Theoretical Computer Science 412 (2011) 822–834

Contents lists available at ScienceDirect

Theoretical Computer Science

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

# On average and highest number of flips in pancake sorting

Josef Cibulka *

*Department of Applied Mathematics, Charles University, Malostranské nám. 25, 118 00 Prague, Czech Republic*

A R T I C L E   I N F O

*Article history:*
Received 14 September 2008
Received in revised form 31 October 2010
Accepted 11 November 2010
Communicated by G. Italiano

*Keywords:*
Pancake problem
Burnt pancake problem
Permutations
Prefix reversals
Average-case analysis

A B S T R A C T

We are given a stack of pancakes of different sizes and the only allowed operation is to take several pancakes from the top and flip them. The unburnt version requires the pancakes to be sorted by their sizes at the end, while in the burnt version they additionally need to be oriented burnt-side down. We are interested in the largest value of the number of flips needed to sort a stack of $n$ pancakes, both in the unburnt version ($f(n)$) and in the burnt version ($g(n)$).

We present exact values of $f(n)$ up to $n = 19$ and of $g(n)$ up to $n = 17$ and disprove a conjecture of Cohen and Blum by showing that the burnt stack $-I_{15}$ is not the hardest to sort for $n = 15$.

We also show that sorting a random stack of $n$ unburnt pancakes can be done with at most $17n/12 + O(1)$ flips on average. The average number of flips of the optimal algorithm for sorting stacks of $n$ burnt pancakes is shown to be between $n + \Omega(n/\log n)$ and $7n/4 + O(1)$ and we conjecture that it is $n + \Theta(n/\log n)$.

Finally we show that sorting the stack $-I_n$ needs at least $\lfloor(3n + 3)/2\rfloor$ flips, which slightly increases the lower bound on $g(n)$. This bound together with the upper bound for sorting $-I_n$ found by Heydari and Sudborough in 1997 [10] gives the exact number of flips to sort it for $n \equiv 3 (\mathrm{mod}\ 4)$ and $n \geq 15$.



## 1. Introduction

The pancake problem was first posed in [4]. We are given a stack of pancakes each of which has a unique size and our aim is to sort them in as few operations as possible to obtain a stack of pancakes with sizes increasing from top to bottom. The only allowed sorting operation is a "spatula flip", in which a spatula is inserted beneath an arbitrary pancake, and all pancakes above the spatula are lifted and replaced in reverse order.

We can see the stack as a permutation $\pi$. A flip is then a prefix reversal of the permutation. The set of all permutations on $n$ elements is $S_n$, $f(\pi)$ is the minimum number of flips needed to obtain $(1, 2, 3, \ldots, n)$ from $\pi$ and $f(n)$ is the largest $f(\pi)$ over all permutations $\pi \in S_n$.

The exact values of $f(n)$ are known for all $n \leq 19$, see Table 1 for their list and references. In general $15\lfloor n/14\rfloor \leq f(n) \leq 18n/11+O(1)$. The upper bound is due to Chitturi et al. [2] and the lower bound was proved by Heydari and Sudborough [10]. These bounds improved the previous bounds $17n/16 \leq f(n) \leq (5n + 5)/3$ due to Gates and Papadimitriou [6], where the upper bound was also independently found by Györi and Turán [7].

A related problem in which the reversals are not restricted to intervals containing the first element received considerable attention in computational biology; see e.g. [9].

* Tel.: +420 776723819.
*E-mail addresses:* cibulka@kam.mff.cuni.cz, josef.cibulka@seznam.cz.

0304-3975/$ – see front matter 
doi:10.1016/j.tcs.2010.11.028

**Table 1**
Known values of $f(n)$, $g(n)$ and $g(-I_n)$.

| $n$ | $f(n)$ | | $g(n)$ | | $g(-I_n)$ | |
|---|---|---|---|---|---|---|
| 2 | 1 | [5] | 4 | [3] | 4 | [3] |
| 3 | 3 | [5] | 6 | [3] | 6 | [3] |
| 4 | 4 | [5] | 8 | [3] | 8 | [3] |
| 5 | 5 | [5] | 10 | [3] | 10 | [3] |
| 6 | 7 | [5] | 12 | [3] | 12 | [3] |
| 7 | 8 | [5] | 14 | [3] | 14 | [3] |
| 8 | 9 | [13] | 15 | [3] | 15 | [3] |
| 9 | 10 | [13] | 17 | [3] | 17 | [3] |
| 10 | 11 | [3] | 18 | [3] | 18 | [3] |
| 11 | 13 | [3] | 19 | [11] | 19 | [3] |
| 12 | 14 | [10] | 21 | [11] | 21 | [3] |
| 13 | 15 | [10] | 22 | Section 4 | 22 | [3] |
| 14 | 16 | [12] | 23 | Section 4 | 23 | [3] |
| 15 | 17 | [12] | 25 | Section 4 | 24 | [3] |
| 16 | 18 | [1] | 26 | Section 4 | 26 | [3] |
| 17 | 19 | [1] | 28 | Section 4 | 28 | [3] |
| 18 | 20 | Section 4 | | | 29 | [3] |
| 19 | 22 | Section 4 | | | 30 | Section 4 |
| 20 | | | | | 32 | Section 4 |
| $n \equiv 3 \pmod 4$ | | | | | $\lfloor \frac{3n+3}{2} \rfloor$ | Corollary 6 |

A variation on the pancake problem is the burnt pancake problem in which the pancakes are burnt on one of their sides. This time, the aim is not only to sort them by their sizes, but we also require that at the end, they all have their burnt sides down. Let $C = (\pi, v)$ denote a stack of $n$ burnt pancakes, where $\pi \in S_n$ is the permutation of the pancakes and $v \in \{0, 1\}^n$ is the vector of their orientations ($v_{\underline{i}} = 0$ if the $i$-th pancake from top is oriented burnt side down). Pancake $i$ will be represented by $\underline{i}$ if its burnt side is down and $\overline{i}$ if up. Let

$$I_n = \begin{pmatrix} \underline{1} \\ \underline{2} \\ \vdots \\ \underline{n} \end{pmatrix} \quad \text{and} \quad -I_n = \begin{pmatrix} \overline{1} \\ \overline{2} \\ \vdots \\ \overline{n} \end{pmatrix}.$$

Let $g(C)$ be the minimum number of flips that can transform $C$ to $I_n$ and let $g(n)$ be the largest $g(C)$ over all stacks $C$ of $n$ burnt pancakes.

Exact values of $g(n)$ are known for all $n \leq 17$, see Table 1. In 1979 Gates and Papadimitriou [6] provided the bounds $3n/2 - 1 \leq g(n) \leq 2n + 3$. Since then these were improved only slightly by Cohen and Blum [3] to $3n/2 \leq g(n) \leq 2n - 2$, where the upper bound holds for $n \geq 10$. The result $g(16) = 26$ further improves the upper bound to $2n - 6$ for $n \geq 16$. Cohen and Blum conjectured that the maximum number of flips for every $n$ is required by the stack $-I_n$. We present two counterexamples to the conjecture when $n = 15$ in Section 4.

The stack $-I_n$ can be sorted in $(3(n+1))/2$ flips for $n \equiv 3 \pmod 4$ and $n \geq 23$ [10]. In Section 3 we present a new formula for determining a lower bound on the number of flips needed to sort a given stack of burnt pancakes. The highest value this formula gives for a stack of $n$ pancakes, is $\lfloor (3(n+1))/2 \rfloor$ for the stack $-I_n$. These bounds together with the known values of $g(-I_{15})$ and $g(-I_{19})$ give $g(-I_n) = (3(n+1))/2$ if $n \equiv 3 \pmod 4$ and $n \geq 15$.

We are also interested in the average number of flips that algorithms will make while sorting a random stack. In Section 5 we design an algorithm that needs on average $7n/4 + O(1)$ flips to sort a stack of $n$ burnt pancakes. Section 6 describes a randomized algorithm for sorting $n$ unburnt pancakes with at most $17n/12 + O(1)$ flips on average. We also show that sorting a stack of $n$ unburnt pancakes requires on average at least $n - O(1)$ flips and in the burnt version, $n + \Omega(n/\log n)$ flips are needed on average. Section 7 introduces a conjecture that the average number of flips of the optimal algorithm for sorting burnt pancakes is $n + \Theta(n/\log n)$.

## 2. Terminology and notation

The stack obtained by flipping the whole stack $C$ is $\overline{C}$. The stack $-C$ is obtained from $C$ by changing the orientation of each pancake while keeping the order of pancakes unchanged.

If the top $i$ pancakes are flipped, the flip is an *$i$-flip*.

Two unburnt pancakes located next to each other are *adjacent* if their sizes differ by 1. Two burnt pancakes located next to each other are *adjacent* if they form a substack of $I_n$ or of $\overline{I_n}$, that is, if their sizes differ by 1 and the burnt side of the smaller pancake neighbors the unburnt side of the larger one. Two burnt pancakes located next to each other are *anti-adjacent* if they form a substack of $-I_n$ or of $\overline{-I_n}$, that is, if their sizes differ by 1 and the unburnt side of the smaller pancake neighbors the burnt side of the larger one.

$$\begin{pmatrix} 6 \\ 5 \\ 4 \\ \hline 1 \\ 2 \\ 3 \end{pmatrix} \begin{array}{l} \text{first block} \\ \\ \text{second block} \end{array} \qquad \begin{pmatrix} \underline{6} \\ \underline{5} \\ \hline \overline{4} \\ \hline \underline{1} \\ \underline{2} \\ \underline{3} \end{pmatrix} \begin{array}{l} \text{clan} \\ \text{free pancake} \\ \text{block} \end{array} \quad \xrightarrow{\text{contraction}} \quad \begin{pmatrix} \underline{4} \\ \underline{3} \\ \overline{2} \\ \underline{1} \end{pmatrix}$$

**Fig. 1.** Examples of blocks, clans and a contraction of a block.

In both versions a *block* in a stack $C$ is an inclusion-wise maximal substack $S$ of $C$ such that each two pancakes of $S$ on consecutive positions are adjacent. A substack $S$ of a stack $C$ with burnt pancakes is called a *clan* if $-S$ is a block in $-C$. Thus each pair of consecutive pancakes in a clan forms an anti-adjacency. Pancakes not taking part in a block or a clan are *free*. See Fig. 1. Observe that the blocks and clans are always pairwise disjoint.

A *contraction* of two adjacent burnt pancakes is an operation consisting of removing the one with larger number and decreasing by one the number of each pancake that was larger than the contracted ones. To *contract a block* means to contract all its pairs of adjacent pancakes one by one, thus transforming the block into a single pancake.

**Observation 1.** *Let $C$ be a stack of burnt pancakes with a pair $(p_1, p_2)$ of adjacent pancakes and let $C'$ be obtained from $C$ by contracting the two adjacent pancakes to a single pancake $p$. Then $g(C') = g(C)$.*

**Proof.** If we can sort $C'$ in $m$ steps, we can sort $C$ in $m$ steps as well — in each step, we insert the spatula below the same pancake as in an optimal sorting sequence for $C'$. The only difference is that whenever the spatula was inserted below $p$ in $C'$, we insert it below the lower of $p_1, p_2$ in $C$.

The stack $C'$ can be also obtained from $C$ by removing $p_1$. We can sort $C'$ by inserting the spatula below the same pancakes as in a sorting sequence for $C$. If a spatula was to be inserted below $p_1$ in $C$, we insert it below the pancake above it in $C'$. $\square$

A pair of adjacent unburnt pancakes can also be considered as a single burnt pancake with the burnt side where the larger pancake was. Their contraction leads to stacks with both burnt and unburnt pancakes, that we call *mixed stacks*. We say that two pancakes in a mixed stack are *adjacent* if the unburnt ones among the two can be oriented so that the two resulting burnt pancakes are adjacent. Notice that this definition generalizes the definition of adjacency for burnt and unburnt pairs of pancakes. The following are all the possibilities how a pair of pancakes can be adjacent:

$$\begin{pmatrix} \vdots \\ i \\ i+1 \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ i+1 \\ i \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ \underline{i} \\ i+1 \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ i \\ \underline{i+1} \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ i+1 \\ \overline{i} \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ \overline{i+1} \\ i \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ \underline{i} \\ \underline{i+1} \\ \vdots \end{pmatrix} \quad \begin{pmatrix} \vdots \\ \overline{i+1} \\ \overline{i} \\ \vdots \end{pmatrix}$$

We can *contract* any adjacent pair of pancakes in a mixed stack making it a single pancake burnt on the side where the larger pancake was. However, if we do not restrict contractions to burnt pancakes, it can be said only that the number of flips does not decrease:

**Observation 2.** *Let $C$ be a mixed stack with a pair $(p_1, p_2)$ of adjacent pancakes and let $C'$ be obtained from $C$ by contracting the two adjacent pancakes to a single pancake $p$. Then $g(C') \geq g(C)$.*

**Proof.** Again, while sorting $C$, we always insert the spatula below the same pancake (and below the lower of $p_1, p_2$ instead of $p$) as in an optimal sorting sequence for $C'$. $\square$

The inequality cannot be changed to an equality because, for example, the only sorting sequence of 3 flips for the following stack breaks the initial adjacency:

$$\begin{pmatrix} \overline{1} \\ 3 \\ 2 \end{pmatrix} \to \begin{pmatrix} 3 \\ \underline{1} \\ 2 \end{pmatrix} \to \begin{pmatrix} 2 \\ \overline{1} \\ 3 \end{pmatrix} \to \begin{pmatrix} \underline{1} \\ 2 \\ 3 \end{pmatrix}.$$

Contraction of the initial adjacency results in the stack $\begin{pmatrix} \overline{1} \\ \overline{2} \end{pmatrix} = -I_2$, which needs 4 flips.

## 3. Lower bound in the burnt version

In this section we improve the lower bound on the number of flips needed to sort $-I_n$.

A block (clan) is called a *surface block (clan)* if the topmost pancake is part of it, otherwise it is *deep*.

We will assign to each stack $C$ the value $v(C)$:

$$v(C) \stackrel{\text{def}}{=} a(C) - a^-(C) - \frac{1}{3}(b(C) - b^-(C)) + \frac{1}{3}(o(C) - o^-(C)) + l(C) - l^-(C) + \frac{1}{3}(ll(C) - ll^-(C)),$$

where

$$\begin{aligned}
a(C) &\stackrel{\text{def}}{=} \text{number of adjacencies}\\
b(C) &\stackrel{\text{def}}{=} \text{number of deep blocks}\\
o(C) &\stackrel{\text{def}}{=} \begin{cases} 1 & \text{if the pancake on top of the stack is the free } \overline{1} \text{ or} \\ & \text{if 1 is in a block (necessarily with 2)} \\ 0 & \text{otherwise} \end{cases}\\
l(C) &\stackrel{\text{def}}{=} \begin{cases} 1 & \text{if the lowest pancake is } \underline{n} \\ 0 & \text{otherwise} \end{cases}\\
ll(C) &\stackrel{\text{def}}{=} \begin{cases} 1 & \text{if the lowest pancake is } \underline{n} \text{ and the second lowest is } \underline{n-1} \\ 0 & \text{otherwise} \end{cases}\\
a^-(C) &\stackrel{\text{def}}{=} a(-C) = \text{number of anti-adjacencies in } C\\
b^-(C) &\stackrel{\text{def}}{=} b(-C) = \text{number of deep clans in } C\\
o^-(C) &\stackrel{\text{def}}{=} o(-C)\\
l^-(C) &\stackrel{\text{def}}{=} l(-C)\\
ll^-(C) &\stackrel{\text{def}}{=} ll(-C).
\end{aligned}$$

We want to show that a single flip never changes the value of $v$ by more than $4/3$. We consider two stacks $C$ and $C'$ of at least three pancakes, where $C'$ can be obtained from $C$ by a single flip and we let

$$\Delta v \stackrel{\text{def}}{=} v(C') - v(C).$$

First we introduce notation for contributions of each of the functions to $\Delta v$:

$$\begin{aligned}
\Delta a &\stackrel{\text{def}}{=} a(C') - a(C) & \Delta a^- &\stackrel{\text{def}}{=} -(a^-(C') - a^-(C))\\
\Delta b &\stackrel{\text{def}}{=} -\tfrac{1}{3}(b(C') - b(C)) & \Delta b^- &\stackrel{\text{def}}{=} \tfrac{1}{3}(b^-(C') - b^-(C))\\
\Delta o &\stackrel{\text{def}}{=} \tfrac{1}{3}(o(C') - o(C)) & \Delta o^- &\stackrel{\text{def}}{=} -\tfrac{1}{3}(o^-(C') - o^-(C))\\
\Delta l &\stackrel{\text{def}}{=} l(C') - l(C) & \Delta l^- &\stackrel{\text{def}}{=} -(l^-(C') - l^-(C))\\
\Delta ll &\stackrel{\text{def}}{=} \tfrac{1}{3}(ll(C') - ll(C)) & \Delta ll^- &\stackrel{\text{def}}{=} -\tfrac{1}{3}(ll^-(C') - ll^-(C)).
\end{aligned}$$

**Observation 3.** *Values of $\Delta a$, $\Delta a^-$, $\Delta l$ and $\Delta l^-$ are among $\{0, 1, -1\}$. Values of $\Delta b$, $\Delta b^-$, $\Delta o$, $\Delta o^-$, $\Delta ll$ and $\Delta ll^-$ are among $\{0, 1/3, -1/3\}$.*

**Proof.** The only nontrivial part is $\Delta b \leq 1/3$ and symmetrically $\Delta b^- \leq 1/3$. For a contradiction suppose that $\Delta b > 1/3$, which can only happen when one block was split into two free pancakes and another block became the surface block in a single flip. But the higher of the two pancakes that formed the split block will end on top of the stack after the flip. Therefore no block became the surface block. To show $\Delta b^- \leq 1/3$ we consider the flip $\phi : -C' \to -C$, for which we already know that $\Delta_\phi b \leq \frac{1}{3}$ and

$$\Delta_\phi b = -\frac{1}{3}(b(-C) - b(-C')) = -\frac{1}{3}(b^-(C) - b^-(C')) = \frac{1}{3}(b^-(C') - b^-(C)) = \Delta b^-. \quad \square$$

**Lemma 4.** *If $C'$ is a stack of at least three pancakes obtainable from $C$ by a single flip, then*

$$\Delta v = v(C') - v(C) \leq \frac{4}{3}.$$

*Therefore the minimum number of flips needed to sort a stack $C$ satisfies*

$$g(C) \geq \left\lceil \frac{3}{4}(v(I_n) - v(C)) \right\rceil.$$

**Proof.** The proof is based on restricting possible combinations of values of the above defined functions.

- Both $\Delta l$ and $\Delta l^-$ are positive. This implies that the pancake $n$ was at the bottom of the stack before the flip and also after the flip, but with a different orientation. This is not possible since $n > 1$.
- Exactly one of $\Delta l$ and $\Delta l^-$ is positive. The case $\Delta l^- > 0$ can be transformed to the case $\Delta l > 0$ by considering the flip $\phi : -C' \to -C$, for which

$$\begin{aligned}
\Delta_\phi v &\stackrel{\text{def}}{=} v(-C) - v(-C') = -v(C) - (-v(C')) = v(C') - v(C) = \Delta v,\\
\Delta_\phi l &\stackrel{\text{def}}{=} l(-C) - l(-C') = l^-(C) - l^-(C') = -(l^-(C') - l^-(C)) = \Delta l^-,\\
\Delta_\phi l^- &\stackrel{\text{def}}{=} l^-(-C) - l^-(-C') = \Delta l.
\end{aligned}$$

  The equality $v(-C) = -v(C)$ follows from the definition of $v(C)$.

  If the value of $l$ changes, the flip must be an $n$-flip. Therefore $\Delta a = \Delta a^- = 0$. Because $\Delta l = 1$, the pancake $\underline{n}$ has to be at the bottom of the stack after the flip, so $\Delta ll^- = 0$. Moreover neither a clan nor the pancake $\underline{1}$ could be on top of the stack before the flip so $\Delta b^- \le 0$ and $\Delta o^- \le 0$. Because $\Delta ll = 1/3$ implies a block on top of the stack before the flip and $\Delta o = 1/3$ implies no block on top of the stack after the flip, we obtain

$$\begin{aligned}
&\Delta ll = \frac{1}{3} \;\&\; \Delta o \le 0 &&\Rightarrow \Delta b \le 0,\\
&\Delta ll \le 0 \;\&\; \Delta o = \frac{1}{3} &&\Rightarrow \Delta b \le 0,\\
&\Delta ll = \frac{1}{3} \;\&\; \Delta o = \frac{1}{3} &&\Rightarrow \Delta b \le -\tfrac{1}{3}.
\end{aligned}$$

  In any of the cases $\Delta ll + \Delta o + \Delta b \le 1/3$ and $\Delta v \le 4/3$.

  From now on, we can assume $\Delta l$, $\Delta l^- \le 0$.
- At least one of $\Delta ll$ and $\Delta ll^-$ is positive. If both of them were positive then again the pancake $n$ would be at the bottom of the stack before and after the flip, each time with a different orientation. Similar to the previous case, we can choose $\Delta ll^- = 0$ and $\Delta ll = 1/3$. Because $\Delta l \le 0$, the last flip was an $(n-1)$-flip, the pancake at the bottom of the stack is $\underline{n}$ and the pancake on top of the stack before the flip was $\overline{(n-1)}$. Therefore $\Delta a = 1$, $\Delta a^- = 0$, $\Delta o^- \le 0$ and $\Delta b^- \le 0$.

  If pancake $n-1$ was part of a block before the flip, then this block became deep, otherwise pancakes $n-1$ and $n$ created a new deep block. Thus $\Delta b \le 0$. No block was destroyed and if $\Delta o = 1/3$, then no block became the surface block and thus $\Delta b = -1/3$. All in all $\Delta v \le 4/3$.

  In the remaining cases we have $\Delta l$, $\Delta l^-$, $\Delta ll$, $\Delta ll^- \le 0$.
- Both $\Delta o$ and $\Delta o^-$ are positive. Because $\Delta o^- > 0$ then either 1 was in a clan or on top of the stack with the burnt side down before the flip. If 1 was in a clan, then a single flip would not make it either a part of a block or a free $\overline{1}$ on top of the stack and thus $\Delta o$ would not be positive. Using a similar reasoning for $\Delta o$, we obtain that the flip was a 1-flip, the topmost pancake before the flip was $\underline{1}$ and the second pancake from the top is different from 2. Thus $\Delta a = \Delta a^- = \Delta b = \Delta b^- = 0$ and $\Delta v \le 2/3$.
- Exactly one of $\Delta o$ and $\Delta o^-$ is positive; without loss of generality it is $\Delta o$. This can happen only in two ways.
  - We did an $i$-flip, the topmost pancake before the flip was $\underline{2}$ and the $(i+1)$-st pancake is $\overline{1}$. Then $\Delta a = 1$, $\Delta a^- = 0$, $\Delta b \le 0$ and $\Delta b^- \le 0$ and so $\Delta v \le 4/3$.
  - We did an $i$-flip, the $i$-th pancake before the flip was $\underline{1}$ and neither the $(i-1)$-st nor the $(i+1)$-st pancake was $\underline{2}$. Then $\Delta b \le 0$ and $\Delta a^- \le 0$. If $\Delta a \le 0$, then $\Delta v \le 2/3$, otherwise $\Delta b^- \le 0$ and $\Delta v \le 4/3$.

  Now only $\Delta a$, $\Delta a^-$, $\Delta b$ and $\Delta b^-$ can be positive.
- If $\Delta a = \Delta a^- = 1$, then the flip was either

$$\begin{pmatrix} \overline{i-1} \\ \vdots \\ \underline{i+1} \\ \underline{i} \\ \vdots \end{pmatrix} \to \begin{pmatrix} \overline{i+1} \\ \vdots \\ \underline{i-1} \\ \underline{i} \\ \vdots \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} \underline{i+1} \\ \vdots \\ \overline{i-1} \\ \bar{i} \\ \vdots \end{pmatrix} \to \begin{pmatrix} \underline{i-1} \\ \vdots \\ \overline{i+1} \\ \bar{i} \\ \vdots \end{pmatrix}.$$

  In both cases the topmost pancake before the flip was not part of a clan and the topmost pancake after the flip is not part of a block, so the number of deep blocks increased and the number of deep clans decreased and $\Delta v \le 4/3$.
- Exactly one of $\Delta a$ and $\Delta a^-$ is positive; without loss of generality $\Delta a = 1$, $\Delta a^- \le 0$. Neither was a new clan created, nor did one become deep, so $\Delta b^- \le 0$ and $\Delta v \le 4/3$.
- None of $\Delta a$ and $\Delta a^-$ is positive, so $\Delta v \le 2/3$. $\square$

**Table 2**
Numbers of stacks of $n$ unburnt pancakes requiring exactly $m$ flips to sort.

| $n$ | $m$ | $\lvert S_n^m\rvert$ | $n$ | $m$ | $\lvert S_n^m\rvert$ | $n$ | $m$ | $\lvert S_n^m\rvert$ |
|---|---|---|---|---|---|---|---|---|
| 14 | 13 | 30,330,792,508 | 15 | 15 | 310,592,646,490 | 16 | 17 | 756,129,138,051 |
| 14 | 14 | 20,584,311,501 | 15 | 16 | 45,016,055,055 | 16 | 18 | 4,646,117 |
| 14 | 15 | 2,824,234,896 | 15 | 17 | 339,220 | 17 | 19 | 65,758,725 |
| 14 | 16 | 24,974 | | | | | | |

**Theorem 5.** *For every $n$*

$$g(-I_n) \geq \left\lfloor \frac{3(n+1)}{2} \right\rfloor.$$

**Proof.** The claim can easily be verified for $n \leq 2$, so we can assume $n \geq 3$.

It is easy to calculate that $v(I_n) = n + 2/3$ and $v(-I_n) = -n - 2/3$ and thus by Lemma 4 the number of flips needed to transform $-I_n$ to $I_n$ is at least

$$\left\lceil \frac{3}{4}\left(v(I_n) - v(-I_n)\right) \right\rceil = \left\lceil \frac{3}{4}\left(2n + \frac{4}{3}\right) \right\rceil = \left\lceil \frac{3}{2}n + 1 \right\rceil = \left\lfloor \frac{3(n+1)}{2} \right\rfloor. \quad \square$$

**Corollary 6.** *For all integers $n \geq 15$ with $n \equiv 3 \pmod 4$,*

$$g(-I_n) = \left\lfloor \frac{3(n+1)}{2} \right\rfloor.$$

**Proof.** The lower bound comes from Theorem 5. For all $n \geq 23$ with $n \equiv 3 \pmod 4$, the upper bound was proved by Heydari and Sudborough [10]. The exact value for $n = 15$ was computed by Cohen and Blum [3] and the exact value for $n = 19$ is computed in Section 4.

## 4. Computational results

A computer search found the following sequence of 30 flips that sorts the stack $-I_{19}$: (19, 14, 7, 4, 10, 18, 6, 4, 10, 19, 14, 4, 9, 11, 8, 18, 8, 11, 9, 4, 14, 19, 10, 4, 6, 18, 10, 4, 7, 14). Thus, using Theorem 5, $g(-I_{19}) = 30$.

We also determined the value $g(-I_{20}) = 32$ by the following approach. From [3, Theorem 7]: $g(-I_{20}) \leq g(-I_{19}) + 2 = 32$. From Theorem 5: $g(-I_{20}) \geq 31$ and from Lemma 4 it follows that if $g(-I_{20}) = 31$, then each flip of the optimal sorting sequence increases the value of the function $v$ by 4/3. But a depth-first search revealed that starting at $-I_{20}$ we can make a sequence of only at most 29 such flips.

The values $f(18) = 20$ and $f(19) = 22$ were computed by the method of Kounoike et al. [12] and Asai et al. [1]. It is an improvement of the method of Heydari and Sudborough [10]. Let $S_n^m$ be the set of stacks of $n$ unburnt pancakes requiring exactly $m$ flips to sort. For every stack $\pi \in S_n^m$, 2 flips always suffice to move the largest pancake to the bottom of the stack, obtaining stack $\pi'$. Since then, it never helps to move the largest pancake. Therefore $\pi'$ requires exactly the same number of flips as $\pi''$ obtained from $\pi'$ by removing the largest pancake and thus $\pi''$ requires at least $m - 2$ flips.

To determine $S_n^i$ for all $i \in \{m, m+1, \ldots, f(n)\}$, it is thus enough to consider the union of sets $S_{n-1}^{m'}$ with $m' \geq m - 2$. In each stack from this set, we try adding the pancake number $n$ to the bottom, flipping the whole stack and then trying every possible flip. The candidate set composed of the resulting and the intermediate stacks contains all the stacks from $\bigcup_{i=m}^{f(n)} S_n^i$. Now it remains to determine the value of $f(\pi)$ for each stack $\pi$ in the candidate set. As in [12,1], this is done using the A* search [8].

During the A* search, we need to compute a lower bound on the number of flips needed to sort a stack. It is counted in a more complicated way than in [12,1], where the size of the stack minus the number of adjacencies is used. We try all possible sequences of flips that create an adjacency in every flip. If some such sequence sorts the stack, it is optimal and we are done. Otherwise, we obtain a lower bound equal to the number of adjacencies that are needed to be made plus 1 (here we count pancake $n$ at the bottom of the stack as an adjacency).

In addition, we also use a heuristic to compute an upper bound. If the upper bound matches the lower bound they give the exact number of flips.

The sizes of the computed sets $S_n^m$ can be found in Table 2. It was previously known [10], that $f(18) \geq 20$ and $f(19) \geq 22$. No candidate stack of 18 pancakes needs 21 flips thus $f(18) = 20$. Then $f(19) = 22$ because $f(19) \leq f(18) + 2 = 22$.

The following modification of this method was also used to compute the values of $g(n)$ up to $n = 17$. Again, $\mathbb{C}_n^m$, the set of stacks of $n$ burnt pancakes requiring $m$ flips, is determined from the set $\bigcup_{m'=m-2}^{g(n-1)} \mathbb{C}_{n-1}^{m'}$, but in a slightly different way. In every stack of $n$ burnt pancakes other than $-I_n$ (which must be treated separately), some two pancakes can be joined in two flips [3, Theorem 1]. Observation 1 allows us to contract the two adjacent pancakes, which decreases the size of the stack. The reverse process is again used to determine the stacks of the candidate set, which are then processed by the A* search.

**Table 3**
Numbers of stacks of $n$ burnt pancakes requiring exactly $m$ flips to sort.

| $n$ | $m$ | $\lvert\mathbb{C}_n^m\rvert$ | $n$ | $m$ | $\lvert\mathbb{C}_n^m\rvert$ | $n$ | $m$ | $\lvert\mathbb{C}_n^m\rvert$ | $n$ | $m$ | $\lvert\mathbb{C}_n^m\rvert$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 10 | 15 | 22,703,532 | 11 | 17 | 5,928,175 | 12 | 19 | 344,884 | 13 | 21 | 15,675 |
| 10 | 16 | 179,828 | 11 | 18 | 10,480 | 12 | 20 | 265 | 13 | 22 | 4 |
| 10 | 17 | 523 | 11 | 19 | 36 | 12 | 21 | 1 | 14 | 23 | 122 |
| 10 | 18 | 1 | | | | | | | 15 | 25 | 2 |

During the A* search, we compute two lower bounds and take the larger one. One lower bound is computed from the formula in Lemma 4. To compute the other lower bound, we try all possible sequences of flips that create an adjacency in all but at most two flips. If no such sequence sorts the stack, we obtain a lower bound equal to the number of adjacencies that are needed to be made plus 3.

In the stacks visited during the A* search, we can contract all pairs of adjacent burnt pancakes thanks to Observation 1. If the resulting stack has at most nine pancakes, we look up the exact number of flips in a table previously computed by a breadth-first search starting at $I_9$.

Sizes of the computed sets $\mathbb{C}_n^m$ can be found in Table 3. No stack of 16 pancakes needs 27 flips thus $g(16) = 26$ because $g(-I_{16}) = 26$. Then $g(17) = 28$ because $g(-I_{17}) = 28$ and $g(17) \leq g(16) + 2 = 28$ [3, Theorem 8].

The stack obtained from $-I_n$ by flipping the topmost pancake is known as $J_n$ [3]. Let $Y_n$ be the stack obtained from $-I_n$ by changing the orientation of the second pancake from the bottom. The two found stacks of 15 pancakes requiring 25 flips are $J_{15}$ and $Y_{15}$ and they are the first known counterexamples to the Cohen–Blum conjecture which claimed that for every $n$, $-I_n$ requires the largest number of flips among all stacks of $n$ pancakes. However, no other $J_n$ or $Y_n$ with $n \leq 20$ is a counterexample to the conjecture.

The majority of the computations were done on computers of the CESNET METACentrum grid. Some of the computations also took place on computers at the Department of Applied Mathematics of Charles University in Prague.

Data and source codes of programs mentioned above can be downloaded from the following webpage: http://kam.mff.cuni.cz/~cibulka/pancakes.

## 5. Average number of flips in the burnt version

Let $\mathbb{C}_n$ denote the set of all stacks of $n$ burnt pancakes. We are interested in the average number of flips of the optimal algorithm for sorting stacks of $n$ burnt pancakes, that is, in

$$\mathrm{av}_{\mathrm{opt}}(n) \stackrel{\mathrm{def}}{=} \frac{\sum_{C \in \mathbb{C}_n} g(C)}{|\mathbb{C}_n|}.$$

**Theorem 7.** *For any $n \geq 16$*

$$\mathrm{av}_{\mathrm{opt}}(n) \geq n + \frac{n}{16 \log_2 n} - \frac{3}{2}.$$

**Proof.** We first count the expected number of adjacencies in a stack of $n$ burnt pancakes. A stack has $n - 1$ pairs of pancakes in consecutive positions. For each such pair of pancakes, there are $4n(n-1)$ equally probable combinations of their values and orientations and the pancakes form an adjacency in exactly $2(n-1)$ of them. From the linearity of expectation

$$\mathbb{E}[\mathrm{adj}] = (n-1)\frac{1}{2n} = \frac{1}{2}\frac{n-1}{n}.$$

Therefore at least half of the stacks have no adjacency.

- First we take a half of the stacks such that it contains all the stacks which have some adjacency. The stacks of this half have less than 1 adjacency on average. Each flip creates at most one adjacency, therefore when we want to obtain the stack $I_n$ with $n-1$ adjacencies, we need at least $n-2$ flips on average.
- The other half contains $n! \cdot 2^{n-1}$ stacks each with no adjacency, thus requiring at least $n-1$ flips. For each stack we take one of the shortest sequences of flips that create the stack from $I_n$ and call it the *creating sequence* of the stack. We will now count the number of different creating sequences of length at most $m \stackrel{\mathrm{def}}{=} n - 1 + n/(4\log_2 n)$, which will give an upper bound on the number of stacks with no adjacency that can be sorted in $m$ flips. Shorter creating sequences will be followed by several 0-flips, therefore we will consider $n+1$ possible flips. A *split-flip* is a flip in a creating sequence that decreases the number of adjacencies to a value smaller than the lowest value obtained before the flip. Therefore there are exactly $n-1$ split-flips in each of our creating sequences. In a creating sequence, the $i$-th split-flip removes one of

$n-i$ existing adjacencies and therefore there are $n-i$ possibilities to make the $i$-th split-flip. The number of different creating sequences of length $m$ is at most

$$\begin{aligned}\binom{m}{n-1}\cdot(n-1)!\cdot(n+1)^{m-(n-1)} &= \binom{n-1+\frac{n}{4\log_2 n}}{\frac{n}{4\log_2 n}}\cdot(n-1)!\cdot(n+1)^{n/(4\log_2 n)}\\ &\le \left(n-1+\tfrac{n}{4\log_2 n}\right)^{n/(4\log_2 n)}\cdot(n-1)!\cdot(2n)^{n/(4\log_2 n)}\\ &\le (n-1)!\cdot(2n)^{n/(4\log_2 n)}\cdot(2n)^{n/(4\log_2 n)}\\ &\le (n-1)!\cdot\left(n^{5/4}\right)^{2n/(4\log_2 n)}\\ &\le (n-1)!\cdot 2^{5n/8}\\ &< \tfrac{1}{4}n!\cdot 2^n.\end{aligned}$$

Therefore at least half of the stacks with no adjacency need more than $n-1+n/(4\log_2 n)$ flips. Every stack with no adjacency needs at least $n-1$ flips and thus in this case the average number of flips is at least

$$n-1+\frac{n}{8\log_2 n}.$$

The overall average number of flips is then

$$\mathrm{av}_{\mathrm{opt}}(n)\ge n-\frac{3}{2}+\frac{n}{16\log_2 n}.\quad\square$$

The algorithm of Gates and Papadimitriou for sorting burnt pancakes [6] works in iterations. In each iteration, a new adjacency is created while all existing adjacencies are preserved. Thus we can look at the algorithm as if it was contracting adjacent pairs of pancakes. Starting from a stack chosen uniformly at random from $\mathbb{C}_n$, one iteration takes 1.5 flips on average and shrinks the size of the stack to $n-1$. However, the result is not uniformly distributed among the stacks from $\mathbb{C}_{n-1}$, so we cannot conclude that it takes $3/2n+O(1)$ flips on average to sort a random stack. We will overcome this problem by adding two more flips in some cases, which will lead to an algorithm that makes $7n/4+O(1)$ flips on average.

Each iteration of our Algorithm $\mathcal{B}$ will also consist of making two pancakes adjacent and contracting them. However, we will not contract pairs of pancakes existing already in the input stack (as can be seen in the proof of Theorem 7, there are very few such adjacencies, so the benefit would be negligible). We stop when the number of pancakes is two and the algorithm can transform the stack to the stack ($\underline{1}$) in at most four flips.

To *cyclically renumber the pancakes by shift s* means that pancake number $j$ will become $j+s+kn$, where $k$ is an integer chosen so as to have the result inside the interval $\{1,\dots,n\}$. For simplicity of the analysis, we use the trick [6] to consider the numbers $\{1,\dots,n\}$ as a cyclic sequence, thus allowing the pair of pancakes 1 and $n$ to form an adjacency. Under this simplification, a cyclic renumbering does not change the pairs of pancakes that are adjacent.

At the beginning of each iteration, the Algorithm $\mathcal{B}$ cyclically renumbers the pancakes by $s=(2-\pi(1))$, that is, the topmost pancake will have number 2. Let $\mathbb{C}_n^2$ be the set of stacks of $n$ burnt pancakes with the pancake number 2 on top.

The only negative effect of the cyclic renumbering in the course of the algorithm is, that when we end up with the sorted stack, we in fact have some cyclic renumbering of it, that is

$$\begin{pmatrix}\underline{s+1}\\ \underline{s+2}\\ \vdots\\ \underline{n}\\ \underline{1}\\ \underline{2}\\ \vdots\\ \underline{s}\end{pmatrix},$$

where the shift $s\in\{0,1,\dots n-1\}$. This stack needs at most four more flips to become $I_n$. We will do four flips at the end even if they are not necessary. Then the number of flips of the algorithm will not be changed by the cyclic renumbering of pancakes and we may consider only stacks from $\mathbb{C}_n^2$.

1. If the stack from $\mathbb{C}_n^2$ can be flipped so that the topmost pancake will form an adjacency, we will do it:

$$\begin{pmatrix}\underline{2}\\ \overline{X}\\ \underline{1}\\ Y\end{pmatrix}\to\begin{pmatrix}X\\ \underline{2}\\ \underline{1}\\ Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\ \underline{1}\\ Y'\end{pmatrix}\in\mathbb{C}_{n-1}\quad\text{or}\quad\begin{pmatrix}\overline{2}\\ X\\ \underline{3}\\ Y\end{pmatrix}\to\begin{pmatrix}\overline{X}\\ \underline{2}\\ \underline{3}\\ Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\ \underline{2}\\ Y'\end{pmatrix}\Leftrightarrow\begin{pmatrix}X''\\ \underline{1}\\ Y''\end{pmatrix}\in\mathbb{C}_{n-1}.$$

Notice that each stack from $\mathbb{C}_{n-1}$ appears as a result of this process for exactly one stack from $\mathbb{C}_n^2$.

2. If no adjacency can be created in a single flip, we will look at both pancakes 1 and 3 and analyse all possible cases. Note that this time when 2 has its burnt side up, then 3 has its burnt side up and similarly $\underline{2}$ implies $\underline{1}$.

$$\begin{pmatrix}\underline{2}\\X\\\underline{1}\\Y\\\underline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{2}\\X\\\underline{1}\\Y\\\underline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{Y}\\\overline{1}\\\overline{X}\\\underline{2}\\\underline{3}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}Y'\\\overline{1}\\X'\\\underline{2}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}\qquad\begin{pmatrix}\underline{2}\\X\\\underline{3}\\Y\\\underline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{2}\\X\\\underline{3}\\Y\\\underline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{X}\\\underline{2}\\\underline{3}\\Y\\\underline{1}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\underline{2}\\Y'\\\underline{1}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}$$

$$\begin{pmatrix}\underline{2}\\X\\\underline{1}\\Y\\\overline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\underline{3}\\\overline{Y}\\\overline{1}\\\overline{X}\\\overline{2}\\Z\end{pmatrix}\to\begin{pmatrix}X\\\underline{1}\\Y\\\overline{3}\\\overline{2}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\underline{1}\\Y'\\\overline{2}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}\qquad\begin{pmatrix}\underline{2}\\X\\\overline{3}\\Y\\\underline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\underline{3}\\\overline{X}\\\overline{2}\\Y\\\underline{1}\\Z\end{pmatrix}\to\begin{pmatrix}X\\\overline{3}\\\overline{2}\\Y\\\underline{1}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\overline{2}\\Y'\\\underline{1}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}$$

$$\begin{pmatrix}\overline{2}\\X\\\overline{3}\\Y\\\overline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\underline{2}\\X\\\overline{3}\\Y\\\overline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{Y}\\\underline{3}\\\overline{X}\\\overline{2}\\\overline{1}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}Y'\\\underline{2}\\X'\\\overline{1}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}\qquad\begin{pmatrix}\overline{2}\\X\\\overline{1}\\Y\\\overline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\underline{2}\\X\\\overline{1}\\Y\\\overline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{X}\\\overline{2}\\\overline{1}\\Y\\\overline{3}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\overline{1}\\Y'\\\overline{2}\\Z'\end{pmatrix}\in\mathbb{C}_{n-1}$$

$$\begin{pmatrix}\overline{2}\\X\\\overline{3}\\Y\\\underline{1}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{1}\\\overline{Y}\\\underline{3}\\\overline{X}\\\underline{2}\\Z\end{pmatrix}\to\begin{pmatrix}X\\\overline{3}\\Y\\\underline{1}\\\underline{2}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\overline{2}\\Y'\\\underline{1}\\Z'\end{pmatrix}\to\begin{pmatrix}\overline{Z}'\\\overline{1}\\\overline{Y}'\\\underline{2}\\\overline{X}'\end{pmatrix}\to\begin{pmatrix}Y'\\\underline{1}\\Z'\\\underline{2}\\\overline{X}'\end{pmatrix}\in\mathbb{C}_{n-1}$$

$$\begin{pmatrix}\overline{2}\\X\\\underline{1}\\Y\\\overline{3}\\Z\end{pmatrix}\to\begin{pmatrix}\overline{1}\\\overline{X}\\\underline{2}\\Y\\\overline{3}\\Z\end{pmatrix}\to\begin{pmatrix}X\\\underline{1}\\\underline{2}\\Y\\\overline{3}\\Z\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\underline{1}\\Y'\\\overline{2}\\Z'\end{pmatrix}\to\begin{pmatrix}\overline{Z}'\\\underline{2}\\\overline{Y}'\\\overline{1}\\\overline{X}'\end{pmatrix}\to\begin{pmatrix}Y'\\\overline{2}\\Z'\\\overline{1}\\\overline{X}'\end{pmatrix}\in\mathbb{C}_{n-1}.$$

Again each stack from $\mathbb{C}_{n-1}$ appears as a result of the process for exactly one stack from $\mathbb{C}_n^2$.

**Theorem 8.** *Algorithm $\mathcal{B}$ sorts stacks of n burnt pancakes with the average number of flips at most*

$$\frac{7}{4}n+5.$$

**Proof.** Let $h(C)$ be the number of flips used by the algorithm to sort the stack $C$ and let

$$H(n) \stackrel{\text{def}}{=} \textstyle\sum_{C\in\mathbb{C}_n} h(C),$$
$$\text{av}(n) \stackrel{\text{def}}{=} \tfrac{H(n)}{|\mathbb{C}_n|} = \tfrac{H(n)}{2n|\mathbb{C}_{n-1}|}.$$

As was mentioned, the renumbering at the beginning of each iteration is done so that $H(n) = n\cdot\sum_{C\in\mathbb{C}_n^2} h(C)$ and $\text{av}(2)\le 8$.

In case 1, it was enough to make 1 flip while in case 2, the average number of flips we did was 5/2.

All in all

$$H(n) = n\cdot\left(\sum_{C\in\mathbb{C}_{n-1}}(h(C)+1)+\sum_{C\in\mathbb{C}_{n-1}}\left(h(C)+\frac{5}{2}\right)\right) = 2nH(n-1)+\frac{7}{2}n|C_{n-1}|,$$

$$\text{av}(n) = \frac{2nH(n-1)+\frac{7}{2}n|C_{n-1}|}{2n|C_{n-1}|} = \text{av}(n-1)+\frac{7}{4} = \text{av}(2)+\frac{7}{4}(n-2)\le\frac{7}{4}n+5. \quad\square$$

## 6. Randomized algorithm for the unburnt version

**Observation 9.** *Let* $\mathrm{av'}_{\mathrm{opt}}(n)$ *be the average number of flips of the optimal algorithm for sorting a stack of n unburnt pancakes. For any positive n*

$$\mathrm{av'}_{\mathrm{opt}}(n) \geq n - 2.$$

**Proof.** We will now count the expected number of adjacencies in a stack of $n$ pancakes. For the purpose of this proof we will consider the pancake number $n$ at the bottom of the stack as an additional adjacency; this has probability $1/n$. Pancakes on consecutive positions form an adjacency if their values differ by 1; the probability of this is $2/n$. Therefore the expected number of adjacencies is

$$\mathbb{E}[\mathrm{adj}] = \frac{1}{n} + (n-1)\frac{2}{n} < 2.$$

Each flip creates at most one adjacency, therefore when we want to obtain the stack $I_n$ with $n$ adjacencies, the average number of flips is at least $n - 2$. □

The Algorithm $\mathcal{U}$ is similar to Algorithm $\mathcal{B}$ from Section 5. Each iteration consists of creating a pair of adjacent pancakes and contracting them to a single burnt pancake. Therefore the algorithm will work with mixed stacks.

Let $\mathbb{M}_{n,b}$ denote the set of all mixed stacks of $n$ pancakes $b$ of which are burnt and let $\mathbb{M}^2_{n,b}$ be the stacks from $\mathbb{M}_{n,b}$ with pancake number 2 on top.

When there are only two pancakes left, Algorithm $\mathcal{U}$ sorts the stack in at most 4 flips. Similarly to Algorithm $\mathcal{B}$, we will sometimes cyclically renumber the pancakes. After renumbering them back at the end, we will do four flips to get the sorted stack.

The algorithm first cyclically renumbers the pancakes so as to have the topmost pancake numbered 2 thus obtaining a stack from $\mathbb{M}^2_{n,b}$. Then we look at the topmost pancake. If it is unburnt, we uniformly at random select whether to look at 1 or 3. Otherwise if the burnt side is down, we look at 1 and if it is up, we look at 3.

Notice that we could also look at both pancakes 1 and 3. But if we joined only two of the pancakes 1, 2 and 3 we would have to count the average number of flips for each combination not only of the number of pancakes and the number of burnt pancakes, but also of the number of pairs of pancakes of consecutive sizes exactly one of which is burnt. This would make the calculations very complicated. We could also join all three of them, but this would lead to a weaker result.

We split the set $\mathbb{M}^2_{n,b}$ into 4 sets $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$, $\mathbb{M}^{2,\mathrm{ub}}_{n,b}$, $\mathbb{M}^{2,\mathrm{bu}}_{n,b}$ and $\mathbb{M}^{2,\mathrm{bb}}_{n,b}$ based on whether the two pancakes we looked at are burnt (represented by the letter 'b') or unburnt ('u'). The first letter in the superscript stands for the pancake number 2 and the second one for the other pancake we looked at. A stack is *undecided* if the pancake 2 is unburnt and exactly one of pancakes 1 and 3 is unburnt. Undecided stacks have the same probability of getting into $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$ and into $\mathbb{M}^{2,\mathrm{ub}}_{n,b}$. Therefore we define the *grade of membership* of a stack in a set: If a stack is undecided, then its grade of membership is $1/2$ in $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$ and in $\mathbb{M}^{2,\mathrm{ub}}_{n,b}$ and 0 otherwise. Each other stack always gets to the same set in which its grade of membership is 1 and it is 0 for the other sets.

(uu) Both the pancakes we looked at are unburnt, so we join them in a single flip.

$$\begin{pmatrix} 2 \\ X \\ 1 \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{X} \\ 2 \\ 1 \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X' \\ \overline{1} \\ Y' \end{pmatrix} \in \mathbb{M}_{n-1,b+1} \quad \text{or} \quad \begin{pmatrix} 2 \\ X \\ 3 \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{X} \\ 2 \\ 3 \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X' \\ \underline{2} \\ Y' \end{pmatrix} \Leftrightarrow \begin{pmatrix} X'' \\ \underline{1} \\ Y'' \end{pmatrix} \in \mathbb{M}_{n-1,b+1}.$$

Each stack from $\mathbb{M}_{n-1,b+1}$ can be cyclically renumbered in exactly $b+1$ ways in order to have the pancake number 1 burnt. Observe that for a stack chosen uniformly at random from $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$ (with probabilities proportional to the grade of membership), the resulting stack is uniformly distributed among the stacks from $\mathbb{M}_{n-1,b+1}$ with pancake 1 burnt. Similar observations hold in the remaining cases as well.

(ub) The topmost pancake is unburnt, while the other pancake we looked at is burnt.

$$\begin{pmatrix} 2 \\ X \\ \overline{1} \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{X} \\ 2 \\ \overline{1} \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X' \\ \overline{1} \\ Y' \end{pmatrix} \in \mathbb{M}_{n-1,b} \quad \text{or} \quad \begin{pmatrix} 2 \\ X \\ \underline{1} \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{1} \\ \overline{X} \\ 2 \\ Y \end{pmatrix} \to \begin{pmatrix} X \\ \underline{1} \\ 2 \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X' \\ \underline{1} \\ Y' \end{pmatrix} \in \mathbb{M}_{n-1,b}.$$

The case when we looked at pancake 3 is similar.

(bu) The topmost pancake is burnt, while the other one we looked at is unburnt.

$$\begin{pmatrix} \overline{2} \\ X \\ 3 \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{X} \\ \underline{2} \\ 3 \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X' \\ \underline{2} \\ Y' \end{pmatrix} \Leftrightarrow \begin{pmatrix} X'' \\ \underline{1} \\ Y'' \end{pmatrix} \in \mathbb{M}_{n-1,b} \quad \text{or} \quad \begin{pmatrix} \underline{2} \\ X \\ 1 \\ Y \end{pmatrix} \to \begin{pmatrix} \overline{X} \\ \overline{2} \\ 1 \\ Y \end{pmatrix} \Leftrightarrow \begin{pmatrix} X'' \\ \overline{1} \\ Y'' \end{pmatrix} \in \mathbb{M}_{n-1,b}.$$

(bb) Both the pancakes we looked at are burnt. In half of the cases the two pancakes can be joined in a single flip:

$$\begin{pmatrix}\overline{2}\\X\\\underline{3}\\Y\end{pmatrix}\to\begin{pmatrix}\overline{X}\\\underline{2}\\\underline{3}\\Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\underline{2}\\Y'\end{pmatrix}\Leftrightarrow\begin{pmatrix}X''\\\underline{1}\\Y''\end{pmatrix}\in\mathbb{M}_{n-1,b-1}\quad\text{or}\quad\begin{pmatrix}\underline{2}\\X\\\overline{1}\\Y\end{pmatrix}\to\begin{pmatrix}\overline{X}\\\overline{2}\\\overline{1}\\Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X''\\\overline{1}\\Y''\end{pmatrix}\in\mathbb{M}_{n-1,b-1}.$$

Otherwise we need three flips to join the two pancakes:

$$\begin{pmatrix}\overline{2}\\X\\\overline{3}\\Y\end{pmatrix}\to\begin{pmatrix}\underline{2}\\X\\\overline{3}\\Y\end{pmatrix}\to\begin{pmatrix}\underline{\underline{3}}\\\overline{X}\\\overline{2}\\Y\end{pmatrix}\to\begin{pmatrix}X\\\overline{3}\\\overline{2}\\Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X'\\\overline{2}\\Y'\end{pmatrix}\Leftrightarrow\begin{pmatrix}X''\\\overline{1}\\Y''\end{pmatrix}\in\mathbb{M}_{n-1,b-1}$$

$$\text{or}\quad\begin{pmatrix}\underline{2}\\X\\\underline{1}\\Y\end{pmatrix}\to\begin{pmatrix}\overline{2}\\X\\\underline{1}\\Y\end{pmatrix}\to\begin{pmatrix}\overline{1}\\\overline{X}\\\underline{2}\\Y\end{pmatrix}\to\begin{pmatrix}X\\\underline{1}\\\underline{2}\\Y\end{pmatrix}\Leftrightarrow\begin{pmatrix}X''\\\underline{1}\\Y''\end{pmatrix}\in\mathbb{M}_{n-1,b-1}.$$

**Theorem 10.** *Algorithm $\mathcal{U}$ sorts a stack of n unburnt pancakes with the average number of flips at most*

$$\frac{17}{12}n+9,$$

*where the average is taken both over the stacks and the random bits.*

**Proof.** Let $k(C)$ be the average number of flips used by the algorithm to sort the stack $C$ and let

$$K(n,b)\ \stackrel{\text{def}}{=}\ \textstyle\sum_{C\in\mathbb{M}_{n,b}}k(C),$$
$$\mathrm{av}'(n,b)\ \stackrel{\text{def}}{=}\ \frac{K(n,b)}{|\mathbb{M}_{n,b}|}.$$

We have $\mathrm{av}'(1,0)=\mathrm{av}'(1,1)=4$, $\mathrm{av}'(2,b)\le 8$ for any $b\in\{0,1,2\}$ and $K(n,b)=n\cdot\sum_{C\in\mathbb{M}^2_{n,b}}k(C)$.

Let $\mathrm{av}'_{\mathrm{uu}}(n,b)$ be the weighted average number of flips used by Algorithm $\mathcal{U}$ to sort a stack from $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$, where the weight is the grade of membership of the stack in $\mathbb{M}^{2,\mathrm{uu}}_{n,b}$. In the same way we define $\mathrm{av}'_{\mathrm{ub}}(n,b)$, $\mathrm{av}'_{\mathrm{bu}}(n,b)$ and $\mathrm{av}'_{\mathrm{bb}}(n,b)$ for sets $\mathbb{M}^{2,\mathrm{ub}}_{n,b}$, $\mathbb{M}^{2,\mathrm{bu}}_{n,b}$ and $\mathbb{M}^{2,\mathrm{bb}}_{n,b}$, respectively.

The average number of flips in case (uu) is

$$\mathrm{av}'_{\mathrm{uu}}(n,b)=\mathrm{av}'(n-1,b+1)+1,$$

in case (ub)

$$\mathrm{av}'_{\mathrm{ub}}(n,b)=\mathrm{av}'(n-1,b)+\frac{3}{2},$$

in case (bu)

$$\mathrm{av}'_{\mathrm{bu}}(n,b)=\mathrm{av}'(n-1,b)+1$$

and in case (bb)

$$\mathrm{av}'_{\mathrm{bb}}(n,b)=\mathrm{av}'(n-1,b-1)+2.$$

After summing up all the above average numbers of flips multiplied by their probabilities, we obtain:

- For $1\le b<n$

$$\begin{aligned}\mathrm{av}'(n,b)\ &=\ \frac{(n-b)(n-b-1)}{n(n-1)}\mathrm{av}'_{\mathrm{uu}}(n,b)+\frac{(n-b)b}{n(n-1)}\left(\mathrm{av}'_{\mathrm{ub}}(n,b)+\mathrm{av}'_{\mathrm{bu}}(n,b)\right)+\frac{b(b-1)}{n(n-1)}\mathrm{av}'_{\mathrm{bb}}(n,b)\\&=\ \frac{(n-b)(n-b-1)}{n(n-1)}(1+\mathrm{av}'(n-1,b+1))+\\&\quad+2\frac{(n-b)b}{n(n-1)}\left(\frac{5}{4}+\mathrm{av}'(n-1,b)\right)+\frac{b(b-1)}{n(n-1)}\left(2+\mathrm{av}'(n-1,b-1)\right).\end{aligned}$$

- For $b=0$

$$\mathrm{av}'(n,0)=\frac{n(n-1)}{n(n-1)}\mathrm{av}'_{\mathrm{uu}}(n,0)=1+\mathrm{av}'(n-1,1).$$

- For $b = n$

$$\mathrm{av}'(n,n) = \frac{n(n-1)}{n(n-1)}\mathrm{av}'_{\mathrm{bb}}(n,n) = 2 + \mathrm{av}'(n-1,n-1).$$

Instead of solving these recurrent formulas, we will use them to bound $\mathrm{av}'(n,b)$ from above by the following function:

$$\mathrm{av}^+(n,b) \stackrel{\mathrm{def}}{=} \frac{17}{12}n + \frac{7}{12}b - \frac{1}{6}\frac{(n-b+1)b}{n} + 9.$$

It remains to show that for every nonnegative $n$ and $b$, such that $b$ is not greater than $n$ we have

$$\mathrm{av}^+(n,b) \geq \mathrm{av}'(n,b).$$

We use induction on $n$.

- For $n = 1$ we have $\mathrm{av}'(1,b) = 4$ and it is easy to verify that $\mathrm{av}^+(1,b) \geq 4$ for $b \in \{0,1\}$.
- If $b = 0$, then the induction hypothesis gives

$$\begin{aligned}\mathrm{av}'(n,0) &= 1 + \mathrm{av}'(n-1,1) \leq 1 + \mathrm{av}^+(n-1,1)\\ &= 1 + \frac{17}{12}(n-1) + \frac{7}{12} - \frac{1}{6}\frac{n-1}{n-1} + 9 = \frac{17}{12}n + 9 = \mathrm{av}^+(n,0).\end{aligned}$$

- For $b = n$ we get

$$\begin{aligned}\mathrm{av}'(n,n) &= 2 + \mathrm{av}'(n-1,n-1) \leq 2 + \mathrm{av}^+(n-1,n-1)\\ &= 2 + \frac{17}{12}(n-1) + \frac{7}{12}(n-1) - \frac{1}{6} + 9 = \frac{17}{12}n + \frac{7}{12}n - \frac{1}{6} + 9 = \mathrm{av}^+(n,n).\end{aligned}$$

- In the case $1 \leq b < n$

$$\begin{aligned}n(n-1)(\mathrm{av}^+(n,b) - \mathrm{av}'(n,b)) &\geq n(n-1)\mathrm{av}^+(n,b) - (n-b)(n-b-1)(1 + \mathrm{av}^+(n-1,b+1))\\ &\quad - 2(n-b)b\left(\frac{5}{4} + \mathrm{av}^+(n-1,b)\right) - b(b-1)\left(2 + \mathrm{av}^+(n-1,b-1)\right)\\ &= \frac{b}{n-1}\left(\frac{1}{3}n - \frac{1}{3}b\right) > 0.\end{aligned}$$

Therefore

$$\mathrm{av}'(n,0) \leq \mathrm{av}^+(n,0) = \frac{17}{12}n + 9. \quad \square$$

## 7. An open problem

Although the algorithms $\mathcal{B}$ and $\mathcal{U}$ presented in Sections 5 and 6 have a good guaranteed average number of flips, experimental results show that both of them are usually outperformed by the corresponding algorithms of Gates and Papadimitriou. The experimental average numbers of flips of the two new algorithms are very close to their upper bounds calculated in Theorems 8 and 10 and the averages for the algorithms of Gates and Papadimitriou are in Table 4.

We will now design one more polynomial-time algorithm for the burnt version, for which no guarantee of the average number of flips will be given, but its experimental results are close to the lower bound from Theorem 7.

Call a sequence of flips, each of which creates an adjacency, a *greedy sequence*. Note that since we are in the burnt version, there is always at most one possible flip that creates a new adjacency. In a random stack the probability that we can make an adjacency with the pancake on top in a single flip is 50%, therefore starting from a random stack, we can perform a greedy sequence of length $\log_2 n$ with probability roughly $1/n$. The idea of the algorithm is that whenever we cannot create an adjacency in a single flip we try all $n$ possible flips and do the one that can be followed by the longest greedy sequence.

As in the previous algorithms, two adjacent pancakes are always contracted. Pancakes 1 and $n$ can create an adjacency (1 is viewed as $(n+1) \bmod n$). Thus when the algorithm obtains the stack $(\underline{1})$, we need at most four more flips.

The experimental results together with Theorem 7 support the following conjecture.

**Conjecture 1.** *The average number of flips of the optimal algorithm for sorting burnt pancakes satisfies*

$$\mathrm{av}_{\mathrm{opt}}(n) = n + \Theta\left(\frac{n}{\log n}\right).$$

**Table 4**
Experimental results of algorithms. The average numbers of flips to sort a randomly generated stack of $n$ pancakes are: $s_{GP}$ for the algorithm of Gates and Papadimitriou for the unburnt version, $s_{GPB}$ for the algorithm of Gates and Papadimitriou for the burnt version and $s_N$ for the algorithm described in this section (for the burnt version).

| $n$ | $s_{GP}$ | $s_{GPB}$ | $s_N$ | $n + n/\log_2 n$ | Stacks generated |
|---:|---:|---:|---:|---:|---:|
| 10 | 11.129 | 15.383 | 14.935 | 13.010 | 1000,000 |
| 100 | 122.925 | 150.887 | 123.463 | 115.051 | 100,000 |
| 1,000 | 1,240.949 | 1,502.926 | 1,127.901 | 1,100.343 | 10,000 |
| 10,000 | 12,408.686 | 15,002.212 | 10,863.502 | 10,752.570 | 1,000 |
| 100,000 | 124,115.000 | 150,063.000 | 106,608.900 | 106,220.600 | 10 |
| 1,000,000 | 1,241,263.600 | 1,499,875.600 | 1,053,866.000 | 1,050,171.666 | 5 |

## Acknowledgements

I would like to thank Pavel Valtr and Jan Kratochvíl who led the seminar under which this article has originated. I would also like to thank Jan Kynčl, Bernard Lidický, Radovan Šesták and Marek Tesař, who were participants of the seminar, for their notable comments. I am especially grateful to Pavel Valtr for suggesting to study the average numbers of flips.

Work on this paper was supported by the project 1M0545 of the Ministry of Education of the Czech Republic and by the Czech Science Foundation under the contract no. 201/09/H057. The access to the METACentrum computing facilities provided under the research intent MSM6383917201 is highly appreciated.

## References

[1] Shogo Asai, Yuusuke Kounoike, Yuji Shinano, Keiichi Kaneko, Computing the diameter of 17-pancake graph using a PC cluster, in: Wolfgang E. Nagel, Wolfgang V. Walter, Wolfgang Lehner (Eds.), Euro-Par, in: Lecture Notes in Computer Science, vol. 4128, Springer, 2006, pp. 1114–1124.
[2] B. Chitturi, W. Fahle, Z. Meng, L. Morales, C.O. Shields, I.H. Sudborough, W. Voit, An $(18/11)n$ upper bound for sorting by prefix reversals, Theoret. Comput. Sci. 410 (36) (2009) 3372–3390.
[3] David S. Cohen, Manuel Blum, On the problem of sorting burnt pancakes, Discrete Appl. Math. 61 (2) (1995) 105–120.
[4] Harry Dweighter, Problem E2569, Amer. Math. Monthly 82 (1975) 1010.
[5] M.R. Garey, D.S. Johnson, S. Lin, Amer. Math. Monthly 84 (1977) 296.
[6] William H. Gates, Christos H. Papadimitriou, Bounds for sorting by prefix reversal, Discrete Math. 27 (1979) 45–57.
[7] E. Györi, G. Turán, Stack of pancakes, Studia Sci. Math. Hungar. 13 (1978) 133–137.
[8] P. Hart, N. Nilsson, B. Raphael, A formal basis for the heuristic determination of minimum cost paths., IEEE Trans. Syst. Sci. Cybern. 4 (2) (1968) 100–107.
[9] Brian Hayes, Sorting out the genome, Amer. Scientist 95 (2007) 386–391.
[10] Mohammad H. Heydari, I. Hal Sudborough, On the diameter of the pancake network, J. Algorithms 25 (1) (1997) 67–94.
[11] Richard E. Korf, Minimizing disk I/O in two-bit breadth-first search, in: Dieter Fox, Carla P. Gomes (Eds.), AAAI, AAAI Press, 2008, pp. 317–324.
[12] Yuusuke Kounoike, Keiichi Kaneko, Yuji Shimano, Computing the diameters of 14- and 15-pancake graphs, in: ISPAN '05: Proceedings of the 8th International Symposium on Parallel Architectures, Algorithms and Networks, IEEE Computer Society, Washington, DC, USA, 2005, pp. 490–495.
[13] D.P. Robbins, Personal communication cited in W.H. Gates and C.H. Papadimitriou, Bounds for sorting by prefix reversal, Discrete Math. 27 (1979) 47–57.