Theoretical Computer Science 108 (1993) 271–290
Elsevier

The complexity of matrix transposition on one-tape off-line Turing machines with output tape*

Martin Dietzfelbinger**, ***

Fachbereich Mathematik/Informatik and Heinz-Nixdorf-Institut, Universität-GH-Paderborn, W-4790 Paderborn, Germany

Wolfgang Maass****

IIG, Technische Universität Graz, Klosterwiesgasse 32, A-8010 Graz, Austria, and Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Box 4348, Chicago, IL 60680, USA

Communicated by R.V. Book
Received June 1989
Revised April 1991

Abstract

Dietzfelbinger, M. and W. Maass, The complexity of matrix transposition on one-tape off-line Turing machines with output tape, Theoretical Computer Science 108 (1993) 271–290.

A series of existing lower bound results for deterministic one-tape Turing machines is extended to another, stronger such model suitable for the computation of functions: one-tape off-line Turing machines with a write-only output tape. ("Off-line" means: having a two-way input tape.) The following optimal lower bound is shown: Computing the transpose of Boolean $l \times l$-matrices takes $\Omega(l^{5/2}) = \Omega(n^{5/4})$ steps on such Turing machines. ($n = l^2$ is the length of the input.)

1. Introduction

During the last few years lower bound arguments for a sequence of restricted Turing machines (TMs) of increasing power have been developed. Techniques have

Correspondence to: M. Dietzfelbinger, Fachbereich Mathematik/Informatik, Universität-GH-Paderborn, W-4790 Paderborn, Germany. Email addresses of the authors: martind@uni-paderborn.de and maass@igi.tu-graz.ac.at.

* A preliminary version of this paper appeared in: T. Lepistö, A. Salomaa, eds., *Proc. 15th ICALP*, Lecture Notes in Computer Science, Vol. 317 (Springer, Berlin, 1988) 188–200.

** Partially supported by NSF Grant DCR-8504247 and DFG Grants Me 872/1-1 and We 1066/1-2.

*** This work is based on a part of the first author's Ph.D. Thesis at the University of Illinois at Chicago, Chicago, Illinois, USA.

**** Partially supported by NFS Grants DCR-8504247, CCR-8703889 and CCR-8903398.

been devised that make it possible to prove optimal superlinear lower bounds on the computation time for several concrete computational problems on one-tape TMs without input tape [4], on one-tape TMs with a one-way input tape ("on-line one-tape TMs") [9, 14], and finally on one-tape TMs with a two-way input tape ("off-line one-tape TMs"; this is the standard model for the definition of space-complexity classes). For this model an optimal lower bound of $\Omega(n^{3/2}/(\log n)^{1/2})$ for the matrix transposition function [10, 3] and a barely superlinear lower bound of $\Omega(n \log n/\log\log n)$ for a related decision problem [11] have been established.

In this paper we consider the next more powerful type of restricted Turing machine (for which the preceding lower bound arguments do not suffice): off-line one-tape TMs with an additional output tape. Whereas the addition of the output tape obviously makes no difference for solving decision problems, it was already noted in [10], respectively, [3] that these machines can perform matrix transposition in $O(n^{5/4})$ steps, as opposed to $\Omega(n^{3/2}/(\log n)^{1/2})$ steps for the previously considered version without output tape, where the output has to appear on the worktape.

This stronger model is also of some interest from a technical point of view, because it exhibits a feature that is characteristic for TMs with several worktapes (which are so far intractable for lower bound arguments): the extensive use of the worktape as an intermediate storage device. This feature played only a minor role in the analysis of matrix transposition on one-tape off-line TMs without output tape, because one could easily show that any use of the worktape as an intermediate storage device is inefficient for this model: Once some bits have been written on the worktape, they can be moved later only by time-consuming sweeps of the worktape head. During each sweep at most $\log n$ bits can be moved, where n is the length of the input. (The number of bits that can be moved during one sweep is about $\log n$ rather than constant since the input tape can be used as a unary counter, thus can store up to $\log n$ bits. This feature of one-tape TMs with two-way input tape can be used to show that such machines can simulate $f(n)$-time-bounded k-tape TMs in $O(f(n)^2/\log n)$ steps, see [2].)

In this paper, we prove an optimal lower bound of $\Omega(n^{5/4})$ for the transposition of Boolean matrices on one-tape off-line TMs with output tape. This result also separates such TMs from k-tape TMs with $k \geqslant 2$: as is well known, 2-tape TMs can compute the transpose of an $l \times l$-matrix in $O(l^2 \cdot \log l) = O(n \log n)$ steps. (For a short proof of this fact see [3].) The lower bound argument employs Kolmogorov complexity to enable us to analyze the possible flow of information during the transposition of a suitably chosen matrix on such a machine. (For other lower bound proofs using Kolmogorov complexity see [6, 7, 12]. For a survey of the use of Kolmogorov complexity in lower bound proofs see [8].) This analysis differs from previous lower bound arguments with Kolmogorov complexity by its emphasis on the time-dimension of the computation: it is not enough to watch which information *ever* reaches a certain interval on the worktape, rather it is essential to note which information may be present in such an interval at specific time points. In particular, the argument exploits the fact that in certain situations the same information may have to be brought into the same tape area several times (because after it was first brought there, it had to be overwritten to make space for some other information).

Moreover, the Kolmogorov complexity lemmata (Lemmas 4.1 and 5.10) employ a new trick (from [1]), which allows us to prove optimal bounds for matrix transposition even in the case where the entries of the matrix are single bits. (The technique of [10] could only handle the case with entries of length at least $\log n$. In [3] the results of [10] are extended to entries of all lengths.)

The following notions and definitions are used in this paper. The definition of Turing machines is standard (see, e.g., [5]). A k-tape TM is a TM with k (read/write) worktapes. The worktape alphabet is assumed to be $\{0, 1, B\}$. (If larger worktape alphabets Γ were allowed, the lower bound in this paper would change by the constant factor $1/\log(|\Gamma|)$.) The output tape (if present) is initially blank. It is a two-way write-only tape, i.e., the output tape head can move in both directions but it cannot read. When positioned on some cell on the output tape, the head can write a 0 or a 1 or not write at all. If an output tape cell contains $b \in \{0, 1\}$ at the end of the computation, then b must be written to this cell at least once, may be several times, but no symbol different from b must ever be written to this cell.

Remark 1.1. This restriction on the capabilities of the output tape is slightly more general than the more natural requirement that the output tape head can move only from left to right. Thus, this simpler model is also covered by the proof in this paper. There is an even more general convention for output tapes, namely, where it is permitted to overwrite symbols already written by different symbols. Although it is not clear if it really is stronger, the latter model is not covered by our lower bound proof, as we explicitly use the property that if the output tape head writes a symbol then it is the correct one.

The function Matrix Transposition is induced by the operation of transposing a matrix: given an input $x \in \{0, 1\}^n$, $n = l^2$, regard x as the representation of a Boolean matrix $A \in \{0, 1\}^{l \times l}$ in row-major order, and output the transpose A^{T} in row-major order (or, equivalently, A in column-major order). That means, if the input is $x = b_1 b_2 \ldots b_n$ with $b_m \in \{0, 1\}$, for $1 \leqslant m \leqslant n$, then the output is $y = b_{\pi(1)} b_{\pi(2)} \ldots b_{\pi(n)}$, where the permutation π of $\{1, 2, \ldots, n\}$ is defined by $\pi((i-1) \cdot l + j) = (j-1) \cdot l + i$, for $1 \leqslant i, j \leqslant l$. (A variation of this function was used in [10, 11] for separating two-tape TMs from one-tape off-line TMs without output tape; before that, it had occurred in [13] as an example of a permutation that is hard to realize on devices similar to Turing machines.)

Remark 1.2. For the sake of simplicity, we do not specify Matrix Transposition on inputs of length n, where n is not a square, and ignore such n in the following. It is easy to extend the function Matrix Transposition to nonsquare n, so that both the upper and the lower bound hold for all n. (For example, ignore the last $n - (\lfloor \sqrt{n} \rfloor)^2$ bits of the input.)

The Kolmogorov complexity of a finite binary string is defined as follows. Let an effective coding of all deterministic Turing machines (with any number of tapes) as binary strings be given and assume that no code is a prefix of any other code. Denote the code of a TM M by $\ulcorner M \urcorner$. Then the Kolmogorov complexity of $x \in \{0, 1\}^*$ (with respect to this fixed coding) is $K(x) := \min\{|\ulcorner M \urcorner u| \mid u \in \{0, 1\}^*, M \text{ on input } u \text{ prints } x\}$. A string $x \in \{0, 1\}^*$ is called *incompressible* if $K(x) \geqslant |x|$. (A trivial counting argument shows that for each n there is an $x \in \{0, 1\}^n$ with $K(x) \geqslant n = |x|$.)

The paper is organized as follows: in Section 2 we state the theorem, sketch the proof of the upper bound, and give a detailed outline of the lower bound proof. In Sections 3–6 we prove the lower bound. (The proofs of the Kolmogorov complexity lemmata are given in Section 6.)

2. Main result and outline of the proof

Theorem 2.1. *The time complexity of* MATRIX TRANSPOSITION *on one-tape off-line Turing machines with a one-way output tape is* $\Theta(n^{5/4})$. (*Here* $n = l^2$ *is the length of the input, which is a Boolean* $l \times l$*-matrix given in row-major order.*)

Proof of the upper bound (*sketch*). (This was already noted in [10, 3].) Let the input $x \in \{0, 1\}^{l^2}$ represent the Boolean $l \times l$-matrix $A = (a_{ij})_{1 \leqslant i, j \leqslant l}$, i.e.,

$$x = a_{11} \dots a_{1l} a_{21} \dots a_{2l} \dots a_{l1} \dots a_{ll}.$$

Split A into submatrices A_k, $1 \leqslant k \leqslant l^{1/2}$, where A_k consists of the columns $(k-1) \cdot l^{1/2} + 1, \dots, k \cdot l^{1/2}$ of A. For $k = 1, 2, \dots, l^{1/2}$ compute and output the transpose A_k^{T} of A_k as follows: first, write A_k in row-wise order on the worktape (this takes one sweep over the input, hence $O(l^2)$ steps); then, output A_k column by column (this takes $l^{1/2}$ sweeps over the representation of A_k on the worktape, which consists of $l^{3/2}$ bits, hence $O(l^2)$ steps). Altogether, A is printed column by column, and $O(l^{5/2}) = O(n^{5/4})$ steps are made.

One may ask how the TM orients itself on the input tape so that it is able to pick out those entries of each row of A that constitute A_k. But this is easy, once it has placed markers at regular distances of $l^{1/2}$ cells on the worktape and has constructed one "yardstick" of length l, which is done once and for all at the beginning of the computation. The markers at distance $l^{1/2}$ can be used to measure the length of the rows of A_k and to carry the "yardstick" along during the copying procedure at (almost) no extra cost; the "yardstick" itself is used to measure the distances between the left ends of successive rows of A_k on the input tape. Further, the markers at distance $l^{1/2}$ are used for printing out A_k^{T} row by row. We leave it to the reader to work out the details. □

As the proof of the lower bound is long and quite involved, we will outline its overall structure in the remainder of this section. In the course of this description, we will also indicate the meaning of and motivation for most of the notation used in the formal development of the proof.

The proof is indirect: We fix an incompressible input $x = b_1 \ldots b_n$ of length $n = l^2$ and assume that M makes fewer than $C \cdot n^{5/4} = C \cdot l^{5/2}$ steps on this input, for some constant $C \leqslant 2^{-18}$. The goal is to reach a contradiction. In principle, the whole argument is one big case analysis—each of the cases leads to a contradiction. As some of the cases are trivial, we choose a slightly different way of developing the argument: based on the assumption that fewer than $C \cdot n^{5/4}$ steps are made, we identify more and more features that must be present in the computation. At one point we actually distinguish between two cases (Section 4 versus Section 5) and show that both of them lead to a contradiction.

In the course of identifying more and more properties of the computation of M we introduce more and more notation, and restrict our attention to smaller and smaller sets of input bits (these sets are called $B_1, B_2, B_3, \ldots$). Sometimes the structure identified and given a name is quite natural (e.g., the "printing times" in Definition 3.1 or the concept of "visibility" in Definition 3.6), others may at first seem artificial (e.g., the "early" and "late" bits in Definition 5.3). All the structure we will deal with will concern the position of the worktape head at certain time steps, namely when some output bit is printed or some input bit is read. In the following, descriptions of the notation we use will be set off by paragraphs numbered A, B, C, etc.

(A) (Definition 3.1) Each bit b_m, $1 \leqslant m \leqslant n$, is associated with a *printing time* $t_{\mathrm{pr}}(m)$ (the first step at which b_m is printed to the $\pi(m)$th output tape cell).

(B) (Definition 3.2) Printing times come in clusters: We may identify $\frac{1}{2}l^{3/2}$ disjoint time intervals P_γ (where $\gamma \in G_1$, for some index set G_1 of size $\frac{1}{2}l^{3/2}$), the *printing phases*, so that each of the printing phases has length at most l steps and contains $l^{1/2}$ printing times $t_{\mathrm{pr}}(m)$. The set of those $\frac{1}{2}n$ bits whose printing times lie within one of these P_γ's is called B_1.

Each γ is regarded as a "*color*"; a bit b_m with printing time $t_{\mathrm{pr}}(m)$ in P_γ is also colored with color γ. Thus, B_1 is partitioned into $\frac{1}{2}l^{3/2}$ color classes of size $l^{1/2}$.

(C) (Definition 3.4) Next, we force an additional structure upon the computation: *worktape intervals*. We partition the worktape into disjoint blocks W (of length $4l$) with center V (of length $2l$). If the position of these intervals on the worktape is chosen properly, we may identify a set $G_2 \subseteq G_1$ of colors of size $|G_2| = \frac{1}{4}|G_1|$ so that for each $\gamma \in G_2$ the printing phase P_γ fits within the block structure: during P_γ, the worktape head stays within the center V_γ of one of the blocks W_γ. The set of $\frac{1}{8}l^2$ bits that are printed during these "well-aligned" printing phases is called B_2.

The "buffers" of l cells that separate V_γ from the outside of W_γ play the following role. Since the bits of color γ in B_2 are printed "from V_γ" (i.e., while the worktape head is in V_γ), the information necessary for printing them must in some sense "be contained in" V_γ at the beginning of the printing phase P_γ. Intuitively, if (part of) the information necessary for printing the bits of color γ is not even present in the bigger interval W_γ at a certain time step t before P_γ and cannot be transported into W_γ between t and P_γ by reading these bits off the input tape, then this information must be "carried" across the two "buffers" by worktape head movements, which costs $\Omega(r/\log n)$ steps if r bits of information are to be transported into V_γ.

Next, we must clarify how information about the bits of color γ may reach W_γ. This is the purpose of the following, quite natural, definition.

(D) (Definition 3.6) A bit b_m is *visible* from a worktape interval W at step t if at this step the input tape head scans b_m and the worktape head is in W.

Assume that b_m is a bit in B_2 of color γ. Informally, there are two possibilities for M to get b_m from the input tape to its destination on the output tape:

(α) b_m is visible from W_γ before $t_{\text{pr}}(m)$. (Thus, there is an opportunity to copy b_m from the input tape to some place in W_γ before $t_{\text{pr}}(m)$, so that this information is available when b_m is printed from V_γ during P_γ.)

(β) Otherwise, i.e., b_m is never visible from W_γ before $t_{\text{pr}}(m)$. (Thus, b_m has to be carried into V_γ by movements of the worktape head.)

As either the majority of bits satisfies (α) or the majority of the bits satisfies (β), at least one of the following two cases applies.

Case 1: For at least half the colors in γ at least half the bits in color class γ are treated as in (β).

Case 2: For at least half the colors in γ at least half the bits in color class γ are treated as in (α).

In Section 4 we deal with Case 1, the case of many "underinformed" intervals.

(E) Suppose Case 1 applies. We choose a set $G_3 \subseteq G_2$ of size $\frac{1}{2}|G_2|$ so that for all $\gamma \in G_3$ there are $\frac{1}{2}l^{1/2}$ bits of color γ as in (β). The collection of these $\frac{1}{4}|B_2| = l^2/32$ bits is called B_3.

In Section 4 it is shown via a Kolmogorov complexity argument that this situation entails that M makes $\Omega(l^3/\log n)$ steps. (Of course, this contradicts the initial assumption on the running time of M.) Here, we give a simple informal argument why this lower bound should be expected to hold. Note that here the role of the "buffer" of length l around V_γ within W_γ is evident.

First "pebble argument". Imagine that the input bits b_m are identifiable, atomic objects ("pebbles"), which are to be transported from their original position on the input tape to their final position on the output tape. Whenever the input tape head visits b_m, this bit may be copied to the place on the worktape where the worktape head is positioned. Similarly, whenever the worktape head visits a cell, any bit stored in this cell may be printed to the output cell currently scanned by the output tape head. Finally, the worktape head has the capability to carry bits from one place of the worktape to another; however, it may carry at most $\log n$ bits at the same time. (See Section 1 for the reason for this convention.) Now consider some bit from B_2 that is never visible from its interval W_γ, but printed from V_γ, the central $2l$ cells of W_γ. The following must happen: first, b_m is copied from the input tape to some cell outside of W_γ; then, it is carried by the worktape head across the l cells that separate the outside of W_γ from V_γ; finally, it is printed from V_γ. Overall, the worktape head spends $\Omega(l^3/\log n)$ steps for carrying each of these $l^2/32$ bits across a distance of l cells.

In Section 5 we take care of Case 2, the case of many "overburdened" intervals, which is much more difficult to deal with than Case 1.

(F) If Case 2 applies, we have a set $G_4 \subseteq G_2$ of size $\frac{1}{2}|G_2|$ so that for all $\gamma \in G_4$ there are $\frac{1}{2}l^{1/2}$ bits of color γ as in (α). The collection of these $\frac{1}{4}|B_2| = l^2/32$ bits is called B_4.

We will show that also in this case M makes $\Omega(l^3/\log n)$ steps, contradicting the initial assumption. However, this is not intuitively clear at all: We are dealing with a set of $l^2/32$ bits that all may be copied to some tape interval W_γ and later printed from the center V_γ or W_γ. Why should this cause problems?

We need to introduce more notation.

(G) (Definition 5.2) Each bit b_m in B_4 is associated with a *visibility time* $t_{\text{vis}}(m)$ (which perhaps should more properly be called "last-visibility time"). If b_m has color γ, then $t_{\text{vis}}(m)$ is the last time step before the printing time $t_{\text{pr}}(m)$ at which b_m is visible from W_γ.

Visibility times of bits of the same color are relatively far apart, namely at least l steps. This is the essential consequence of our computational problem being matrix transposition: bits that are closer together than l cells on the output tape have preimages on the input tape that are further than l cells apart. Let us look at the bits in a color class $\gamma \in G_4$ in the order of their visibility times.

(H) First, consider those $\frac{1}{8}l^{1/2} = \frac{1}{4} \cdot \frac{1}{2}l^{1/2}$ bits in the color class whose visibility times come first. They are called the *"early"* bits; the set of all "early" bits is $B_4^{\mathrm{E}} \subseteq B_4$.

The "early" bits may be copied into W_γ at their respective visibility times; but, if this recording is to be of any use they have to be kept stored in this interval over a long period of time, namely at least $\frac{3}{4} \cdot \frac{1}{2}l^{3/2}$ steps, since the printing phase P_γ has only l steps and cannot end before the last of the $\frac{1}{2}l^{1/2}$ visibility times. The reader may already have a vague idea that this may not work well, as the storage capacity of each worktape interval W_γ, measured in bits, is bounded (by $\lceil \log_2 3 \rceil \cdot 4l \leqslant 8l$). However, to really get a handle on this, we must impose another structure on the computation.

(I) (Definition 5.3) For each color $\gamma \in G_4$, we consider the *third quarter* of the $\frac{1}{2}l^{1/2}$ visibility times of bits of color γ in their natural order in time. We call the corresponding bits the *"late"* bits of color γ. The set of all these bits is called B_4^{L}.

There are three properties of the visibility times $t_{\text{vis}}(m)$ of these "late" bits that we will exploit: at $t_{\text{vis}}(m)$,

(i) the visibility times of the $\frac{1}{8}l^{1/2}$ "early" bits of color γ are over (intuitively, these bits should now be stored in W_γ);

(ii) the printing phase P_γ has not yet started;

(iii) the worktape head is in W_γ.

As there are many (namely $|B_4^{\mathrm{L}}| = l^2/128$) late visibility times and the overall computation time is shorter than $l^{5/2} \cdot 2^{-18}$, the "late" visibility times come in large clusters, just as the printing phases.

(J) We may identify $l^{3/2} \cdot 2^{-17}$ disjoint time intervals P'_δ (where $\delta \in D$, for D some index set of size $l^{3/2} \cdot 2^{-17}$), the *visibility phases*, so that each of the visibility phases has length at most $\frac{1}{2}l$ steps and contains $512 \cdot l^{1/2}$ "late" visibility times. (These visibility phases concern *only* last-visibility times of "late" bits.) The set of $l^2/256$ "late" bits b_m in B_4^{L} with visibility time in one of these short visibility phases is called B_5^{L}.

At this point, we may pin down the effect that makes it impossible for M to store all the "early" bits in their respective tape interval W_γ over the long period of time mentioned above. Consider an arbitrary visibility phase P'_δ. The $512 \cdot l^{1/2}$ "late" visibility times within P'_δ belong to bits of different colors, as P'_δ is so short. Further, property (iii) of "late" visibility times from above entails that at most two neighboring tape intervals W_γ and $W_{\gamma'}$ can belong to the colors that occur in the visibility phase. One of the two intervals, which we will call W'_δ, has to handle the majority of the colors, i.e., at least $256 \cdot l^{1/2}$ many. By property (i) and (ii) of "late" visibility times above, there are $256 \cdot l^{1/2} \cdot (\frac{1}{8} l^{1/2}) = 32l$ bits (namely, the "early" bits of colors occurring in P'_δ) whose visibility times come well before P'_δ but whose printing phase starts only after P'_δ. That is, all these bits should be stored in W'_δ at the time of P'_δ. But this cannot work, since the storage capacity of W'_δ is smaller than $8l$.

The situation can be formulated more precisely as follows.

(K) (Definition 5.8 and Lemma 5.9) For each $\delta \in D$ there is a time step t_δ (in P'_δ), a worktape interval W'_δ (with center V'_δ), and a set $B_\delta \subseteq B_4^{\mathrm{E}}$ with $|B_\delta| = 32l$ so that

(a) for all $\delta \in D$ and all $m \in B_\delta$ we have that b_m is printed from V'_δ after t_δ, but b_m is never visible from W'_δ in the time interval $(t_\delta, t_{\mathrm{pr}}(m)]$;

(b) each bit in B_4^{E} occurs in at most $\frac{1}{8} l^{1/2}$ of the B_δ's.

We have just described how our setup leads us to identifying many situations where a tape interval W_γ is "overburdened". We now must combine these many situations in one closed argument that proves the time bound of $\Omega(l^3/\log n)$ we are aiming at. It is easy to combine lower bounds for the time the worktape head spends in different worktape intervals, since these are disjoint. But many of the W'_δ may coincide, and many of the bits in B_4^{E} may occur in many B_δ's. We will once more use our "pebble model" of the TM computation (see above) to intuitively explain why the lower bound should be expected to hold.

Second "pebble argument". We concentrate on one tape interval W and consider $D_0 \subseteq D$ such that $W'_\delta = W$ for all $\delta \in D_0$. In the "pebble model", we may regard the interval W as a box, in which pebbles (the bits) are deposited (at their visibility times) and from which they are retrieved (at their printing times). This box can keep at most $8l$ pebbles at the same time. Once a pebble is deposited in the box, it may be kept there until it is retrieved again, or it may be thrown away. In the latter case, however, we have to pay a "penalty" of $l/\log n$ steps when the pebble is claimed. (This corresponds to the idea that a bit may be kept in W from its visibility time until its printing time, or that it may be erased to make place for other information. In the latter case, it has to be "carried into" V again from outside W, at the cost of $l/\log n$ steps—recall that the worktape head must be assumed to have a storage capacity of $\log n$ bits).

We consider only bits in $B := \bigcup \{ B_\delta \mid \delta \in D_0 \}$. Statements (a) and (b) in (K) translate into the pebble model as follows: For each time step t_δ, there is a set B_δ of pebbles that have been deposited before step δ but not yet retrieved; each B_δ has size at least $32l$. Further, each pebble occurs in at most $\frac{1}{8} l^{1/2}$ of the B_δ's. The dynamics of pebbles entering and leaving the box may be quite complex. Still, we may show via a simple counting argument that the total penalty paid is $\Omega(|D_0| \cdot l^{3/2}/\log n)$ (which is exactly what is needed to obtain the overall bound $\Omega(l^3/\log n)$). Namely, let $\hat{B}$ denote the set of pebbles from B thrown away in the course of the game. It suffices to show that $|\hat{B}|=\Omega(|D_0|\cdot l^{1/2})$. First, note three simple inequalities. Since every B_δ has at least $32l$ elements, we have

$$|D_0|\cdot 32l \leqslant |\{(m,\delta)\mid \delta\in D_0,\ m\in B_\delta\}|.$$

By the capacity constraint on W, for all $\delta\in D_0$ all but $8l$ pebbles from B_δ are thrown away, i.e., $|B_\delta-\hat{B}|\leqslant 8l$; this entails that

$$|\{(m,\delta)\mid \delta\in D_0,\ m\in B_\delta-\hat{B}\}|\leqslant |D_0|\cdot 8l.$$

Since removing one pebble affects at most $\min\{|D_0|,\frac{1}{8}l^{1/2}\}$ many B_δ, we finally have that

$$|\{(m,\delta)\mid \delta\in D_0,\ m\in B_\delta\cap\hat{B}\}|\leqslant |\hat{B}|\cdot\min\{|D_0|,\tfrac{1}{8}l^{1/2}\}.$$

Adding up these three inequalities, we obtain

$$|D_0|\cdot 24l\leqslant |\hat{B}|\cdot\min\{|D_0|,\tfrac{1}{8}l^{1/2}\}.$$

Obviously, this implies that $|\hat{B}|=\Omega(|D_0|\cdot l^{1/2})$ no matter whether $|D_0|$ is smaller or larger than $\frac{1}{8}l^{1/2}$. This finishes the second "pebble argument".

It is easy to see that these estimates for distinct tape intervals W can be combined to yield the overall time bound $\Omega(l^3/\log n)$, which is the desired contradiction for Case 2.

Note that in a rigorous proof we may not use the concept of single, distinguishable bits being stored in an interval W, since the "meaning" of the inscription of a work-tape interval is not accessible to analysis. Instead, we have to exploit the fact that the input is incompressible and find a way to push the argument through with sets of bits that have no individual identity. This is done in Section 5 via another Kolmogorov complexity argument.

3. Printing phases, worktape intervals, and visibility

This and the following three sections contain the details of the proof of the lower bound. In this section, we give some basic definitions and note some basic facts. Sections 4 and 5 contain the analysis of the two main cases.

Fix a one-tape off-line TM M with output tape that computes Matrix Transposition. Choose l large enough (how large l has to be can be seen from the proofs of the Kolmogorov complexity lemmata in Section 6), and fix an incompressible string $x\in\{0,1\}^n$, where $n=l^2$. (Assume for simplicity that $l^{1/2}\cdot 2^{-18}$ is a natural number.) Consider the computation of M on x as input, consisting of, say, T steps. We want to show that $T\geqslant C\cdot l^{5/2}$, for some fixed C (e.g., $C=2^{-18}$). The input $x=b_1b_2\ldots b_n$ represents $A=(a_{ij})_{1\leqslant i,j\leqslant l}\in\{0,1\}^{l\times l}$, where $a_{ij}=b_{(i-1)\cdot l+j}$. The output $y=b_{\pi(1)}b_{\pi(2)}\ldots b_{\pi(n)}$ represents A^{T}.

Definition 3.1 (*Printing times and printing phases*). For $1 \leqslant m \leqslant n$, let

$$t_{\mathrm{pr}}(m) := \text{the first time step at which the output tape head prints a symbol to the } \pi(m)\text{th output cell.}$$

(By the definition of our model, the symbol printed equals b_m; note that $\pi = \pi^{-1}$.) Clearly, $t_{\mathrm{pr}}(m) \neq t_{\mathrm{pr}}(m')$ for $m \neq m'$. Split $\{1, 2, \ldots, T\}$ into $l^{3/2}$ disjoint intervals P_γ (the *printing phases*), so that each P_γ contains exactly $l^{1/2}$ of the *printing times* $t_{\mathrm{pr}}(m)$. Informally, we talk of γ as the "*color*" of printing phase P_γ. The bits b_m whose printing times belong to P_γ inherit the color: if $t_{\mathrm{pr}}(m) \in P_\gamma$, both copies of b_m (the mth input bit and the $\pi(m)$th output bit) are said to have color γ.

We are only interested in *short* printing phases, because of their nice properties given in Lemma 3.3 below. Obviously, if T is to be smaller than $C \cdot l^{5/2}$, then M cannot print too slowly, i.e., there must be many short printing phases. More precisely, we can assume w.l.o.g. that at least $\frac{1}{2} l^{3/2}$ of the P_γ do not last longer than l steps. (Otherwise, M makes at least $\frac{1}{2} l^{3/2} \cdot l = \frac{1}{2} l^{5/2}$ steps, and we are done.) Thus, the following sets are well-defined.

Definition 3.2. Let G_1 denote some subset of $\{1, 2, \ldots, l^{3/2}\}$ of cardinality $|G_1| = \frac{1}{2} l^{3/2}$ so that for all $\gamma \in G_1$ the printing phase P_γ lasts fewer than l steps. Further, let

$$B_1 := \{m \mid t_{\mathrm{pr}}(m) \in P_\gamma \text{ for some } \gamma \in G_1\}$$

denote the set of bits with color in G_1. (It is obvious that $|B_1| = \frac{1}{2} l^2$.)

We will focus on these bits in the following (and regard the other bits as "uncolored"). We list some simple observations.

Lemma 3.3. *Let $\gamma \in G_1$. Then*

(a) *on the output tape, the bits of color γ are contained in an interval of length l;*
(b) *on the input tape, the bits of color γ have distance at least l from one another;*
(c) *during P_γ, the worktape head visits at most l cells.*

Proof. (a) and (c) follow immediately from the fact that P_γ lasts no more than l steps.

(b): We use the fact that M computes Matrix Transposition: Because of (a), and since on the output tape the matrix A is represented in column-major order, all bits of color γ belong to two consecutive columns of A, but no two of them to the same row. Hence these bits are at least l cells apart if A is represented in row-major order. $\square$

We now turn to the tape intervals (of length at most l) that the worktape head scans during different P_γ's. Intuitively, at the beginning of P_γ this interval must contain all the information necessary to print the $l^{1/2}$ bits of color γ. (By Lemma 3.3(b) and (c), at most one bit of color γ on the input tape can be inspected during P_γ. The incompressibility of the input x entails that the other bits of the input inspected during P_γ will not contain any information useful for printing the bits of color γ.) In order to have a clearer picture, we want these tape intervals to be either identical or disjoint for different γ. For technical reasons, we moreover need a "buffer" of length l on each side of these tape intervals. This can be achieved as follows, without reducing the number of useful (colored) bits by more than a constant factor.

Definition 3.4 (*Worktape intervals*). Split the worktape into blocks of l cells each. For $\gamma \in G_1$, let V_γ be an interval consisting of two adjacent blocks such that during P_γ the worktape head is always in V_γ (such an interval exists by Lemma 3.3(c)). Let W_γ be V_γ augmented by the block to the left and the one to the right of V_γ. (W_γ has $4l$ cells.)

We may split the set of all W_γ's into 4 classes such that the W_γ's within each class are disjoint. One of these classes contains the W_γ's for at least one quarter of all $\gamma \in G_1$. Thus, the following sets are well-defined.

Definition 3.5. Let G_2 be a subset of G_1 with $|G_2| = \frac{1}{4}|G_1| = \frac{1}{8}l^{3/2}$ such that for $\gamma, \gamma' \in G_2$ the intervals W_γ and $W_{\gamma'}$ are either disjoint or identical. Let

$$B_2 := \{ m \in B_1 \mid t_{\mathrm{pr}}(m) \in P_\gamma \text{ for some } \gamma \in G_2 \}$$

denote the set of bits with color in G_2. (Obviously, $|B_2| = \frac{1}{4}|B_1| = \frac{1}{8}l^2$.)

We focus on the colors in G_2 and bits in B_2 from here on. Virtually all information needed for printing the bits of color γ are stored in V_γ at the beginning of P_γ. There are several ways for M to get the information about the bits of color γ into V_γ before P_γ. The most natural possibility motivates the following definition.

Definition 3.6 (*Visibility*). Let W be any interval on the worktape, and b_m, $1 \leqslant m \leqslant n$, any input bit. We say that b_m *is visible from* W *at step* t, if at step t the input tape head scans b_m and the worktape head scans a cell in W.

For b_m a bit of color γ, we know that M prints b_m to cell $\pi(m)$ "from V_γ", that is, while the worktape head is in V_γ. Intuitively, it seems reasonable for M to make such bits b_m visible at least from W_γ at some step before $t_{\mathrm{pr}}(m)$, to allow for a "direct" transfer of b_m from the input tape to W_γ and from there to the output tape. For each such bit, there are two cases:

(α) b_m is visible from W_γ before the printing time $t_{\mathrm{pr}}(m)$.

(β) b_m is not visible from W_γ before $t_{\mathrm{pr}}(m)$.

We may use this to distinguish two cases regarding the overall strategy of M: either the majority of the bits b_m in B_2 behave as in (α) or the majority of these bits behaves as in (β). Thus, the following two cases cover all possibilities.

Case 1: For at least half the colors γ in G_2 half the bits in color class γ are treated as in (β).

Case 2: For at least half the colors γ in G_2 half the bits in color class γ are treated as in (α).

Case 1 (which is easier) is treated in Section 4; Case 2 is dealt with in Section 5. Both cases will lead to the conclusion that M makes $\Omega(l^3/\log n)$ steps.

4. The case of many "underinformed" intervals

In this section we assume that Case 1 (see end of Section 3) applies. That is, there are at least $\frac{1}{4}|B_2|=l^2/32$ bits b_m in B_2 so that b_m is never visible from W_γ before $t_{\mathrm{pr}}(m)$, where γ is the color of b_m. We have to show that in this situation M makes $\Omega(l^3/\log n)$ steps. The core of this proof is the following technical lemma, which will be proved later by a Kolmogorov complexity argument.

Lemma 4.1 (The "underinformed" interval). *Let M, l, n, and x be as above, l large enough. Assume that W is an interval of length $4l$ on the worktape, and that V consists of the $2l$ cells in the center of W. Let $r \geqslant \frac{1}{2}l^{1/2}$, and assume that there are r bits b_m that are printed "from V" (i.e., at $t_{\mathrm{pr}}(m)$ the worktape head is in V) but are never visible from W before $t_{\mathrm{pr}}(m)$. Then the worktape head spends at least $r\cdot l/(16\cdot\log n)$ steps in W.*

Proof. See Section 6. $\square$

As we are assuming that Case 1 from above applies, we may make the following definition.

Definition 4.2. Let G_3 be a subset of G_2, with $|G_3|=\frac{1}{2}|G_2|=l^{3/2}/16$, and let B_3 be a subset of B_2 with $|B_3|=\frac{1}{4}|B_2|=l^2/32$ such that for each $\gamma\in G_3$ there are exactly $\frac{1}{2}l^{1/2}$ indices $m\in B_3$ so that b_m has color γ and b_m is never visible from W_γ before $t_{\mathrm{pr}}(m)$.

Then, for each $\gamma\in G_3$ there are at least

$$r_\gamma := |\{\gamma'\in G_3 \mid W_\gamma = W_{\gamma'}\}|\cdot\tfrac{1}{2}l^{1/2}$$

bits b_m that satisfy the hypothesis of Lemma 4.1 with $W=W_\gamma$, $V=V_\gamma$, namely all bits with a color γ' such that $W_{\gamma'}=W_\gamma$. From Lemma 4.1 it follows that M spends at least $r_\gamma\cdot l/(16\cdot\log n)$ steps with the worktape head in W_γ. By summing up these bounds for a family of $\gamma\in G_3$ that form a set of representatives for the equivalence relation over G_3 defined by $W_\gamma=W_{\gamma'}$, we see that M makes at least

$$|G_3|\cdot\tfrac{1}{2}l^{1/2}\cdot l/(16\cdot\log n)=l^3/(512\cdot\log n)$$

steps altogether, which is more than $C\cdot l^{5/2}$, for l large enough. This is the desired contradiction for Case 1.

5. The case of many "overburdened" intervals

In this section we assume that Case 2 (see end of Section 3) applies. That is, many bits are visible from their tape interval before they are printed. In the following definition, we fix one set of such bits.

Definition 5.1. Let G_4 be a subset of G_2, with $|G_4| = l^{3/2}/16$, and let B_4 be a subset of B_2 with $|B_4| = \frac{1}{4}|B_2| = l^2/32$, such that the following is satisfied: for each $\gamma \in G_4$ there are $\frac{1}{2}l^{1/2}$ indices $m \in B_4$ so that b_m has color γ and b_m is visible from W_γ at some step before $t_{\text{pr}}(m)$.

We focus on the $l^2/32$ bits in B_4 and "uncolor" all the other bits and printing phases. As all the bits in B_4 are visible from "their" tape interval, the following definition is quite natural.

Definition 5.2 ([*Last-*] *Visibility times*). For $\gamma \in G_4$ and $m \in B_4$, where b_m has color γ, we let

$$t_{\text{vis}}(m) := \text{the largest } t \leqslant t_{\text{pr}}(m) \text{ such that } b_m \text{ is visible from } W_\gamma \text{ at step } t.$$

Note that the visibility times of bits of the same color are at least l steps apart from each other, by Lemma 3.3(b).

For each color class γ, we consider "early" and "late" visibility times. The basic relation has to be that all "early" visibility times come before all "late" visibility times. These two subsets serve two different purposes: Intuitively, bits with "early" visibility times should be kept stored in W_γ for a long period of time, so they take up storage space in W_γ; on the other hand, "late" visibility times mark time steps at which

(i) bits of color γ with "early" visibility times will not be visible again before P_γ;
(ii) the printing phase P_γ has not yet started;
(iii) the worktape head is in W_γ.

In order to keep the "late" visibility times way before P_γ and to have as many "late" as "early" bits, we declare the first quarter of the visibility times (in their chronological order) as "early", and the third quarter of the visibility times as "late", separately for each color.

Definition 5.3. (a) $B_4^{\text{E}} := \{m \in B_4 \mid t_{\text{vis}}(m) < t_{\text{vis}}(m') \text{ for } \geqslant \frac{3}{4} \cdot (\frac{1}{2}l^{1/2}) \text{ bits } b_{m'} \text{ of the same color } \gamma \text{ as } b_m\}$
(the bits with "early" visibility times),
(b) $B_4^{\text{L}} := \{m \in B_4 \mid t_{\text{vis}}(m') < t_{\text{vis}}(m) \text{ for } \geqslant \frac{1}{2} \cdot (\frac{1}{2}l^{1/2}) \text{ bits } b_{m'} \text{ and } t_{\text{vis}}(m) < t_{\text{vis}}(m') \text{ for } \geqslant \frac{1}{4} \cdot (\frac{1}{2}l^{1/2}) \text{ bits } b_{m'} \text{ of the same color } \gamma \text{ as } b_m\}$
(the bits with "late" visibility times).

Clearly, $|B_4^{\mathrm{E}}| = |B_4^{\mathrm{L}}| = \frac{1}{4}|B_4| = l^2/128$. We are interested in identifying time periods in which many "late" visibility times cluster together. By (i)–(iii) above this immediately leads to a situation where one tape interval W_γ is "overburdened", i.e., should store many more bits than its capacity. Finding such time periods is easy, by a simple averaging argument just like that used for identifying short printing phases (cf. Definitions 3.1 and 3.2).

Definition 5.4 (*Visibility phases*). Partition $\{1, 2, \ldots, T\}$ into $l^{3/2} \cdot 2^{-16}$ disjoint intervals P'_δ, $1 \leqslant \delta \leqslant l^{3/2} \cdot 2^{-16}$, so that each P'_δ contains exactly $512 \cdot l^{1/2}$ time steps $t_{\mathrm{vis}}(m)$ with $m \in B_4^{\mathrm{L}}$. The P'_δ are called the *visibility phases*.

Just as in the case of printing phases there cannot be too many long visibility phases. The total number of steps is at most $C \cdot l^{5/2} \leqslant l^{5/2} \cdot 2^{-18}$; as the visibility phases are disjoint, there can be at most $l^{3/2} \cdot 2^{-17}$ many that are longer than $\frac{1}{2}l$ steps. This is at most half of all visibility phases. Thus, the following set is well-defined.

Definition 5.5. Let D be a subset of $\{1, 2, \ldots, l^{3/2} \cdot 2^{-16}\}$ with $|D| = \frac{1}{2} \cdot (l^{3/2} \cdot 2^{-16}) = l^{3/2} \cdot 2^{-17}$, so that for $\delta \in D$ the visibility phase P'_δ consists of fewer than $\frac{1}{2}l$ steps.

For the rest of the argument, it is crucial that each of the short printing phases P'_δ marks a point in time t_δ and a tape interval W_γ so that W_γ is "overburdened" at step t_δ. The following simple lemma gives the basic reason for this to be true. Afterwards, we develop precise notation for this situation.

Lemma 5.6. (a) *If $m, m' \in B_4^{\mathrm{L}}$ are such that $m \neq m'$ and $t_{\mathrm{vis}}(m)$, $t_{\mathrm{vis}}(m') \in P'_\delta$ for some $\delta \in D$, then b_m, $b_{m'}$ have different colors.*

(b) *If $t_{\mathrm{vis}}(m)$, $t_{\mathrm{vis}}(m') \in P'_\delta$ for some $\delta \in D$, $m, m' \in B_4^{\mathrm{L}}$, and γ, γ' are the colors of b_m, $b_{m'}$, respectively, then W_γ and $W_{\gamma'}$ are either adjacent or identical.*

Proof. Note that during P'_δ both worktape head and input tape head can move at most $\frac{1}{2}l$ cells, and that at $t_{\mathrm{vis}}(m)$ the worktape head is in W_γ and the input tape head scans b_m, and accordingly for $b_{m'}$. By choice of G_2, the intervals W_γ and $W_{\gamma'}$ are either identical, adjacent, or at least $4l$ cells apart; clearly the last alternative is impossible. For (a) recall Lemma 3.3(b). $\square$

In order to have a clearer picture, we want each short visibility phase to correspond to only one worktape interval W_γ. By Lemma 5.6(b), there are at most two such intervals that can be touched during a visibility phase. We choose that one to which the majority of colors occurring in P'_δ belongs.

Definition 5.7. For each $\delta \in D$, we choose $256 \cdot l^{1/2}$ bits b_m with $m \in B_4^{\mathrm{L}}$ and $t_{\mathrm{vis}}(m) \in P'_\delta$ so that all the bits chosen have the same W_γ. We then call this interval W'_δ, its central $2l$ cells V'_δ. The subset of B_4^{L} consisting of all the bits chosen is called B_5^{L}. (Clearly, $|B_5^{\mathrm{L}}| = \frac{1}{2}|B_4^{\mathrm{L}}| = l^2/256$.)

Definition 5.8. For $\delta \in D$, define

(a) $t_\delta :=$ the first step of P'_δ, and

(b) $B_\delta := \{m \in B_4^{\mathrm{E}} \mid \text{for some } m' \in B_5^{\mathrm{L}},\ t_{\mathrm{vis}}(m') \in P'_\delta \text{ and } b_m, b_{m'} \text{ have the same color } \gamma\}$

(the set of "early" bits whose colors "occur" in P'_δ).

The crucial properties of the bits b_m with $m \in B_\delta$ are summarized in the following lemma: at t_δ, all these bits should be "stored" within V'_δ. Lemma 5.9(c) is a technical property; its significance will become clearer below.

Lemma 5.9. (a) *For all $\delta \in D$ and all $m \in B_\delta$ we have that b_m is printed from V'_δ after t_δ, and b_m is never visible from W'_δ in the time interval $(t_\delta, t_{\mathrm{pr}}(m)]$.*

(b) *For all $\delta \in D$ we have that $|B_\delta| = 32l$.*

(c) *Each $m \in B_4^{\mathrm{E}}$ occurs in at most $\frac{1}{8}l^{1/2}$ of the B_δ's.*

Proof. (a): Let $\delta \in D$, $m \in B_\delta$, and let γ be the color of b_m. There is a bit $b_{m'}$, where $m' \in B_5^{\mathrm{L}}$, of color γ with $t_{\mathrm{vis}}(m') \in P'_\delta$, by Definition 5.8(b). It is immediate from the definitions that $V'_\delta = V_\gamma$ and $W'_\delta = W_\gamma$. So we must show that $t_{\mathrm{vis}}(m) < t_\delta < t_{\mathrm{pr}}(m)$. By the definition of "early" and "late" bits, there are at least $\frac{1}{8}l^{1/2}$ visibility times of other bits of color γ between $t_{\mathrm{vis}}(m)$ and $t_{\mathrm{vis}}(m')$ and, hence, by Lemma 3.3(b), $t_{\mathrm{vis}}(m)$ and $t_{\mathrm{vis}}(m')$ are at least $(\frac{1}{4}l^{1/2} - 1) \cdot l$ steps apart. Since $t_{\mathrm{vis}}(m') - t_\delta < \frac{1}{2}l$, this implies that $t_{\mathrm{vis}}(m) < t_\delta$. On the other hand, there are at least $\frac{1}{8}l^{1/2}$ visibility times of other bits of color γ after $t_{\mathrm{vis}}(m')$ and, hence, $t_{\mathrm{vis}}(m')$ precedes the last visibility time $t_{\mathrm{pr}}(m'')$ for some bit $b_{m''}$ of color γ by more than $(\frac{1}{8}l^{1/2} - 1) \cdot l$ steps. Since P_γ has fewer than l steps, $t_{\mathrm{vis}}(m')$ precedes the first step of P_γ, and hence $t_{\mathrm{pr}}(m)$, by at least $(\frac{1}{8}l^{1/2} - 2) \cdot l$ steps. Thus, $t_\delta < t_{\mathrm{pr}}(m)$.

(b): There are $256l^{1/2}$ bits $b_{m'}$, $m' \in B_5^{\mathrm{L}}$, with $t_{\mathrm{vis}}(m') \in P'_\delta$, all of different colors, and $\frac{1}{8}l^{1/2}$ bits b_m, $m \in B_4^{\mathrm{E}}$, belong to each of these colors.

(c): Let γ be the color of b_m. Then there are at most $\frac{1}{8}l^{1/2}$ bits $b_{m'}$ of color γ with $m' \in B_5^{\mathrm{L}} \subseteq B_4^{\mathrm{L}}$, by Definition 5.3(b) and, hence, at most $\frac{1}{8}l^{1/2}$ many δ with $m \in B_\delta$, by Definition 5.8(b). $\square$

The following technical lemma is the core of the argument. It will be proved later, by a Kolmogorov complexity argument.

Lemma 5.10 (The "overburdened" interval). *Let M, l, n, x be as above, l large enough, and let W be an interval of $4l$ cells on the worktape, V the $2l$ cells in the center of W. Let $t_0 < t_1$ be time steps. Let $r \geq 16l$, and assume that there are r bits b_m that are printed "from V" during $(t_0, t_1]$ but are never visible from W after t_0 before being printed from V. Then the worktape head spends at least $r \cdot l/(8 \cdot \log n)$ steps in W during the interval $(t_0, t_1]$.*

Proof. See Section 6. $\square$

The last complication we have to resolve is caused by the fact that there may be many $\delta \in D$ with the same W'_δ. We have already noted in Section 2 that the dynamics of such an "overburdened" interval may be quite complex, because the set of bits that "belong" to this interval, in the sense that their visibility time is over but that they have not been printed yet, changes constantly. The following technical lemma, whose proof is based on Lemma 5.9, resolves this problem. It shows that $\{1, 2, \ldots, T\}$ can be split into sufficiently many *disjoint* time intervals to which Lemma 5.10 can be applied. The crux of this construction is that (a) for each of these disjoint time intervals a sufficiently large set of bits as required in Lemma 5.10 remains (at least $16l = \frac{1}{2}|B_\delta|$ many), and that (b) the total number of such bits, summed over all applications in disjoint time intervals, is proportional to $|\bigcup_{\delta \in D_0} B_\delta| = \Omega(\sum_{\delta \in D_0} |B_\delta| / \min\{|D_0|, l^{1/2}\}) = \Omega(|D_0| \cdot l^{1/2})$. Exactly this is expressed in statements (a) and (b) of the following lemma.

Lemma 5.11. *Let $D_0 \subseteq D$, and let W be such that $W = W'_\delta$ for all $\delta \in D_0$. Let V be the $2l$ cells in the center of W. Then there is an integer q and there are time steps $T = t_0^* > t_1^* > \cdots > t_q^*$ with $t_1^*, \ldots, t_q^* \in \{t_\delta \mid \delta \in D_0\}$ such that for the sets $B_1^*, \ldots, B_q^*$ defined by*

$$B_s^* := \left\{ m \in \bigcup_{\delta \in D_0} B_\delta \;\middle|\; \begin{array}{l} b_m \text{ is printed from } V \text{ in } (t_s^*, t_{s-1}^*] \\ \text{and is not visible from } W \text{ in } (t_s^*, t_{\mathrm{pr}}(m)] \end{array} \right\}$$

we have the following:

(a) $|B_s^*| \geqslant 16l$ *for* $1 \leqslant s \leqslant q$,

(b) $\sum_{s=1}^{q} |B_s^*| \geqslant 128 \cdot |D_0| \cdot l^{1/2}$.

Proof. We define t_s^*, $0 \leqslant s \leqslant q$, by induction on s. Set $t_0^* := T$ and $t_1^* := \max\{t_\delta \mid \delta \in D_0\}$. Clearly, $t_1^* < t_0^*$, and (a) is satisfied for $s = 1$ by Lemma 5.9(b). Now assume as induction hypothesis that $t_1^* > \cdots > t_s^*$ have been defined, that these steps are in $\{t_\delta \mid \delta \in D_0\}$, that (a) is satisfied for $1, \ldots, s$, and that

(c) $\quad |B_\delta - (B_1^* \cup \cdots \cup B_s^*)| < \frac{1}{2}|B_\delta| \quad$ for $\delta \in D_0$, $t_\delta \geqslant t_s^*$.

(This is obviously true if $s = 1$.) We consider two cases.

Case 1: $|B_\delta - (B_1^* \cup \cdots \cup B_s^*)| \geqslant \frac{1}{2}|B_\delta|$ for some $\delta \in D_0$.

Let t_{δ_0} be the maximal t_δ with $\delta \in D_0$ that satisfies this inequality. We let

$$t_{s+1}^* := t_{\delta_0} \in \{t_\delta \mid \delta \in D_0\}.$$

By (c), $t_{s+1}^* < t_s^*$. Since, by the definitions,

$$B_{s+1}^* \supseteq B_{\delta_0} - (B_1^* \cup \cdots \cup B_s^*),$$

we have $|B^*_{s+1}|\geqslant\frac{1}{2}|B_{\delta_0}|\geqslant 16l$, by Lemma 5.9(b). Hence, (a) holds for $s+1$. That (c) is satisfied for $s+1$ follows trivially from the fact that t_{δ_0} was chosen maximal.

Case 2: $|B_\delta-(B^*_1\cup\cdots\cup B^*_s)|<\frac{1}{2}|B_\delta|$ for all $\delta\in D_0$.

We let $q:=s$ and stop the induction. We must check that (b) is satisfied. For this, we define auxiliary sets

$$Y:=\{(m,\delta)\mid m\in B_\delta \text{ and } \delta\in D_0\},$$

$$Y^*:=\{(m,\delta)\mid m\in B^*_1\cup\cdots\cup B^*_s,\ m\in B_\delta,\ \delta\in D_0\}.$$

We know, by Lemma 5.9(c), that each m occurs in at most $\frac{1}{8}l^{1/2}$ many B_δ. Hence

$$(*)\qquad |B^*_1\cup\cdots\cup B^*_s|\geqslant|Y^*|/(\tfrac{1}{8}l^{1/2}).$$

For every $\delta\in D_0$ we have by the assumption $|B_\delta-(B^*_1\cup\cdots\cup B^*_s)|<\frac{1}{2}|B_\delta|$ that

$$|\{(m,\delta)\mid m\in B_\delta,(m,\delta)\in Y^*\}|\geqslant\tfrac{1}{2}|B_\delta|.$$

Adding this inequality up for all $\delta\in D_0$, we get

$$|Y^*|\geqslant\sum_{\delta\in D_0}\tfrac{1}{2}|B_\delta|\geqslant|D_0|\cdot 16l.$$

(The last inequality follows from Lemma 5.9(b)). Substituting this into $(*)$, we obtain

$$|B^*_1\cup\cdots\cup B^*_s|\geqslant|D_0|\cdot l^{1/2}\cdot 128,$$

and this is (b), as desired. $\square$

By applying Lemma 5.10 in the situation of Lemma 5.11, we get the result we need.

Corollary 5.12. *If M, l, n, x are as in Lemma 5.10, and D_0 and W are as in Lemma 5.11, then M spends at least $16\cdot|D_0|\cdot l^{3/2}/\log n$ steps with the worktape head in W.*

Proof. Let q, t^*_s, for $0\leqslant s\leqslant q$, and B^*_s, for $1\leqslant s\leqslant q$, be as in Lemma 5.11. Since (by Lemma 5.11(a)) $|B^*_s|\geqslant 16l$, we can apply Lemma 5.10 to each of the intervals $(t^*_s,t^*_{s-1}]$, for $1\leqslant s\leqslant q$, to conclude that during $(t^*_s,t^*_{s-1}]$ the worktape head spends at least $|B^*_s|\cdot l/(8\cdot\log n)$ steps in W. Since these time intervals are disjoint, we get from Lemma 5.11(b) that altogether M spends at least

$$(128\cdot|D_0|\cdot l^{1/2})\cdot l/(8\cdot\log n)$$

steps with its worktape head in W, as was to be shown. $\square$

Using Corollary 5.12, we can finish the proof of Theorem 2.1. Let $\delta\in D$ be arbitrary. Let $D_0:=\{\delta'\in D\mid W_{\delta'}=W'_\delta\}$, and apply Corollary 5.12 to conclude that the worktape head spends at least

$$|D_0|\cdot 16\cdot l^{3/2}/\log n$$

steps in W'_δ. Summing up these lower bounds for a family of $\delta \in D$ that form a system of class representatives for the equivalence relation on D defined by $W'_\delta = W'_{\delta'}$, we conclude that M makes at least

$$|D| \cdot 16 \cdot l^{3/2}/\log n = l^3/(2^{13} \cdot \log n)$$

steps altogether, and this is certainly larger than $C \cdot l^{5/2}$ for l large enough. This is the desired contradiction for Case 2 from Section 3; thus, the proof of Theorem 2.1 is finished.

6. Proofs of the Kolmogorov complexity lemmata

In this section, we supply the proofs of Lemmas 4.1 and 5.10. We begin with Lemma 5.10; the proof of Lemma 4.1 will be a slight variation of that of Lemma 5.10.

Proof of Lemma 5.10. This is a refinement of an argument in [10]. Let $L(R)$ be the leftmost (rightmost) l cells of W. (So, W is the union of L, V, R.) Choose a cell boundary c_L to the right of a cell in L so as to minimize the number of times in $(t_0, t_1]$ the worktape head crosses this boundary from left to right, and let the number of crossings be $\# C_L$. Similarly, choose a cell boundary c_R to the left of a cell in R that minimizes the number of times in $(t_0, t_1]$ the work head crosses this boundary from right to left, and let the number of crossings be $\# C_R$.

Clearly, the worktape head spends at least $l \cdot (\# C_L + \# C_R)$ steps in L and R taken together (by minimality) and, hence, in W. Thus, it suffices to show that the worktape head enters $[c_L, c_R]$ (the interval between c_L and c_R) at least $r/(8 \cdot \log n)$ times in $(t_0, t_1]$. For this, we describe a method for producing the input x as output of some Turing machine.

Suppose we are given

(i) the program of M, coded as a bitstring in some standard form;

(ii) the contents of $[c_L, c_R]$ at time t_0;

(iii) the number l and the positions of all three tape heads at the first time in $(t_0, t_1]$ at which the worktape head visits $[c_L, c_R]$;

(iv) the position of the input and output tape heads, and the state of M at each time the worktape head crosses c_L or c_R towards V;

(v) the bits b_m that are visible from $[c_L, c_R]$ during $(t_0, t_1]$ but are not printed from $[c_L, c_R]$ during $(t_0, t_1]$ before being visited on the input tape (these bits are given as a single string in the order they are visited by the input tape head);

(vi) the bits b_m of the input that are neither visited by the input tape head during $(t_0, t_1]$ while the worktape head is in $[c_L, c_R]$ nor printed from $[c_L, c_R]$ during $(t_0, t_1]$ (these bits are given in one consecutive string in the order they appear in x);

(vii) the code $\ulcorner M' \urcorner$ for a Turing machine that works as follows. First, simulate the computation of M on input x during all time periods in $(t_0, t_1]$ that the worktape head spends in $[c_L, c_R]$: Starting with an empty tape, using the information given by (ii) and (iii), start simulating M at the first time step in $(t_0, t_1]$ at which the worktape head is in $[c_L, c_R]$. Whenever the (simulated) input tape head visits a cell that has no bit written to it as yet, copy the next bit given by the string described in (v) to this cell, and continue the simulation. Whenever M prints a bit b_m to the $\pi(m)$th cell on the output tape, immediately copy this bit to the corresponding mth cell on the simulated input tape. Whenever the worktape head leaves $[c_L, c_R]$, interrupt the simulation and resume it with the step the worktape head enters $[c_L, c_R]$ again, using the information given by (iv). The simulation is finished when the worktape head leaves $[c_L, c_R]$ for the last time in $(t_0, t_1]$, or when M halts. After this happens, fill in the bits still missing on the input tape, using the string described in (vi). Finally, output the contents of the input tape.

It is clear that the procedure just described outputs x. (Note that here the convention concerning the output tape is used: whenever some symbol is printed to the $\pi(m)$th cell of the output tape, it is equal to the correct mth input bit b_m.) So if we estimate the number of bits needed to code the information described in (i)–(vii), in the form required by the definition of Kolmogorov complexity (see Section 1), we obtain an upper bound for $K(x)$. For the different parts of the string, we get the following estimates:

(i) c_M bits, for some constant c_M;
(ii) $\leqslant 4l \cdot \log 3 \leqslant 8l$ bits;
(iii) $\leqslant 4 \cdot \log n$ bits;
(iv) $(\#C_L + \#C_R) \cdot (2 \cdot \log n + c_M)$ bits;
(v), (vi) $\leqslant l^2 - r$ bits (recall that, by the hypothesis of Lemma 5.10, at least r bits are printed from V before they are visible from W; at least these bits are printed from $[c_L, c_R]$ during $(t_0, t_1]$ before being visited by the input tape head);
(vii) c_0 bits, for some constant c_0.

Furthermore, $O(\log n)$ bits are needed to separate the substrings that belong to (i)–(vii), when concatenated to a single string. (For example, we can precede each of these substrings by its length in binary, with each bit doubled.) We get

$$K(x) \leqslant c_M + 8l + (\#C_L + \#C_R) \cdot (2 \cdot \log n + c_M) + l^2 - r + c_0 + c_1 \cdot \log n.$$

Since x is incompressible, $l^2 \leqslant K(x)$. We get, for l so large that $\log l \geqslant c_M$, and for some constant c'_M:

$$r - 8l - c'_M \cdot \log n \leqslant (\#C_L + \#C_R) \cdot 3 \cdot \log n.$$

By assumption, $r \geqslant 16l$; hence, $r - 8l \geqslant r/2$. This, together with a trivial transformation, yields

$$r \cdot (4 - 8c'_M \cdot \log(n)/r)/(24 \cdot \log n) \leqslant \#C_L + \#C_R.$$

For l so large that $8c'_M \cdot \log n < 16l \leqslant r$ we get

$$r/(8 \cdot \log n) \leqslant \#C_L + \#C_R,$$

as desired. $\square$

Proof of Lemma 4.1. This proof is essentially the same as the previous one, excepting that we simulate the computation of M for *all* time periods in $\{1, 2, \ldots, T\}$ the worktape head spends in $[c_L, c_R]$, and that in (ii) we just need the number of cells between c_L and c_R—the worktape being initially blank. We get the following estimate for $K(x)$:

$$l^2 \leqslant K(x) \leqslant c_M + 2 \cdot \log n + (\#C_L + \#C_R) \cdot (2 \cdot \log n + c_M) + l^2 - r + c_0 + c_1 \cdot \log n;$$

hence, for l large enough, $r - c'_M \cdot \log n \leqslant (\#C_L + \#C_R) \cdot 3 \cdot \log n$. As before, we conclude that $r/(4 \cdot \log n) \leqslant \#C_L + \#C_R$, for l large enough; hence, M spends at least $l \cdot r/(4 \cdot \log n)$ steps in M. □

Acknowledgment

The authors are most grateful to two anonymous referees for their careful reading of the paper and their insightful comments, which were very helpful in improving the exposition of the paper.

References

[1] M. Dietzfelbinger, Lower bounds on computation time for various models in computational complexity theory, Ph.D. Thesis, University of Illinois at Chicago, 1987.

[2] M. Dietzfelbinger, The speed of copying on one-tape off-line Turing machines, *Inform. Process. Lett.* **33** (1989/90) 83–89.

[3] M. Dietzfelbinger, W. Maass and G. Schnitger, The complexity of matrix transposition on one-tape off-line Turing machines, *Theoret. Comput. Sci.* **82** (1991) 113–129.

[4] F.C. Hennie, One-tape off-line Turing machine computation, *Inform. and Control* **8** (1965) 553–578.

[5] J.E. Hopcroft and J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation* (Addison-Wesley, Reading, MA, 1979).

[6] M. Li, L. Longpré, and P.M.B. Vitányi, On the power of the queue, in: A.L. Selman, ed., *Structure in Complexity Theory*, Lecture Notes in Computer Science, Vol. 223 (Springer, Berlin, 1986) 219–233.

[7] M. Li and P.M.B. Vitányi, Tape versus queue and stacks: The lower bounds, *Inform. and Comput.* **78** (1988) 56–85.

[8] M. Li and P.M.B. Vitányi, Kolmogorov complexity and its applications, in: J. van Leeuwen, ed., *Handbook of Theoretical Computer Science, Vol. A* (Elsevier, Amsterdam, 1990) 187–254.

[9] W. Maass, Combinatorial lower bound arguments for deterministic and nondeterministic Turing machines, *Trans. Amer. Math. Soc.* **292** (1985) 675–693.

[10] W. Maass and G. Schnitger, An optimal lower bound for Turing machines with one work tape and a two-way input tape, in: A.L. Selman, ed., *Structure in Complexity Theory*, Lecture Notes in Computer Science, Vol. 223 (Springer, Berlin, 1986) 249–264.

[11] W. Maass, G. Schnitger and E. Szemerédi, Two tapes are better than one for off-line Turing machines, in: *Proc. 19th STOC* (1987) 94–100.

[12] W. Paul, On-line simulation of $k+1$ tapes by k tapes requires nonlinear time, *Inform. and Control* **53** (1982) 1–8.

[13] W. Paul and H.-J. Stoß, Zur Komplexität von Sortierproblemen, *Acta Inform.* **3** (1974) 217–225.

[14] P.M.B. Vitányi, Square time is optimal for the simulation of a pushdown store by an oblivious one-head unit, *Inform. Process. Lett.* **21** (1985) 87–91.