to sort than signed permutations, it follows that if $-I_n$ were indeed the hardest signed permutation, then $3(n+1)/2$ would also be an upper bound for sorting all permutations of length n. It is also known that $3n/2$ prefix reversals are necessary to sort $-I_n$ [2,3]. The current upper bound for sorting signed permutations by prefix reversals [2] is $2n - 3$.

The authors would like to thank the referees for their helpful comments.

Appendix. Tables 15–17

Table 15
Generating strings for the solution of (3-y) expanded cases, for y∈{1, 3}

(3-y) Expanded cases for y∈{1, 3}	Generating string
(3-1a) a-ac	B S* _ H _ ~A I _ ~C _ T*
(3-1a-c) a-r	B S _ ~T _ H _ ~A I _ ~C _ R*
(3-1a-j) a-r	B S _ H _ ~A I _ ~T _ ~C _ R*
(3-1a-n) a-r	B S _ H _ ~A I _ ~C _ ~T _ R*
(3-1a-ac) a-k	B H _ ~A I _ ~C _ G*
(3-1b) a-ac	B S* _ ~A I _ H _ ~C _ T*
(3-1b-c) a-r	B S _ ~T _ ~A I _ H _ ~C _ R*
(3-1b-g) a-r	B S _ ~A I _ ~T _ H _ ~C _ R*
(3-1b-n) a-r	B S _ ~A I _ H _ ~C _ ~T _ R*
(3-1b-ac) a-j	B Y~A I _ H _ ~C _ X*
(3-1c) a-ac	B _ ~A I _ ~C _ S H _ T
(3-1c-c) a-r	B _ ~T _ ~A I _ ~C _ S H _ G*
(3-1c-c-i) a-u	B _ ~T _ ~A I _ G _ ~C _ S H _ R*
(3-1c-c-k) a-v	B _ ~T _ ~A I _ ~G _ ~C _ S H _ J*
(3-1c-c-p) a-u	B _ ~T _ ~A I _ ~C _ S H _ G _ R*
(3-1c-f) a-r	B _ ~A I _ T~ _ ~C _ S H _ G*
(3-1c-f-a) a-u	B _ G _ ~A I _ T~ _ ~C _ S H _ R*
(3-1c-f-e) a-u	B _ ~A I _ G _ T~ _ ~C _ S H _ R*
(3-1c-f-g) a-u	B _ ~A I _ ~G _ T~ _ ~C _ S H _ J*
(3-1c-f-p) a-u	B _ ~A I _ T~ _ ~C _ S H _ G _ R*
(3-1c-h) a-m	B _ ~A I _ T~C _ S H _ R*
(3-1c-h-c) a-q	B _ ~R _ ~A I _ T~C _ S H _ J*
(3-1c-h-f) a-q	B _ ~A I _ R~ _ T~C _ S H _ G*
(3-1c-h-f-g) a-u	B _ ~A I _ ~G _ R~ _ T~C _ S H _ J*
(3-1c-h-i) a-q	B _ ~A I _ T~C _ R~ _ S H _ G*
(3-1c-j) a-r	B _ ~A I _ ~C _ T~ _ S H _ J*

Table 15
(*continued*)

(3-y) Expanded cases for y∈{1, 3}	Generating string
(3-1c-j-a)	B _ J _ ∼A I _ ∼C _ T∼ _ S H _ (this is a (3-1a) case)
(3-1c-j-e)	B _ ∼A I _ J _ ∼C _ T∼ _ S H _ (this is a (3-1b) case)
(3-1c-j-i) a-t	B _ ∼A I _ ∼C _ J _ T∼ _ S H _ R*
(3-1c-j-i-c)	B _ M∼R _ ∼A I _ ∼C _ J _ T∼ _ S H _ N*
(3-1c-j-i-f)	B _ ∼A I _ R∼ _ ∼C _ J _ T∼ _ S H _ G*
(3-1c-j-i-h)	B _ M∼A I _ R∼C _ J _ T∼ _ S H _ N*
(3-1c-j-i-m)	B _ ∼A I _ ∼C _ J _ R∼ _ T∼ _ S H _ G*
(3-1c-j-p) a-u	B _ ∼A I _ ∼C _ T∼ _ S H _ J _ R*
(3-1c-n) a-r	B _ ∼A I _ ∼C _ S H _ ∼T _ R*
(3-1c-x) a-s	B _ ∼A I _ ∼C _ T∼ _ ∼S H _ J*
(3-1c-x-a)	B _ J _ ∼A I _ ∼C _ T∼ _ ∼S H _ (this is a (3-1a) case)
(3-1c-x-e)	B _ ∼A I _ J _ ∼C _ T∼ _ ∼S H _ (this is a (3-1b) case)
(3-1c-x-i) a-u	B _ ∼A I _ ∼C _ J _ T∼ _ R∼S H _ Q*
(3-1c-x-i-e)	B _ ∼A I _ Q _ ∼C _ J _ T∼ _ R∼S H _ G*
(3-1c-x-i-i)	B _ ∼A I _ ∼C _ Q _ J _ T∼ _ R∼S H _ G*
(3-1c-x-i-p)	B _ ∼A I _ ∼C _ J _ T∼ _ Q _ R∼S H _ G*
(3-1c-x-i-s)	B _ ∼A I _ ∼C _ J _ T∼ _ R∼S H _ Q _ G*
(3-1c-x-q) a-u	B _ ∼A I _ ∼C _ T∼ _ R∼S H _ J _ Q*
(3-1c-x-q-i)	B _ ∼A I _ ∼C _ Q _ T∼ _ R∼S H _ J _ G*
(3-1c-x-q-s)	B _ ∼A I _ ∼C _ T∼ _ R∼S H _ J _ Q _ G*
(3-1c-ac) a-w	B X* _ ∼A I _ ∼C H _ Y*
(3-1c-ac-c)	B X _ ∼Y _ ∼A I _ ∼C H _ W*
(3-1c-ac-w)	B U∼A I _ ∼C H _ V*
(3-3a) a-r	B _ H∼ _ J∼ _ ∼A I _ S∼C _ T*
(3-3a-r) a-u	B _ H∼ _ J∼ _ U∼A I _ S∼C _ ∼T _ V*
(3-3h) a-m	B _ H∼ _ J∼A I _ S∼C _ T*
(3-3h-m) a-p	B _ H∼ _ J∼A I _ S∼C _ V∼T _ W*

Table 16
Generating strings for the solution of (8-y) expanded cases, for y∈{1, 8}

(8-y) Expanded cases for y∈{1, 8}	Generating string
(8-1b) a-aa	B∼C P* _ ∼A I _ H _ D _ Q*
(8-1b-c) a-q	B∼C P _ ∼Q _ ∼A I _ H _ D _ O*
(8-1b-e) a-ag	B∼C P _ ∼A I _ Q _ *T H _ D _ S*
(8-1b-e-f) a-t	B∼C P _ ∼A I _ S∼ _ Q _ T H _ D _ U*
(8-1b-e-i) a-t	B∼C P _ ∼A I _ Q _ S∼ _ T H _ D _ U*
(8-1b-e-i-i) a-x	B∼C P _ ∼A I _ Q _ U∼ _ S∼ _ T H _ D _ J*
(8-1b-e-i-i-u)	B∼C P _ ∼A I _ Q _ U∼ _ S∼ _ T H _ ∼J _ D _ O*
(8-1b-e-i-m)	B∼C P _ ∼A I _ Q _ S∼ _ U∼ _ T H _ D _
(8-1b-g) a-q	B∼C P _ ∼A I _ ∼Q _ H _ D _ O*
(8-1b-h) a-af	B∼C P _ ∼A I _ *T H _ Q _ D _ S*
(8-1b-h-b) a-t	B∼C P _ S∼ _ ∼A I _ T H _ Q _ D _ U*
(8-1b-h-d) a-o	B∼C P _ S∼A I _ T H _ Q _ D _ U*
(8-1b-h-f) a-t	B∼C P _ ∼A I _ S∼ _ T H _ Q _ D _ U*
(8-1b-h-f-n) a-x	B∼C P _ ∼A I _ S∼ _ T H _ ∼U _ Q _ D _ J*
(8-1b-h-f-n-e)	B∼C P _ X∼A I _ J _ S∼ _ T H _ ∼U _ Q _ D _ W*
(8-1b-h-j) a-t	B∼C P _ ∼A I _ T H _ ∼S _ Q _ D _ U*
(8-1b-h-j-f) a-x	B∼C P _ ∼A I _ U∼ _ T H _ ∼S _ Q _ D _ J*
(8-1b-h-j-f-e)	B∼C P _ X∼A I _ J _ U∼ _ T H _ ∼S _ Q _ D _ W*
(8-1b-h-j-j) a-x	B∼C P _ ∼A I _ T H _ ∼U _ ∼S _ Q _ D _ J*
(8-1b-h-j-j-e)	B∼C P _ X∼A I _ J _ T H _ ∼U _ ∼S _ Q _ D _ W*
(8-1b-h-j-j-s)	B∼C P _ X∼A I _ T H _ ∼U _ ∼S _ Q _ J _ D _ W*
(8-1b-h-j-n)	B∼C P _ ∼A I _ T H _ ∼S _ ∼U _ Q _ D _ (this is (8-1b-h-j-j))
(8-1b-h-m) a-t	B∼C P _ ∼A I _ T H _ Q _ ∼S _ D _ U*
(8-1b-h-p) a-t	B∼C P _ ∼A I _ T H _ Q _ D _ ∼S _ U*
(8-1b-j) a-q	B∼C P _ ∼A I _ H _ ∼Q _ D _ O*
(8-1b-m) a-q	B∼C P _ ∼A I _ H _ D _ ∼Q _ O*
(8-1b-aa) a-I	B∼C U∼A I _ H _ D _ V*
(8-1c) a-aa	B∼C P* _ ∼A I _ D _ H _ Q*
(8-1c-c) a-q	B∼C P _ ∼Q _ ∼A I _ D _ H _ O*
(8-1c-g) a-q	B∼C P _ ∼A I _ ∼Q _ D _ H _ O*
(8-1c-j) a-q	B∼C P _ ∼A I _ D _ ∼Q _ H _ O*
(8-1c-m) a-q	B∼C P _ ∼A I _ D _ H _ ∼Q _ O*
(8-1c-aa) a-I	B∼C P∼A I _ D _ H _ O*
(8-8c) a-p	B∼C _ K _ S∼A I∼J _ ∼H _ D _

Table 17
Generating strings for the solution of (9-1) expanded cases

(9-1) Expanded cases	Generating string
(9-1a) a-aa	B~C _ H S* _ D G _ ~A _ T*
(9-1a-b) a-q	B~C _ T~ _ H S _ D G _ ~A _ R*
(9-1a-f) a-q	B~C _ H S _ ~T _ D G _ ~A _ R*
(9-1a-h) a-q	B~C _ H S _ D G _ T~ _ ~A _ R*
(9-1a-j) a-l	B~C _ H S _ D G _ T~A _ R*
(9-1a-m) a-q	B~C _ H S _ D G _ ~A _ ~T _ R*
(9-1a-s) a-q	B~C _ H S~V _ ~T _ D G _ ~A _ W*
(9-1a-aa) a-j	B~C _ H D G _ ~A _ E*
(9-1b) a-z	B~C S* _ D I _ H _ ~A _ T*
(9-1b-c) a-q	B~C S _ ~T _ D I _ H _ ~A _ R*
(9-1b-f) a-q	B~C S _ D I _ ~T _ H _ ~A _ R*
(9-1b-m) a-q	B~C S _ D I _ H _ ~A _ ~T _ R*
(9-1c) a-af	B~C _ D I _ ~A _ *S H _ T*
(9-1c-i) a-r	B~C _ D I _ ~A _ T~ _ ~S H _ J*
(9-1c-i-a) a-t	B~C _ J _ D I _ ~A _ T~V _ ~S H _ W*
(9-1c-i-d)	B~C _ D I _ J _ ~A _ T~ _ ~S H _
(9-1c-i-h) a-t	B~C _ D I _ ~A _ J _ T~ _ O~S H _ N*
(9-1c-i-h-h)	B~C _ D I _ ~A _ N _ J _ T~ _ O~S H _ G*
(9-1c-i-h-o)	B~C _ D I _ ~A _ J _ T~ _ N _ O~S H _ G*
(9-1c-i-h-r)	B~C _ D I _ ~A _ J _ T~ _ O~S H _ N _ G*
(9-1c-i-p) a-t	B~C _ D I _ ~A _ T~ _ O~S H _ J _ N*
(9-1c-i-p-h) a-w	B~C _ D I _ ~A _ N _ T~ _ O~S H _ J _ G*
(9-1c-i-p-r) a-w	B~C _ D I _ ~A _ T~ _ O~S H _ J _ N _ G*
(9-1c-s) a-q	B~C _ ~T _ D I _ ~A _ S H _ R*
(9-1c-s-i) a-u	B~C _ ~T _ D I _ R~ _ ~A _ S H _ J*
(9-1c-s-k) a-p	B~C _ ~T _ D I _ R~A _ S H _ J*
(9-1c-s-m) a-u	B~C _ ~T _ D I _ ~A _ R~ _ S H _ E*
(9-1c-u) a-q	B~C _ D I _ T~ _ ~A _ S H _ R*
(9-1c-u-c) a-u	B~C _ ~R _ D I _ T~ _ ~A _ S H _ J*
(9-1c-w) a-l	B~C _ D I _ T~A _ S H _ R*
(9-1c-w-c) a-p	B~C _ ~R _ D I _ T~A _ S H _ J*
(9-1c-y) a-q	B~C _ D I _ ~A _ T~ _ S H _ R*
(9-1c-y-c) a-u	B~C _ ~R _ D I _ ~A _ T~ _ S H _ G*
(9-1c-y-i) a-u	B~C _ D I _ ~A _ R~ _ T~ _ S H _ J*
(9-1c-y-i-a)	B~C _ J _ D I _ ~A _ R~ _ T~ _ S H _
(9-1c-y-i-d)	B~C _ D I _ J _ ~A _ R~ _ T~ _ S H _
(9-1c-y-i-h)	B~C _ D I _ ~A _ J _ R~ _ T~ _ S H _ G*
(9-1c-y-m)	B~C _ D I _ ~A _ T~ _ R~ _ S H _ (this is 9-1c-y-i)
(9-1c-y-i)	B~C _ D I _ ~A _ T~ _ R~ _ S H _
(9-1c-ac) a-q	B~C _ D I _ ~A _ S H _ ~T _ R*

References

[1] S. Akers, B. Krishnamurthy, A group theoretic model for symmetric interconnection networks, IEEE Trans. Comput. 38 (4) (1989) 555–566.
[2] D.S. Cohen, M. Blum, Improved bounds for sorting pancakes under a conjecture, Discrete Appl. Math. 61 (1995) 105–120.
[3] W.H. Gates, C.H. Papadimitriou, Bounds for sorting by prefix reversal, Discrete Math. 27 (1979) 47–57.
[4] M.H. Heydari, I.H. Sudborough, On the diameter of the pancake network, J. Algorithms 25 (1997) 67–94.
[5] Y. Kounoike, K. Kaneko, Y. Shinano, Computing the diameters of 14- and 15-pancake graphs, in: Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks, ISPAN 2005, 2005.
[6] K. Kaneko, Y. Shinano, Personal communication.
[7] N.J.A. Sloane (Ed.) The on-line encyclopedia of integer sequences, 2003. Published electronically at: http://www.research.att.com/~njas/sequences/.
[8] www.utdallas.edu/~hal/PancakeSolution.