Levin–Kolmogorov Complexity is not in Linear Time

Nicholas Brandt

Department of Computer Science
ETH Zurich
Zurich, Switzerland
nicholas.brandt@inf.ethz.ch

Understanding the computational hardness of Kolmogorov complexity is a central open question in complexity theory. An important notion is Levin's version of Kolmogorov complexity, Kt, and its decisional variant, MKtP, due to its connections to universal search, derandomization, and oneway functions, among others. The question whether MKtP can be computed in polynomial time is particularly interesting because it is not subject to known technical barriers such as algebrization or natural proofs that would explain the lack of a proof for $\mathsf{MKtP} \notin \mathsf{P}$.

We take a significant step towards proving $\mathsf{MKtP} \notin \mathsf{P}$ by developing an algorithmic approach for showing *unconditionally* that $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathcal{O}(n)]$ cannot be decided in deterministic linear time in the worst-case. This allows us to partially affirm a conjecture by Ren and Santhanam [RS22] about a non-halting variant of Kt complexity. Additionally, we give *conditional* lower bounds for MKtP that tolerate either more runtime or one-sided error.

Contents

1 Introduction

The formal concept of “complexity” was spearheaded in the 1960’s by Solomonoff [Sol60; Sol64a; Sol64b], Kolmogorov [Kol63; Kol65], and Chaitin [Cha66; Cha69]. In particular, ideas and techniques from meta-complexity—the computational hardness of complexity—have diffused into adjacent subfields like learning theory, derandomization and even cryptography. The recent years have seen a flurry of works on meta-complexity ([ABK$^+$06; Hir18; Oli19; GII$^+$19; Ila20a; Hir20c; Ila20b; Hir20a; Hir20b; LOS21; RS21; HN22; Hir22a; MP24] to name only a few). We refer to Trakhtenbrot [Tra84] for a historical survey of complexity and to the more recent survey by Allender [All21].

In this work we focus on Levin’s notion of Kolmogorov complexity Kt [Lev84], which elegantly incorporates a time bound and thus evades the undecidability of the original Kolmogorov complexity. The Levin–Kolmogorov complexity of a given string x is the minimum over all programs that produce x of the sum of the program’s length plus the logarithm of its runtime, i.e., $\mathrm{Kt}(x) = \min_{\Pi \mapsto x}(|\Pi| + \lceil \log_2(t) \rceil)$ where Π computes the string x in time t. Its decisional problem is defined as $\mathsf{MKtP} \coloneqq \{(x,k) \mid \mathrm{Kt}(x) \leq k\}$. For an in-depth introduction to meta-complexity problems we refer the reader to [LV08].
Since it is significant for this work of technical reasons, we note that in our notion of Kt complexity the time t measures the number of steps of the program Π and *not* the steps of the universal machine executing Π. This issue is mood (i.e., both definitions are equivalent) in any machine model that has a linear universal simulation overhead, e.g. random-access machines. Because such register machines model physical computers much more closely than e.g. (tape-based) Turing machines, our restriction on such linear-overhead models should not be taken as a fundamental drawback of our approach but rather as a technical curiosity of tape machines. We expound on this robustness issue in Section 3.

In fascinating works Liu and Pass [LP20; LP21b] uncover a surprising connection between derandomization and the existence of oneway functions (OWF) through Kt complexity. On the one hand, they show that (weak) derandomization $\mathsf{BPP} \neq \mathsf{EXP}$ is equivalent to the *zero*-sided average-case hardness of MKtP, and on the other that the existence of OWFs is equivalent to the *two*-sided average-case hardness of MKtP. One-way functions are central to modern cryptography: they characterizes *symmetric* cryptography, dubbed “Minicrypt” by Impagliazzo [Imp95]. They are necessary and sufficient for: digital signatures [Rom90], (cryptographic) pseudorandom generators [BM82; HIL$^+$99], pseudorandom functions [GGM84], private-key encryption [GM84], commitment schemes [Nao91] and much more. Moreover, the existence of OWFs is itself equivalent to the hardness of many other meta-complexity problems (see at the end of Section 2).

These results add to the importance of understanding the hardness of Kt complexity. Unfortunately, only a comparatively weak *unconditional* lower bound for Kt complexity is known. Namely, Hirahara [Hir20b] shows that the Kt-random strings $R_{\mathrm{Kt}} \coloneqq \{x \mid \mathrm{Kt}(x) \geq |x|\}$ are immune[1] to the circuit class P-uniform ACC^0. Now, one might ask:

Why are there no stronger lower bounds for Kt complexity?

In fact the EXP-completeness of MKtP under BPP [LP21b] explains why there is no worst-case lower bound against probabilistic polynomial-time algorithms (BPP); because it would imply $\mathsf{BPP} \neq \mathsf{EXP}$ which itself is subject to the relativization barrier [BGS75]. In the face of this barrier we might ask about a weaker worst-case lower bound against a deterministic polynomial-time algorithms (P). Interestingly, whether $\mathsf{MKtP} \notin \mathsf{P}$ (mentioned e.g. in [Oli19; Hir20b]) remains open at least since Allender, Buhrman, Koucký, van Melkebeek, and Ronneburger [ABK$^+$02] posed it explicitly in 2002. This is particularly unsatisfactory because the result $\mathsf{MKtP} \notin \mathsf{P}$ is *not* subject to technical barriers like algebrization [AW08; IKK09; AB18] or natural proofs [RR97].

[1] No infinite subset of R_{Kt} is in P-uniform ACC^0.

2 Contributions & Related Work

Our main contribution is a significant step towards unconditionally proving $\mathsf{MKtP} \notin \mathsf{P}$.

Theorem 1. *The Levin–Kolmogorov complexity cannot be decided in deterministic linear time in the worst-case, i.e.,* $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathcal{O}(n)]$.

On the $\widetilde{\mathrm{Kt}}$ *notion of Ren and Santhanam.* Because our lower bound relativizes we can partially affirm a conjecture (Open Problem 4.7.) by Ren and Santhanam [RS22]. They introduce a "non-halting" variant $\widetilde{\mathrm{Kt}}$ of Levin–Kolmogorov complexity whose definition is almost identical to the standard Kt complexity except that the witness program producing a given string need not halt after writing the string on its tape. Ren and Santhanam conjecture that—despite their close definitions—the two notions behave quite differently in that infinitely many strings x have $\widetilde{\mathrm{Kt}}(x) \lneq \mathrm{Kt}(x)$. By analyzing the proof of Theorem 1 we can give a concrete example affirming their conjecture. Concretely, infinitely many prefixes of Chaitin's constant Ω have $\widetilde{\mathrm{Kt}}(\Omega_1||...||\Omega_\ell) <_{\mathsf{io}} \mathrm{K}(\Omega_1||...||\Omega_\ell) \leq \mathrm{Kt}(\Omega_1||...||\Omega_\ell)$. To see this assume the opposite (all-but-finitely many prefixes have $\widetilde{\mathrm{Kt}}(\Omega_1||...||\Omega_\ell) \geq_{\mathsf{abf}} \mathrm{Kt}(\Omega_1||...||\Omega_\ell)$), then our proof of Theorem 1 allows us to prove the linear-time hardness of $\widetilde{\mathrm{Kt}}$ relative to any oracle. However, Ren and Santhanam [RS22] already give an oracle relative to which $\widetilde{\mathrm{Kt}}$ is computable in linear time. Pushing the limits of our technique we find $\widetilde{\mathrm{Kt}}(\Omega_1||...||\Omega_\ell) \leq_{\mathsf{io}} \mathrm{Kt}(\Omega_1||...||\Omega_\ell) - \Theta(\ln\ln(\ell))$ falling short of $\widetilde{\mathrm{Kt}}(\Omega_1||...||\Omega_\ell) \leq_{\mathsf{io}} \mathrm{Kt}(\Omega_1||...||\Omega_\ell)/\,\Theta(1)$ as required by Ren and Santhanam in their full conjecture.

Comparison to Hirahara's Kt *lower bound.* Hirahara [Hir20b] shows an incomparable unconditional lower bound for Kt complexity, namely, that the Kt-random strings R_{Kt} are immune to P-uniform ACC^0 (see [All21] for a nice description of Hirahara's approach). Compared to Hirahara's immunity lower bound (no infinite subset can be decided), our result is weaker in that it only provides worst-case hardness (no algorithm can decide correctly for *every* string). On the other hand, our lower bound holds against deterministic linear time $\mathsf{DTIME}[\mathcal{O}(n)]$ which—we argue—is closer to P than the rather weak circuit class P-uniform ACC^0 for which Hirahara's lower bound holds. The only case in which our result would be subsumed by [Hir20b] in the implausible case that $\mathsf{P} = \mathsf{P}$-uniform ACC^0 which would imply that R_{Kt} is immune to P which in turn is stronger than $\mathsf{MKtP} \notin \mathsf{P}$.
Lastly, our proof strategy differs conceptually from the one in [Hir20b]. The approach of Hirahara is based on the "algorithmic method" of Williams [Wil13; Wil14] where a nontrivial satisfiability algorithm for a circuit class yields a lower bound against that class. Obtaining a stronger immunity of R_{Kt} using the Hirahara–Williams approach is equivalent to satisfiability algorithms for stronger circuit classes which may be subject to known barriers such as algebrization [AW08; IKK09; AB18] or natural proofs [RR97]. In comparison, our approach opens new avenues for improved lower bounds that possibly evade these barriers. See Section 3 for a discussion of the limitations of our technique and possible ways to overcome them.

Stronger conditional bounds. We give conditional lower bounds which either tolerate larger runtime or one-sided error.

Theorem 2. *For each time bound $\mathfrak{t}(n) \geq n$ at least one of the following is true:*

1. $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathfrak{t}]$,
2. $\mathsf{MKtP} \notin \mathsf{Heur}_{\gamma_{\mathsf{fp}},\gamma_{\mathsf{fn}}}\mathsf{DTIME}[\mathcal{O}(n)]$ *with no false positive error* $\gamma_{\mathsf{fp}}(n) := 0$ *and false negative error* $\gamma_{\mathsf{fn}}(n) := {}^{1}\!/_{2n\mathfrak{t}(2n)} - {}^{2}\!/_{2^n}$,

More related work. To contextualize our lower bound for MKtP we list some related notions of complexity and their lower bounds.

The canonical time-bounded variant $\mathsf{MK^tP}$ [Kol63; Sip83; Har83; Ko86] of Kolmogorov complexity is parameterized over some time bound t and limits the witness program of a given string x to run in time at most $\mathsf{t}(|x|)$. Limiting the witness program's runtime makes this notion computable, opposed to standard Kolmogorov complexity. For exponential time bounds t Hirahara [Hir20b] shows that $\mathsf{MK^tP}$ is EXP-complete under ZPP reductions and even that the set of K^{t}-random strings is immune to P (no infinitely large subset of K^{t}-random strings is in P).

Allender, Buhrman, Koucký, van Melkebeek, and Ronneburger [ABK+06] show that the Levin–Kolmogorov complexity MKtP is EXP-complete under $\mathsf{P/poly}$ or NP reductions, i.e., $\mathsf{MKtP} \in \mathsf{P/poly} \iff \mathsf{EXP} \subseteq \mathsf{P/poly}$. Liu and Pass [LP21b] improve this to BPP reductions, i.e., $\mathsf{MKtP} \in \mathsf{BPP} \iff \mathsf{EXP} = \mathsf{BPP}$. Thus, any nontrivial derandomization $\mathsf{BPP} \neq \mathsf{EXP}$ is equivalent to a lower bound $\mathsf{MKtP} \notin \mathsf{BPP}$ against bounded-error probabilistic TMs. In turn, this means that any barrier preventing us from proving $\mathsf{BPP} \neq \mathsf{EXP}$ also prevents us from proving the randomized lower bound $\mathsf{MKtP} \notin \mathsf{BPP}$. In contrast, our lower bound $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathcal{O}(n)]$ is much weaker both in the quantitative runtime (linear vs. polynomial) as well as the computational model (deterministic vs. probabilistic)—and thus evades known barriers.

Oliveira [Oli19] introduces rKt—a randomized version of Levin–Kolmogorov complexity—where the witness program of a given string x must produce that string x on at least a 2/3-fraction of randomnesses. This randomized complexity is BPEXP-complete and Oliveira shows hardness of his notion against quasipolynomial time bounded-error TMs, i.e., $\mathsf{MrKtP} \notin \mathsf{BPTIME}[n^{\log(n)^{\Theta(1)}}]$. Later Hirahara [Hir22b] improves that bound to $\mathsf{GapMrKtP} \notin \mathsf{io\text{-}BPTIME}[2^{\epsilon n}]$ for any $\epsilon \gneq 0$. Oliveira [Oli19] also gives a potential avenue toward proving $\mathsf{MKtP} \notin \mathsf{P}$ via the implication $\mathsf{MrKtP} \in \mathsf{Promise\text{-}EXP} \implies \mathsf{MKtP} \notin \mathsf{P}$.

For a nondeterministic NEXP-complete complexity notion KNt Allender, Koucký, Ronneburger, and Roy [AKR+11] show unconditionally that the set of KNt-random strings is not in $\mathsf{NP} \cap \mathsf{co\text{-}NP}$.

The canonical problem for circuit complexity is nowadays called the minimum circuit size problem (MCSP) [KC00]. It has been previously considered by Trakhtenbrot [Tra84] (Task 4), and Levin reportedly delayed the publication of his work on NP-completeness to include MCSP. Since $\mathsf{MCSP} \in \mathsf{NP}$ an unconditional lower bound seems unlikely; the question is rather whether MCSP is NP-complete which is related to major open questions in theoretical computer science. We refer the interested reader to [All21; AIV21] and references therein for more details about the NP-completeness of MCSP.

Oliveira, Pich, and Santhanam [OPS19] give "hardness magnification" results for gap versions of MKtP and MCSP. They establish that slightly improved lower bounds for these problems can be "magnified" to strong lower bounds. The reason why we cannot use their result to magnify our linear-time lower bound is a difference in the parameter regime (similar to [RS22]). They consider the hardness of distinguishing strings of low complexity from string of even lower complexity (e.g. $n^\epsilon + \Theta(\log n)$ vs. n^ϵ). On the other hand, we crucially use the fact (as our counter assumption) that we are able to exactly compute the complexity of a given string $x \in \{0,1\}^n$ even when $\mathrm{Kt}(x) \approx n$.

Huang, Ilango, and Ren [HIR23] show unconditional hardness of an oracle variant of the minimum circuit size problem (MOCSP) using a cryptographic tool called witness encryption [GGS+13].

Connection to oneway functions. In recent years there has emerged a research effort to characterize oneway functions (OWF) by the hardness of meta-complexity problems. As an incomplete list: OWFs are equivalent to the mild two-sided hardness of $\mathsf{MK^tP}$ [LP20], the two-sided hardness of MKtP [LP21b], the two-sided hardness[2] of an (NP-complete) conditional variant McKTP [ACM+21] of Allender's KT complexity [All01], the mild two-sided hardness of (parameterized versions of) $\mathsf{MK^tP}$ against sublinear time over a smooth range of parameters [LP21a], the mild average-case hardness of the probabilistic

[2] Here, the error probabilities are not equal for both directions.

$\mathsf{MpK^tP}$ (introduced in [GKL$^+$22]) for polynomial t [LP23b], the *worst-case* hardness of a promise version of $\mathsf{MK^tP}$ (with small computational depth) [LP23a], the hardness of a distributional variant of Kolmogorov complexity under the assumption $\mathsf{NP} \not\subseteq \mathsf{io\text{-}P/poly}$ [Hir23].

3 Technical Overview

A natural approach to proving lower bounds for a given meta-complexity problem is to assume that the problem is easy and then leverage an efficient solver for that problem to quickly construct a highly complex string (w.r.t. to the given complexity measure). The historical proof of the undecidability of standard Kolmogorov complexity as well as Hirahara's much more sophisticated lower bound for Kt complexity [Hir20b] are instantiations of this approach.

To directly apply this approach to Kt complexity it is useful to define what we call the "critical threshold" $\theta_{\Pi,t} := |\Pi| + \lceil \log_2(t) \rceil$ of a given TM Π after t steps of its execution. We will assume that the decision problem MKtP can be worst-case decided by a TM Π_{Kt} in linear time. Then we construct a TM (using Π_{Kt} as a subroutine) that quickly outputs a Kt-random string. To reach a formal contradiction, our TM $\Pi_{\lightning}$ must in t steps produce a Kt-random string z that is strictly longer than the critical threshold $\theta_{\Pi_{\lightning},t}$, i.e., $\theta_{\Pi_{\lightning},t} \geq \mathrm{Kt}(z) \geq |z| \gneq \theta_{\Pi_{\lightning},t}$ where the first inequality is by the definition of Kt complexity and the fact that Π outputs z in t steps, the second inequality is the Kt-randomness of z, and the last inequality is by assumption.

Black-box barrier. A conceptual problem to the algorithmic approach for a lower bound for MKtP is that we know little about the *structure* of the Kt-random strings $R_{\mathrm{Kt}} := \{x \mid \mathrm{Kt}(x) \geq |x|\}$. We say a TM Π_{BB} yields a contradiction in a *black-box* way, if given access to *any* set of potentially Kt-random strings $R \neq \{0,1\}^*$ it produces a string $z \notin R$ in t steps such that $\theta_{\Pi_{\mathsf{BB}},t} \lneq |z|$. Intuitively, a potential TM Π_{BB} ignores the structure of the set R since it works for any arbitrary R. Clearly, such a Π_{BB} cannot exists because we can define $R_{\Pi_{\mathsf{BB}}} := \{0,1\}^* \setminus \{z \mid \Pi_{\mathsf{BB}}$ queries z to its oracle or outputs z in t steps and $\theta_{\Pi_{\mathsf{BB}},t} \lneq |z|\}$ that breaks Π_{BB}. So, for the algorithmic approach to succeed we need to exploit some structure of the actual set of Kt-random strings R_{Kt}. Before we explain how, let us first present our rather simple strategy for a TM $\Pi_{\lightning}$.

Our search strategy. As a first step we use the length-monotonic depth-first-search described in Fig. 1. The high-level idea is to traverse the binary tree of finite strings starting with the string 0. Whenever the i-th string z_i is visited our search algorithm TRAVERSE queries z_i to its oracle R_{Kt} and if $z_i \in R_{\mathrm{Kt}}$ descends to the next length with $z_{i+1} := z_i||0$ (the left child of z_i), otherwise it continues with the lexicographically next string of the same length $z_{i+1} := \mathrm{next}(z_i)$ (the right neighbor of z_i). See Fig. 2 for an exemplary run of TRAVERSE. Crucially, the length of the visited strings is non-decreasing. We note that our TRAVERSE algorithm doesn't terminate and hence does not suffice for a proper contradiction (even if it visits a Kt-random string quickly enough). To actually reach a contradiction we have to a) construct a TM Π_{TRA} implementing TRAVERSE that at some point visits a string $\check{z}$ within $\hat{t}$ steps s.t. $\theta_{\Pi_{\mathsf{TRA}},\hat{t}} \lneq |\check{z}|$, and b) implement a mechanism s.t. Π_{TRA} also recognizes this fact—so that it can terminate and output $\check{z}$.

As a stepping stone it will be useful to see that TRAVERSE visits infinitely many different strings $(z_i)_{i\in\mathbb{N}}$. This follows from the existence of at least one Kt-random string of each length on which TRAVERSE descends to the next length. Moreover, we observe that TRAVERSE never "wraps around". That is TRAVERSE never reaches an all 1s string at the right border of the binary tree. Assuming an infinite (1-random) string s whose every prefix is Kt-random, this is also easy to see. Whenever TRAVERSE visits a prefix $z_i = s_1||...||s_\ell$ it descends to the next string $z_{i+1} := s_1||...||s_\ell||0$—thus always staying "to the left" of the infinite string s in the binary tree. Glossing over a minor technical issue, we can take Chaitin's constant Ω (encoded in binary) to be such an infinite 1-random string. We remark that the

```
TRAVERSE
$z_1 := 0 \in \{0,1\}^*$
for $i \in \mathbb{N}_{\geq 1}$
    if $\mathrm{Kt}(z_i) \geq |z_i|$
        $z_{i+1} := z_i||0$
    else
        $z_{i+1} := \mathrm{next}(z_i)$
    endif
endfor
```

Fig. 1: Our traversal algorithm (simplified).

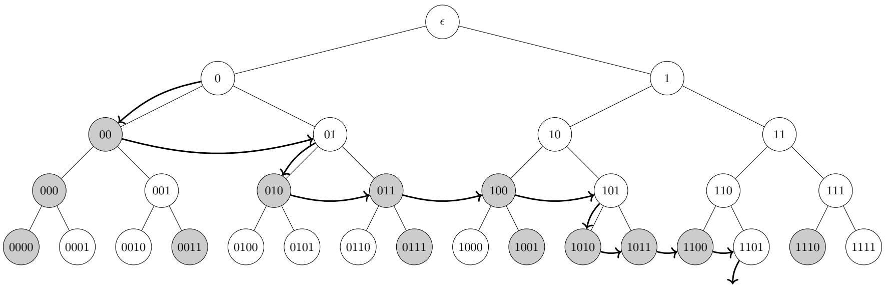

Fig. 2: Exemplary run of TRAVERSE: white strings are Kt-random.

1-randomness of Ω is the crucial information about the actual set of Kt-random strings R_{Kt} that allows our TRAVERSE algorithm to sidestep the aforementioned black-box impossibility.

Analysis. Next, we analyze the behavior of TRAVERSE to prove that after some $\hat{t}$ steps TRAVERSE visits some Kt-random string $z_{\hat{\imath}}$ s.t. $\theta_{\Pi_{\mathsf{TRA}},\hat{t}} \lneq |z_{\hat{\imath}}|$. Let $Z := \{z_i \mid i \in \mathbb{N}\}$ be the set of visited strings. Let $i_\ell := |Z \cap \{0,1\}^{\leq \ell}|$ be the number of strings visited of length at most ℓ. Let $Z_\ell := Z \cap \{0,1\}^\ell = \{z_{i_{\ell-1}}||0, ..., z_{i_\ell}\}$ be the set of visited strings of length exactly ℓ. Let $S_\ell := \{z \in \{0,1\}^\ell \mid \mathrm{int}(z) \gneq \mathrm{int}(z_{i_\ell})\} \subset \{0,1\}^\ell$ be the lexicographical successors of Z_ℓ (the right neighbors of Z_ℓ). Now, note that because TRAVERSE doesn't wrap around, it holds that $Z_{\ell+1} \mathbin{\dot\cup} S_{\ell+1} = (\{z_{i_\ell}\} \mathbin{\dot\cup} S_\ell)||\{0,1\}$ and thus $|Z_{\ell+1}| + |S_{\ell+1}| = 2|S_\ell| + 2$. Let $\gamma_\ell := {}^{|Z_\ell|}/_{|Z_\ell \dot\cup S_\ell|}$ be the fraction of strings of length ℓ that TRAVERSE actually visits to the strings that it could potentially visit. By recursion the number of of visited strings of length ℓ can be expressed as $|Z_\ell| = \gamma_\ell \sum_{\kappa=1}^{\ell} 2^\kappa \prod_{i=\ell-\kappa+1}^{\ell}(1-\gamma_i)$. For our approach we'd like i_ℓ and thus $|Z_\ell|$ to be asymptotically small. An informal argument for this is that the formula for $|Z_\ell|$ expresses a "self-limiting" behavior that emerges from our TRAVERSE algorithm. Namely, the faster γ_i goes to 0 the smaller $|Z_\ell|$ because $|Z_\ell|$ depends linearly on γ_ℓ. On the other hand, $|Z_\ell|$ depends on the product $\prod_{i=j-\kappa+1}^{\kappa}(1-\gamma_i)$ which is closer to 1 the faster γ_i goes to 0. These antagonistic influences suggest there is some asymptotic rate of γ_i that leads to an asymptotically maximal $|Z_\ell|$. This behavior of $|Z_\ell|$ can also be captured informally from the algorithmic view of TRAVERSE. Whenever $|Z_{\ell-1}|$ is large this means that TRAVERSE moves far to the right, forcing the next $|Z_\ell|$ to be small because only few strings remain to the right. In this manner, there cannot be many successive $|Z_j|$ that are large. Turning back to the more formal analysis, we can bound the number of visited strings of length at most ℓ by

$$\begin{aligned} i_\ell &:= \sum_{j=1}^{\ell} |Z_j| && (1) \\ &= \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa \prod_{i=j-\kappa+1}^{j} (1-\gamma_i) && (2) \\ &\leq \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa e^{\sigma_{j-\kappa}-\sigma_j} && (3) \end{aligned}$$

where $\sigma_j := \sum_{i=1}^{j} \gamma_i$. Using the following technical lemma we can bound this quantity.

Lemma 1 (Infinitely-often bound). *For any sequence* $(\gamma_j)_{j\in\mathbb{N}}$ *with* $\gamma_j \in [0,1]$ *and* $\sigma_\ell := \sum_{i=1}^{\ell} \gamma_i$ *it holds that infinitely often* $\sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^{\kappa} e^{\sigma_{j-\kappa}-\sigma_j} \leq_{\mathsf{io}} {}^{2^\ell}/_{\ell \ln(\ell)}$.

The rigorous proof of Lemma 1 is contained in Section 6. This means that for infinitely many "critical" lengths $\hat{\ell}$ when TRAVERSE visits the last string $z_{i_{\hat{\ell}}} \in \{0,1\}^{\hat{\ell}} \cap R_{\mathrm{Kt}}$ it took at most $\mathcal{O}(i_{\hat{\ell}} \cdot \hat{\ell})$ steps to do so—assuming $\mathsf{MKtP} \in \mathsf{DTIME}[\mathcal{O}(n)]$. Recall that for any length ℓ it holds that $z_{i_\ell} \in R_{\mathrm{Kt}}$ because z_{i_ℓ} is the last string of length ℓ that TRAVERSE visits from which it descends to the next length. Now, if TRAVERSE were to output any such $z_{i_{\hat{\ell}}}$, then we would reach the contradiction

$$\hat{\ell} \leq \mathrm{Kt}\left(z_{i_{\hat{\ell}}}\right) \leq |\mathit{\Pi}_{\mathsf{TRA}}| + \left\lceil \log_2\left(\mathcal{O}\left(i_{\hat{\ell}} \cdot \hat{\ell}\right)\right)\right\rceil \leq \hat{\ell} - \ln\ln\left(\hat{\ell}\right) + \mathcal{O}(1)\ . \tag{4}$$

The missing piece is hence to construct a TM $\mathit{\Pi}'_{\mathsf{TRA}}$ that implements TRAVERSE such that it is aware of its own critical threshold—so it knows when to output a string $z_{i_{\hat{\ell}}}$. A generic approach is to simply simulate TRAVERSE with a $\mathcal{O}(n\ln(n))$ slowdown. However, this would result in

$$\hat{\ell} \leq \mathrm{Kt}\left(z_{i_{\hat{\ell}}}\right) \leq |\mathit{\Pi}_{\mathsf{TRA}}| + \left\lceil \log_2\left(\mathcal{O}\left(i_{\hat{\ell}} \cdot \hat{\ell} \cdot \ln\left(i_{\hat{\ell}} \cdot \hat{\ell}\right)\right)\right)\right\rceil \leq \hat{\ell} + \ln\left(\hat{\ell}\right) - \ln\ln\left(\hat{\ell}\right) + \mathcal{O}(1) \tag{5}$$

which does not suffice for a contradiction. Hence, we let $\mathit{\Pi}'_{\mathsf{TRA}}$ count the size $|Z_\ell|$ not one-by-one but only once it reaches a Kt-random string (the last string of each length). This way, each length ℓ incurs an additive runtime overhead of $\mathcal{O}(\ell)$ instead of $\Omega(|Z_\ell|\ell\ln(\ell))$. Due to space restrictions and because the details of the step-counting don't provide much conceptual insight, we defer these details to the formal proof of Theorem 1.

On Lemma 1. In the previous paragraph we have bounded the runtime of our contradicting TM $\mathit{\Pi}'_{\mathsf{TRA}}$ in terms of the number of visited strings i_ℓ which in turn can be bounded by the term in Lemma 1. As alluded to earlier the specific term in Lemma 1 arises from the "self-limiting" behavior of our TRAVERSE algorithm. Recall that on the binary tree TRAVERSE only moves to the right neighbor or the left child of the current string. Fix some length $\hat{\ell}$. If for many of the previous lengths $\ell \lneq \hat{\ell}$ the TM TRAVERSE visited few strings, then the number $i_{\hat{\ell}}$ of visited strings at length $\hat{\ell}$ will be small by definition. On the other hand, if TRAVERSE visits many strings of length ℓ, then TRAVERSE moves farther to the right, leaving fewer strings of subsequent lengths to be potentially visited. With this intuition in mind, it remains to prove Lemma 1 formally, though we defer the rigorous proof of Lemma 1 to Section 6. Instead, here we give a superficial sketch of our proof.

The basic idea is to prove Lemma 1 by contradiction, hence we may assume

$$\sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^{\kappa-\ell} e^{\sigma_{j-\kappa}-\sigma_j} \geq {}^{1}/_{\ell\ln(\ell)} \tag{6}$$

for all $\ell \in \mathbb{N}$. We sum this inequality and bound the inner sum on the left-hand side of Eq. (6) by $2^{j-\ell+1}$ (using the trivial inequality $\sigma_j - \sigma_{j-\kappa} \geq 0$) to obtain a first lower bound for σ_d, i.e.,

$$2\sigma_d \geq \sum_{\ell=1}^{d}\sum_{j=1}^{\ell} \gamma_j 2^{j-\ell+1} \geq 2\sum_{\ell=1}^{d} \frac{1}{\ell\ln(\ell)} \approx \int \frac{1}{d\ln(d)}\mathrm{d}d \in \Omega(\ln\ln(d))\ . \tag{7}$$

Here, we use the crucial property that the antiderivative of ${}^{1}/_{x\ln(x)}$ is superconstant.

In the next steps we reuse the same strategy. Instead of bounding the inner sum on the left-hand side of Eq. (6) trivially by $2^{j-\ell+1}$ we use the stronger bound from Eq. (7) to obtain an even stronger bound $\sigma_d \in \Omega(\ln(d)^{1/17})$. Reapplying the same strategy a third time finally yields the lower bound $\sigma_d \in \Omega(\ln(d)^3)$ which is strong enough to yield a contradiction to Eq. (6). In this brief sketch we glossed over many details and refer the interested reader to the formal proof in Section 6. However, a quick sanity check may be in order at this point. If $\gamma_j \leq {}^{1}/_{j\ln(j)}$, then Lemma 1 holds trivially. Considering slightly larger $\gamma_j \coloneqq {}^{\epsilon}/_{j}$ for any constant $\epsilon > 0$ (thus $\sigma_j \approx \epsilon \ln(j)$) yields $\mathcal{O}(\sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^{\kappa} e^{\sigma_{j-\kappa} - \sigma_j}) = \mathcal{O}(\sum_{j=1}^{\ell} \frac{1}{j} \sum_{\kappa=1}^{j} 2^{\kappa}(1 - \kappa/j)^{\epsilon}) = \mathcal{O}(2^{\ell}/\ell^{1+\epsilon})$ which is also consistent with Lemma 1.

Robustness. The exact notion of Kt complexity has many dimensions. The first dimension is the number of tapes of our Turing machines (TM). In this work we consider linear time which is sensitive to the number of tapes, i.e., $\mathsf{DTIME}_{\mathsf{st}}[\mathcal{O}(n)] \subsetneq \mathsf{DTIME}_{\mathsf{mt}}[\mathcal{O}(n)]$ where st and mt stands for single-tape resp. multi-tape. Thus, it is important to note that the number of tapes in the definition of Kt complexity must match the number of tapes in the definition of DTIME. For example, it follows trivially from our presentation in terms of multi-tape TMs that $\mathsf{MKtP}_{\mathsf{mt}} \notin \mathsf{DTIME}_{\mathsf{st}}[\mathcal{O}(n)] \subsetneq \mathsf{DTIME}_{\mathsf{mt}}[\mathcal{O}(n)]$. On the other hand, we cannot prove $\mathsf{MKtP}_{\mathsf{st}} \notin \mathsf{DTIME}_{\mathsf{mt}}[\mathcal{O}(n)]$ using our technique because the TM in our proof would have multiple tapes whereas $\mathsf{MKtP}_{\mathsf{st}}$ requires that the witness program producing a string must only has a single tape, and we cannot simulate a multi-tape TM by a single-tape TM with constant multiplicative overhead. We do not analyze the single-tape version $\mathsf{MKtP}_{\mathsf{st}} \notin \mathsf{DTIME}_{\mathsf{st}}[\mathcal{O}(n)]$ of Theorem 1.

A second dimension in the definition of Kt complexity is whether a witness program $\mathit{\Pi}$ (the encoding of a TM) needs t steps to produce a given string x (denoted by $\mathrm{Kt}_{\mathit{\Pi}}$), or whether the universal TM (UTM) $\mathcal{U}$ simulating $\mathit{\Pi}$ takes t steps on input $\mathit{\Pi}$ to produce the string x (denoted by $\mathrm{Kt}_{\mathcal{U}}$). We want to clearly state that our Theorem 1 does *not* hold for the latter version if the UTM has a $\mathcal{O}(n \ln(n))$ slowdown when simulating an input TM $\mathit{\Pi}$. The reason is the following: in our proof we construct a TM that finds a Kt-random string z of length $\{0,1\}^{\hat{\ell}}$ in time $\mathcal{O}(i_{\hat{\ell}} \cdot \hat{\ell}) \leq_{\mathsf{io}} \mathcal{O}(2^{\hat{\ell}}/\ln(\hat{\ell}))$ which implies the contradiction $\hat{\ell} \leq \mathrm{Kt}(z) \leq \mathcal{O}(1) + \hat{\ell} - \log_2 \ln(\hat{\ell})$. However, if we instead count the number of steps that the UTM takes to produce z, then we arrive at $\hat{\ell} \leq \mathrm{Kt}(z) \leq \mathcal{O}(1) + \log_2(i_{\hat{\ell}}\hat{\ell} \cdot \ln(i_{\hat{\ell}}\hat{\ell})) \leq \hat{\ell} + \log_2(\hat{\ell}) - \log_2 \ln(\hat{\ell})$ which is not a contradiction.

Now, one might object that our definition of Kt complexity is non-standard.[3] We want to argue that the difference between the two defintions is minor for two reasons. First, the two versions are equivalent for any computational model with a linear-time universal simulation, e.g. Kolmogorov–Uspenskii machines (used by Levin [Lev73]) or random-access machines. Since these models are all Turing-complete, we view them as "morally equivalent" in the context of universal search, derandomization, etc. To support our claim we remark that the two notions differ by at most an additive logarithmic term, i.e., $|\,\mathrm{Kt}_{\mathit{\Pi}}(x) - \mathrm{Kt}_{\mathcal{U}}(x)| \leq \log_2 |x| + \mathcal{O}(1)$ (stemming from the $\mathcal{O}(n \ln(n))$ simulation overhead). Second, we remark that while the statment of Theorem 1 does not hold our technique does apply also for the latter version. Concretely, with our technique a lower bound against $\mathsf{DTIME}[n \cdot \mathsf{t}(n)]$ for the former version implies a lower bound aginst $\mathsf{DTIME}[\mathsf{t}(n)]$ for the latter version. In particular, this means a poylnomial bound $\mathsf{MKtP} \notin \mathsf{P}$ for one version translates into the same bound for the other.

Lastly, we emphasize that our results only hold for the prefix-free version Kt of Levin–Kolmogorov complexity, not for the plain version Ct. The reason is that our proof relies on an infinite 1-random string, which only exists for the prefix-free K complexity but not for the plain C complexity (see Section 6.1 in [DH10]). Again, we argue that (especially considering the lack of Kt lower bounds) this should not be interpreted as a fundamental limitation of the approach for our lower bound, but rather as an interesting technical artifact. In particular, one might find another way of arguing the "no-wrap-around" property of our search algorithm, to extend our result to plain Ct complexity.

Limitations and stronger conditional lower bounds. Our strategy using $\mathit{\Pi}'_{\mathsf{TRA}}$ cannot (unconditionally) tolerate any errors because

- if $\mathit{\Pi}'_{\mathsf{TRA}}$ obtains a false negative query response (false high complexity), then it outputs a string that is not actually Kt-random which does not violate the definition of Kt-randomness, and

[3] For reference, our former notion is used in [LP21b; RS22] among others, while the latter is used in [All01; LV08; ABK⁺06; All17; All20; Hir20b].

- if $\mathit{\Pi}'_{\mathsf{TRA}}$ obtains a false positive query response (false low complexity), then it may skip (from left to right) the separating line defined by Chaitin's 1-random constant, thus potentially increasing the runtime prohibitively.

However, we can conditionally tolerate some (false negative) one-sided errors. For example, suppose $\mathsf{MKtP} \in \mathsf{DTIME}[\mathcal{O}(n^2)]$ can be worst-case decided in quadratic time by a TM $\mathit{\Pi}_{\mathrm{Kt},n^2}$, and $\mathsf{MKtP} \in \mathsf{Heur}_{\gamma_{0,\mathsf{fn}}}\mathsf{DTIME}[\mathcal{O}(n)]$ can be decided in linear time with false negative probability $\gamma_{\mathsf{fn}}(n) \in \mathrm{o}(1/n^2)$ (and no false positives) by a TM $\widetilde{\mathit{\Pi}}_{\mathrm{Kt},n}$. Then we can construct a modified TM $\mathit{\Pi}''_{\mathsf{TRA}}$ which for each visited string z_i first queries z_i to the quicker linear-time heuristic $\widetilde{\mathit{\Pi}}_{\mathrm{Kt},n}$. If $\widetilde{\mathit{\Pi}}_{\mathrm{Kt},n}$ outputs $\widetilde{b} = 0$ (high complexity), $\mathit{\Pi}''_{\mathsf{TRA}}$ queries z_i to the slower $\mathit{\Pi}_{\mathrm{Kt},n^2}$ to obtain the definitive answer $b = 0 \iff z_i \in R_{\mathrm{Kt}}$. If $\widetilde{b} = 0 \wedge b = 0$, then $\mathit{\Pi}''_{\mathsf{TRA}}$ descends to $z_{i+1} \coloneqq z_i||0$, otherwise $z_{i+1} \coloneqq \mathrm{next}(z_i)$. First, note that $\mathit{\Pi}''_{\mathsf{TRA}}$ visits exactly the same set of strings (in the same order) as TRAVERSE. In contrast to the unconditional case, however, we find that whenever $\mathit{\Pi}''_{\mathsf{TRA}}$ visits a string $z_{i_{\hat{\ell}}}$ of critical length $\hat{\ell}$ it took at most

$$\mathcal{O}\left(\sum_{j=1}^{\hat{\ell}} |Z_j| \cdot j + 2^j \gamma_{\mathsf{fn}}(j) \cdot j^2\right) = \mathcal{O}\left(i_{\hat{\ell}} \cdot \hat{\ell} + 2^{\hat{\ell}} \gamma_{\mathsf{fn}}\left(\hat{\ell}\right) \cdot \hat{\ell}^2\right) \subseteq \mathrm{o}\left(2^{\hat{\ell}}\right) \tag{8}$$

steps because at length j there are at most $2^j \gamma_{\mathsf{fn}}(j)$ strings on which $\widetilde{\mathit{\Pi}}_{\mathrm{Kt},n}$ gives a false negative answer.[4] Consequently, when $\mathit{\Pi}''_{\mathsf{TRA}}$ visits and outputs such a string $z_{i_{\hat{\ell}}}$ it yields the contradiction

$$\hat{\ell} \leq \mathrm{Kt}\left(z_{i_{\hat{\ell}}}\right) \leq |\mathit{\Pi}_{\mathsf{TRA}}| + \left\lceil \log_2\left(\mathrm{o}\left(2^{\hat{\ell}}\right)\right)\right\rceil \leq \hat{\ell} - \omega(1) + \mathcal{O}(1) \ . \tag{9}$$

On overcoming the limitations. First, we want to point out a curious effect reminiscent of Williams's algorithmic method where a computational upper bound implies another lower bound. Namely, any nontrivial *worst-case* upper bound for MKtP (Item 1 in Theorem 2 is false) implies an improved linear-time lower bound for MKtP with one-sided error (Item 2 in Theorem 2 is true).

An interesting question specifically connected to a possible false positive tolerance of our approach concerns the measure of "universally" 1-random strings. It is known that (infinite) 1-random strings have uniform measure one [Mar66] (Corollary 6.2.6 in [DH10]). The notion of 1-randomness only requires that for each (infinite) string x there exists some constant $c \in \mathbb{N}$ such that for each $\ell \in \mathbb{N}$ it holds that $\mathrm{K}(x_1, ..., x_\ell) \geq \ell - c$. We say a string x is *c-universally* 1-random, if for each prefix length $\ell \in \mathbb{N}$ it holds that $\mathrm{K}(x_1||...||x_\ell) \geq \ell - c$. If it is the case that for some $c \in \mathbb{N}$ the c-universally 1-random strings have non-zero measure, then there would be many "left-right" borders induced by the c-universally 1-random strings. In this scenario, skipping a few borders (having a few false positives) may be tolerable. This would yield an unconditional linear-time lower bound tolerating some false positive errors.

Another obvious question is whether our technique is capable of proving a worst-case bound beyond linear time. We didn't flesh it out explicitly (because the current proof of Lemma 1 is already quite involved) but by an improved analysis of the proof of Lemma 1 we can push our bound to slightly superlinear time.

Corollary 1. *The Levin–Kolmogorov complexity cannot be decided in deterministic slightly superlinear time in the worst-case, i.e.,* $\mathsf{MKtP} \notin \mathsf{DTIME}[\prod_{i=0}^{k} \ln^{(k)}(n)]$ *for any fixed* $k \in \mathbb{N}$ *where* $\ln^{(k)}$ *is the* k *times iterated logarithm and* $\ln^{(0)}(n) \coloneqq n$.

The reason for this peculiar time bound $\mathfrak{t}(n) \coloneqq \prod_{i=0}^{k} \ln^{(k)}(n)$ is that the antiderivative $\int {}^{1}\!/_{\mathfrak{t}(x)} \mathrm{d}x = \ln^{(k+1)}(x) \in \omega(1)$ is superconstant which is the property that we need to get our proof of Lemma 1 going (compare to the previous paragraph on Lemma 1 and see the end of Section 6 for a sketch).
In contrast, going to some polynomial lower bound, i.e., $\mathsf{MKtP} \notin \mathsf{DTIME}[n^{1+\epsilon}]$ for some $\epsilon > 0$ seems

[4] Here, we naturally assume that $2^j \gamma_{\mathsf{fn}}(j)$ is non-decreasing.

hard. With our current proof strategy this would require a stronger version of Lemma 1 in the form of $\sum_{j=1}^{d} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa / e^{\sigma_j - \sigma_{j-\kappa}} \leq 2^d / d^{1+\epsilon}$ for some $\epsilon > 0$. However, this cannot hold for arbitrary γ_j because of the following counter example: $\gamma_j := \epsilon / j$ implies $\sum_{j=1}^{d} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa / e^{\sigma_j - \sigma_{j-\kappa}} \in \Theta(2^d / d^{1+\epsilon})$. Nonetheless, there is hope to achieve such a stronger bound; by leveraging more structural information about R_{Kt} one might prove a stronger version of Lemma 1 for a specific sequence γ_j corresponding to R_{Kt}.

4 Preliminaries

Notation. Real functions are usually denoted by Greek letters γ, θ, ε, etc., while natural/bit functions by Fraktur script $\mathfrak{t}$, $\mathfrak{f}$, etc. Languages are denoted by the uppercase letter L. Integers related to sizes are denoted by lowercase Latin letters n, m, c, while indices are denoted by i, j, k, κ. Strings are denoted by lowercase Latin letters x, y, z, etc. Turing machines (TM) are denoted by caligraphic letters $\mathcal{U}$, $\mathcal{M}$ as well as $\varPi$ for the code of a TM. Complexity classes are denoted in bold letters P, NP, EXP, etc.

For convenience we add $\boxed{\text{framed boxes}}$ with explanations of relevant (in-)equalities.

Notation 1 (Functional inequalities). *Let $\mathfrak{f}, \mathfrak{g} : \mathbb{N} \to \mathbb{R}$ be two functions. We write*

$$\mathfrak{f} \leq \mathfrak{g} \iff \forall n \in \mathbb{N} : \quad \mathfrak{f}(n) \leq \mathfrak{g}(n) \tag{10}$$

$$\mathfrak{f} \leq_{\mathsf{abf}} \mathfrak{g} \iff \exists n_0 \in \mathbb{N}\ \forall n \geq n_0 : \mathfrak{f}(n) \leq \mathfrak{g}(n) \tag{11}$$

$$\mathfrak{f} \leq_{\mathsf{io}} \mathfrak{g} \iff \forall n_0 \in \mathbb{N}\ \exists n \geq n_0 : \mathfrak{f}(n) \leq \mathfrak{g}(n) \tag{12}$$

when $\mathfrak{f}$ is less or equal to $\mathfrak{g}$ on all inputs, on all but finitely many inputs, or on infinitely many inputs. Note that

$$\mathfrak{f} \leq \mathfrak{g} \implies \mathfrak{f} \leq_{\mathsf{abf}} \mathfrak{g} \iff \overline{\mathfrak{f} >_{\mathsf{io}} \mathfrak{g}} \implies \mathfrak{f} \leq_{\mathsf{io}} \mathfrak{g}\ . \tag{13}$$

It may be that $\mathfrak{g} \leq_{\mathsf{io}} \mathfrak{f}$ while simultaneously $\mathfrak{g} \geq_{\mathsf{io}} \mathfrak{f}$. Sometimes we abuse notation and write $\mathfrak{f}(n) \leq_{\mathsf{abf}} \mathfrak{g}(n)$ to mean $(n \mapsto \mathfrak{f}(n)) \leq_{\mathsf{abf}} (n \mapsto \mathfrak{g}(n))$.

Notation 2 (Languages). *Let $L \subseteq \{0,1\}^*$, then for any $x \in \{0,1\}^*$ we use the abbreviated notation $L(x) = 1 \iff x \in L$ and $L(x) = 0 \iff x \notin L$.*

Notation 3 (Integers and strings). *Let* $\mathrm{int} : \{0,1\}^* \to \mathbb{N} : x \mapsto 2^{|x|+1} + \sum_{i=1}^{|x|} 2^{i-1} x_i$ *be the canonical lexicographical bijection between strings and integers. Let* $\mathrm{bin} := \mathrm{int}^{-1}$ *be its inverse operation. Let* $\mathrm{next}(x) := \mathrm{bin}((\mathrm{int}(x) + 1) \mod 2^{|x|} + 2^{|x|})$ *be the function that returns the lexicographically next string of the same length.*

Computational Model. For generality we choose to present our result for Turing machines. We might as well use a register-based computational model, e.g. random-access machines or Kolmogorov-Uspenskii machines, which gets rid of some definitorial issues of Levin–Kolomogorov complexity. Our results apply analogously to any other computational model.

Let $\mathcal{M}$ be a deterministic (multi-tape) Turing machine (TM). For any $x \in \{0,1\}^*$ denote by $\mathcal{M}(x) \in \{0,1\}^* \cup \{\perp\}$ the content of the output tape after $\mathcal{M}$ has entered a terminal state, or $\perp$ if $\mathcal{M}$ does not terminate on input x.
Let $\mathfrak{t} : \mathbb{N} \to \mathbb{N}$ be a time bound. Let $\mathsf{DTM}[\mathfrak{t}]$ be the set of deterministic TMs that terminate within $\mathfrak{t}(n)$ steps on inputs of length $n \in \mathbb{N}$. For any TM $\mathcal{M}$ let $L_{\mathcal{M}} := \{x \in \{0,1\}^* \mid \mathcal{M}(x) = 1\}$ be its (characteristic) language.

Throughout the paper let $\mathcal{U} : (\mathbb{N} \cup \{\infty\}) \times \{0,1\}^* \to \{0,1\}^* \cup \{\perp\}$ denote a prefix-free universal Turing machine (UTM). For any extended integer $t \in \mathbb{N} \cup \{\infty\}$ (runtime bound) and any bitstring $\varPi \in \{0,1\}^*$

(encoding of a multi-tape TM or "program") we denote by $\mathcal{U}(t, \varPi)$ the output of the program $\varPi$ after at most t steps.[5] If $\varPi$ does not terminate within t steps, we define $\mathcal{U}(t, \varPi) \coloneqq \bot$. Let

$$\mathsf{DTIME}[\mathsf{t}] \coloneqq \left\{ L \subseteq \{0,1\}^* \;\middle|\; \exists \mathcal{M} \in \mathsf{DTM}[\mathsf{t}] : L = L_{\mathcal{M}} \right\} \tag{14}$$

be the class of languages decided by some DTM in time t. Let $\mathsf{DTIME}[\mathcal{O}(\mathsf{t})] \coloneqq \bigcup_{d \in \mathbb{N}} \mathsf{DTIME}[d \cdot \mathsf{t}]$ be the class of languages decided by some DTM in time $\mathcal{O}(\mathsf{t})$.

In the following let C be some class of languages that is closed under intersection. Let

$$\mathsf{Heur}_{\gamma_{\mathsf{fp}},\gamma_{\mathsf{fn}}}\mathsf{C} \coloneqq \left\{ L \subseteq \{0,1\}^* \;\middle|\; \exists L' \in \mathsf{C} : \begin{array}{l} \left|(L \setminus L') \cap \{0,1\}^\lambda\right| \leq_{\mathsf{abf}} \gamma_{\mathsf{fp}}(\lambda) 2^\lambda \\ \left|(L' \setminus L) \cap \{0,1\}^\lambda\right| \leq_{\mathsf{abf}} \gamma_{\mathsf{fn}}(\lambda) 2^\lambda \end{array} \right\} \tag{15}$$

be the class of languages with a C-heuristic with false-positive error at most γ_{fp} and false-negative error at most γ_{fn}.

Complexity Measures. The most basic notion of Kolmogorov complexity is the length of the smallest program (witness program) that produces a given string w.r.t. some UTM?

Definition 1 (Solomonoff–Kolmogorov–Chaitin complexity [Sol60; Kol63; Cha69]). *Let $\mathcal{U}$ be a (prefix-free) UTM. For any string $x \in \{0,1\}^*$ we say*

$$\mathrm{K}_{\mathcal{U}}(x) \coloneqq \min\left\{ |\varPi| \;\middle|\; \varPi \in \{0,1\}^* : \mathcal{U}(\infty, \varPi) = x \right\} \tag{16}$$

is the (prefix-free) Kolmogorov complexity.[6]

While a powerful notion, it is not computable, hence Levin [Lev84] came up with an alternative definition which charges an additional logarithmic term for the runtime of the witness program that produces the given string.

Definition 2 (Levin–Kolmogorov complexity [Lev84; Tra84]). *Let $\mathcal{U}$ be a (prefix-free) UTM. For any string $x \in \{0,1\}^*$ we say*

$$\mathrm{Kt}_{\mathcal{U}}(x) \coloneqq \min\left\{ |\varPi| + \lceil \log_2(t) \rceil \;\middle|\; \varPi \in \{0,1\}^*, t \in \mathbb{N} : \mathcal{U}(t, \varPi) = x \right\} \tag{17}$$

is the (prefix-free) Levin–Kolmogorov complexity. Let

$$\mathsf{MKtP}_{\mathcal{U}} \coloneqq \{(y,k) \in \{0,1\}^m \times [m] \mid m \in \mathbb{N} : \mathrm{Kt}(y) \leq k\} \tag{18}$$

be the decisional minimum Kt*-problem.*

Definition 3 (1-randomness/Martin-Löf-randomness ([DH10] referring to [Mar66; Lev74; Cha75])). *Let $\mathcal{U}$ be a (prefix-free) UTM. An infinite sequence of bits $w = (w_i)_{i \in \mathbb{N}}$ is called* 1*-random, iff there exists some constant $\hat{c}_{\mathcal{U},w} \in \mathbb{N}$ such that for each $n \in \mathbb{N}$ it holds that $\mathrm{K}(w_1||...||w_n) \geq n - \hat{c}_{\mathcal{U},w}$.*

Analogously, an infinite sequence of bits $w = (w_i)_{i \in \mathbb{N}}$ is called 1-Kt*-random, iff there exists some constant $\hat{c}_{\mathcal{U},w} \in \mathbb{N}$ such that for each $n \in \mathbb{N}$ it holds that $\mathrm{Kt}(w_1||...||w_n) \geq n - \hat{c}_{\mathcal{U},w}$.*

Going forward we fix some arbitrary UTM $\mathcal{U}$ but omit it in our notation and simply write K, Kt, MKtP, etc.

Fact 1 (Chaitin's $\varOmega$ constant is 1-random [Cha75])**.** *Let $\varOmega_i$ be the i-th bit of Chaitin's constant [Cha75] in binary representation. Then the sequence $\varOmega = (\varOmega_i)_{i \in \mathbb{N}}$ is* 1*-random and thus* 1-Kt*-random with constant $\hat{c}_\varOmega$.*

[5] Here, we count the number of steps of $\varPi$ and not of $\mathcal{U}$ simulating $\varPi$ (see the paragraph on robustness in Section 3 for more details).

[6] For brevity and in accord with the literature [LV08] we simply use the term "Kolmogorov complexity".

5 Formal Results

Lemma 2. *The algorithm* $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ *in Fig. 3 visits infinitely many different strings* $(z_i)_{i\in\mathbb{N}}$.

Proof. First note that the lengths of z_i are non-decreasing, i.e., $|z_{i+1}| \geq |z_i|$. Suppose for contradiction that there exists some maximal length $\hat{\ell} \in \mathbb{N}$ such that all strings $|z_i| \leq \hat{\ell}$ for all $i \in \mathbb{N}$. By inspection it is apparent that from some point onward $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ cycles through all strings of length $\hat{\ell}$. Because the prefixes of Chaitin's constant $\Omega = (\Omega_i)_{i\in\mathbb{N}}$ are a 1-Kt-random sequence, for each length $\ell \in \mathbb{N}$ the string $\Omega_1||...||\Omega_\ell \in \{0,1\}^\ell$ has complexity $\mathrm{Kt}(\Omega_1||...||\Omega_\ell) \geq \mathrm{K}(\Omega_1||...||\Omega_\ell) \geq \ell - \hat{c}_\Omega$. Thus, once $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ visits the string $z_i = \Omega_1||...||\Omega_{\hat{\ell}}$ the next string is $z_{i+1} = \Omega_1||...||\Omega_{\hat{\ell}}||0$ due to line 3. This contradicts $\hat{\ell} + 1 = |z_i| + 1 \leq \hat{\ell}$. □

Now, we prove our main result.

Theorem 1. *The Levin–Kolmogorov complexity cannot be decided in deterministic linear time in the worst-case, i.e.,* $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathcal{O}(n)]$.

Proof. The intuition of this proof is already outlined in Section 3. The high-level idea is to assume that Kt can be computed quickly, and then construct a sufficiently fast TM that produces a highly complex string. This then contradicts the definition of a complex string needing a large or slow program to compute. Key properties of our constructed TM is that is finds a complex string sufficiently fast and that the TM is aware of its own runtime. For the latter property we use a counter variable in our TM to lower bound its runtime by counting the number of visited strings of a given length. This counter needs to be larger than the actual runtime of the TM (Claim 1) so it is guaranteed to output a string larger than its own critical threshold. On the other hand, the counter must not be too large (Claim 3), for otherwise it would not output critical strings that would actually suffice for a contradiction.

Suppose for contradiction $\mathsf{MKtP} \in \mathsf{DTIME}[\mathcal{O}(n)]$, then there exists some $c_{\mathrm{Kt}} \in \mathbb{N}$ such that $\mathsf{MKtP} \in \mathsf{DTIME}[2^{c_{\mathrm{Kt}}}n]$, i.e., there exists some TM Π_{Kt} that decides $\mathrm{Kt}(z) \leq k$ in time at most $\mathfrak{t}(n + \lceil \log_2(n) \rceil) \leq \mathfrak{t}(2n) := 2^{c_{\mathrm{Kt}}+1}n$ on any instance $(z,k) \in \{0,1\}^n \times [n]$. Later, we will choose a sufficiently large c_{Kt}. Fix the constant $\hat{c}_\Omega$ from Fact 1. For any c_{Kt} let $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ be the smallest TM implementing the $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ algorithm from Fig. 3. There exists some universal constant $c_{\mathsf{fix}} \in \mathbb{N}$ such that for any integer $c_{\mathrm{Kt}} \in \mathbb{N}$ the TM $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ has size $|\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}| \leq c_{\mathsf{fix}} + 2\lfloor \log_2(c_{\mathrm{Kt}}) + 1 \rfloor$ by storing c_{Kt} prefix-free. In particular, for any $c_{\mathrm{Kt}} \geq 2c_{\mathsf{fix}} + 4$ the TM's size is bounded by $|\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}| \leq c_{\mathrm{Kt}}$. We derive a contradiction through a series of claims about the TM $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$.

The TM $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ visits the sequence $(z_i)_{i\in\mathbb{N}}$ of strings. Let $Z := \{z_i \mid i \in \mathbb{N}\}$. For each length $\ell \in \mathbb{N}$ let $Z_\ell := Z \cap \{0,1\}^\ell = \{\hat{z}_\ell, ..., \check{z}_\ell\}$ where $\hat{z}_\ell$ and $\check{z}_\ell$ are the lexicographically first resp. last string in Z_ℓ. Our first claim establishes that—whenever $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ checks whether to output a visited string in line 10—its variable t_ℓ is larger than the number of steps that $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ took so far. This means that $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ can use t_ℓ to effectively bound its own critical threshold.

Claim 1 (Time counter lower bound)**.** *For any length* ℓ *let* $\widetilde{t_\ell}$ *be the number of steps that* $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ *takes to reach line 10 with length* ℓ*. It holds that* $t_\ell \geq \widetilde{t_\ell}$.

Proof. First, under the assumption $\mathsf{MKtP} \in \mathsf{DTIME}[2^{c_{\mathrm{Kt}}}n]$ we argue that $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ takes at most $c_{\mathrm{Kt}}^3\ell^2$ steps to execute line 9. Our first goal is to bound the time needed to execute line 9.
First, we recursively bound the variable t_ℓ by

$$t_\ell := t_{\ell-1} + (\mathrm{int}(z_i) - \mathrm{int}(\hat{z}_\ell) + 2)2^{c_{\mathrm{Kt}}+1}\ell + 2^{2c_{\mathrm{Kt}}+\ell-\lceil \log_2(\ell) \rceil+1} \tag{19}$$

$$= t_{\ell-1} + 2^{c_{\mathrm{Kt}}+2}|Z_\ell|\ell + 2^{2c_{\mathrm{Kt}}+\ell-\lceil \log_2(\ell) \rceil+1} \tag{20}$$

$$\leq t_{\ell-1} + 2^{4\ell+3c_{\mathrm{Kt}}} \tag{21}$$

$\text{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$

```
1 :  $z_1 \coloneqq 0 \in \{0,1\}^*$
2 :  $t_0 \coloneqq 0 \in \mathbb{N}$
3 :  $\ell \coloneqq 1 \in \mathbb{N}$
4 :  $\hat{z}_1 \coloneqq 1 \in \{0,1\}^*$
5 :  for $i \in \mathbb{N}_{\geq 1}$
6 :    if $\mathrm{Kt}(z_i) \gnsim \ell - \hat{c}_\Omega$   // in $2^{c_{\mathrm{Kt}}+1}\ell$ steps
7 :      $z_{i+1} \coloneqq z_i || 0 \in \{0,1\}^{\ell+1}$   // in $4\ell$ steps
8 :      $\hat{z}_{\ell+1} \coloneqq z_{i+1} \in \{0,1\}^{\ell+1}$   // store the starting node of length $\ell + 1$
9 :      $t_\ell \coloneqq t_{\ell-1} + (\mathrm{int}(z_i) - \mathrm{int}(\hat{z}_\ell) + 1)2^{c_{\mathrm{Kt}}+2}\ell + 2^{2c_{\mathrm{Kt}}+\ell-\lceil \log_2(\ell) \rceil+1}$   // in $c_{\mathrm{Kt}}^3\ell^2$ steps / add time spend on length $\ell$ plus safety margin for increasing the counter itself
10 :     if $\mathrm{Kt}(z_i) \gnsim c_{\mathrm{Kt}} + \lceil \log_2(t_\ell) \rceil$   // in $c_{\mathrm{Kt}}\ell + 2^{c_{\mathrm{Kt}}+1}\ell$ steps for computing $c_{\mathrm{Kt}} + \lceil \log_2(t_\ell) \rceil$ and deciding MKtP
11 :       return $z_i$
12 :     endif
13 :     $\ell \coloneqq \ell + 1$   // in $4\ell$ steps
14 :   else
15 :     $z_{i+1} \coloneqq \mathrm{next}(z_i) \in \{0,1\}^\ell$   // in $4\ell$
16 :   endif
17 : endfor
```

Fig. 3: Our search algorithm with runtime bounds under the assumption $\mathsf{MKtP} \in \mathsf{DTIME}[2^{c_{\mathrm{Kt}}}n]$. The parameters $\hat{c}_\Omega, c_{\mathrm{Kt}} \in \mathbb{N}$ are hardcoded. It might not be obvious why line 9 can be executed in $2c_{\mathrm{Kt}}^2\ell^2$ steps, the reason is fleshed out in the proof of Claim 1.

for sufficiently large c_{Kt}. Resolving this recursive upper bound it follows that $t_\ell \leq 2^{4\ell+3c_{\mathrm{Kt}}} + t_0$ where $t_0 \coloneqq 0$ and sufficiently large c_{Kt}.
Now, the value t_ℓ can be computed by simple arithmetic (addition, multiplication) and bit shifting operations taking at most quadratic time in the maximal bit length $\mathcal{O}(\ell)$ of the operands ℓ, t_ℓ $\mathrm{int}(\hat{z}_\ell)$, $\mathrm{int}(z_i) = \mathrm{int}(\check{z}_\ell)$ and c_{Kt}. That means there is some $c' \in \mathbb{N}$ (independent of c_{Kt}) such that t_ℓ can be computed in time $c' \log_2(t_\ell)^2 \leq c'(4\ell + 3c_{\mathrm{Kt}})^2 \leq c_{\mathrm{Kt}}^3\ell^2$ for sufficiently large c_{Kt}.

Taking a step back we observe that the TM $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ takes at most $\widetilde{\Delta}_\ell \coloneqq \widetilde{t}_\ell - \widetilde{t}_{\ell-1}$ actual steps to iterate over the strings Z_ℓ of length ℓ (lines 6 through 13). We see through

$$\begin{aligned}
\widetilde{\Delta}_\ell &\coloneqq \widetilde{t}_\ell - \widetilde{t}_{\ell-1} && (22)\\
&\leq \underbrace{|Z_\ell|\big(2^{c_{\mathrm{Kt}}+1}\ell + 4\ell\big)}_{\text{steps for } Z_\ell \setminus \{\check{z}_\ell\} \text{ in lines 6 and 15}} + \underbrace{2^{c_{\mathrm{Kt}}+1}\ell + 4\ell + c_{\mathrm{Kt}}^3\ell^2 + 2^{c_{\mathrm{Kt}}+1}\ell + c_{\mathrm{Kt}}\ell + 4\ell}_{\text{steps for } \check{z}_\ell \text{ in lines 6–13}} && (23)\\
&\leq \big(2^{c_{\mathrm{Kt}}+1} + 4\big)|Z_\ell|\ell + \big(2^{c_{\mathrm{Kt}}+2} + 8 + c_{\mathrm{Kt}}^3 + c_{\mathrm{Kt}}\big)\ell^2 && (24)\\
&\leq 2^{c_{\mathrm{Kt}}+2}|Z_\ell|\ell + 2^{2c_{\mathrm{Kt}}+\ell-\lceil \log_2(\ell) \rceil+1} \qquad \boxed{c_{\mathrm{Kt}} \geq 4} && (25)\\
&= \Delta_\ell && (26)
\end{aligned}$$

that the variable t_ℓ grows more quickly than $\widetilde{t}_\ell$ and since $t_0 = 0 = \widetilde{t}_0$, it follows that $t_\ell \geq \widetilde{t}_\ell$ for any $\ell \in \mathbb{N}$. ■

Claim 2 (Non-termination). *The TM $\mathcal{M}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ never halts, thus $\text{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ never halts.*

Proof. If $\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ halted and produced a string $\hat{z} \in \{0,1\}^{\hat{\ell}}$ within $\widetilde{t}_{\hat{\ell}}$ steps, then by definition of the Levin–Kolmogorov complexity

$$\mathrm{Kt}(\hat{z}) \leq |\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}| + \left\lceil \log_2\left(\widetilde{t}_{\hat{\ell}}\right)\right\rceil \leq c_{\mathrm{Kt}} + \left\lceil \log_2\left(\widetilde{t}_{\hat{\ell}}\right)\right\rceil \leq c_{\mathrm{Kt}} + \left\lceil \log_2\left(t_{\hat{\ell}}\right)\right\rceil \tag{27}$$

by the fact that $|\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}| \leq c_{\mathrm{Kt}}$ and Claim 1. However, the only way $\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ returns a string is in line 11, thus the condition in line 10 must be fulfilled, namely $\mathrm{Kt}(\hat{z}) \gneq c_{\mathrm{Kt}} + \lceil \log_2(t_{\hat{\ell}})\rceil$. This contradicts Eq. (27). Consequently, under the hypothesis $\mathsf{MKtP} \in \mathsf{DTIME}[2^{c_{\mathrm{Kt}}}n]$ the TM $\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ never halts. ■

Because of Claim 2 the TM $\mathcal{M}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ visits the same sequence of infinitely many different strings $(z_i)_{i\in\mathbb{N}}$ as $\mathrm{TRAVERSE}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$. For any length ℓ let $i_\ell := \sum_{j=1}^{\ell} |Z_j|$ be number of visited string of length at most ℓ.

Claim 3 (Time counter upper bound)**.** *For any length ℓ it holds that $t_\ell \leq 2^{2c_{\mathrm{Kt}}+4}(i_\ell \ell + 2^\ell/\ell)$.*

Proof. Using Eqs. (19) and (26) we can bound the telescope sum

$$\begin{aligned}
t_\ell &= t_0 + \sum_{j=1}^{\ell} \varDelta_j && (28)\\
&= \sum_{j=1}^{\ell}\left(2^{c_{\mathrm{Kt}}+2}|Z_j|j + 2^{2c_{\mathrm{Kt}}+j-\lceil \log_2(j)\rceil+1}\right) & \boxed{\textsf{Eq. (26)}} \quad & (29)\\
&\leq 2^{c_{\mathrm{Kt}}+2}\ell\left(\sum_{j=1}^{\ell}|Z_j|\right) + 2^{2c_{\mathrm{Kt}}+3+\ell}/\ell && (30)\\
&= 2^{c_{\mathrm{Kt}}+2} i_\ell \ell + 2^{2c_{\mathrm{Kt}}+3+\ell}/\ell & \boxed{i_\ell := \textstyle\sum_{j=1}^{\ell}|Z_j|} \quad & (31)\\
&\leq 2^{2c_{\mathrm{Kt}}+4}\left(i_\ell \ell + 2^\ell/\ell\right) \ . && (32)
\end{aligned}$$

■

Now we have upper bounded the counter variable t_ℓ in terms of the number i_ℓ of visited stings of length at most ℓ. It remains to argue that for infinitely many ℓ the value i_ℓ is sufficiently small, to reach a contradiction. To this end, we will reexpress i_ℓ in a different form. Let $S_\ell \subset \{0,1\}^\ell$ be the lexicographical successors of Z_ℓ (the right neighbors of Z_ℓ). Now, note that because $\mathrm{TRAVERSE}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ doesn't wrap around (staying to the left of Chaitin's constant), it holds that $Z_{\ell+1} \mathbin{\dot\cup} S_{\ell+1} = (\{z_{i_\ell}\} \mathbin{\dot\cup} S_\ell)||\{0,1\}$ and thus $|Z_{\ell+1}|+|S_{\ell+1}| = 2|S_\ell|+2$. Let $\gamma_\ell := {}^{|Z_\ell|}/_{|Z_\ell \dot\cup S_\ell|}$ be the fraction of strings of length ℓ that $\mathrm{TRAVERSE}_{\hat{c}_\Omega,c_{\mathrm{Kt}}}$ actually visits to the strings that it could potentially visit. Using this expression for γ_ℓ we can rewrite the previous equality as a recursive formula for $|S_\ell|$ (depending on γ_ℓ), i.e.,

$$\begin{aligned}
2|S_\ell| + 2 &= |Z_{\ell+1}| + |S_{\ell+1}| && (33)\\
&= (|Z_{\ell+1}| + |S_{\ell+1}|)\gamma_{\ell+1} + |S_{\ell+1}| && (34)\\
&= 2(|S_\ell| + 1)\gamma_{\ell+1} + |S_{\ell+1}| && (35)\\
\implies |S_{\ell+1}| &= 2(|S_\ell| + 1)(1 - \gamma_{\ell+1}) && (36)
\end{aligned}$$

By solving this recursion with $|S_1| := 1$ we can express the number of successor strings as

$$|S_\ell| = \sum_{\kappa=1}^{\ell} 2^\kappa \prod_{i=\ell-\kappa+1}^{\ell} (1 - \gamma_i) \ . \tag{37}$$

In turn, we can use the definition of γ_ℓ to express the number of visited strings of length exactly ℓ as

$$|Z_\ell| = 2(|S_{\ell-1}|+1)\gamma_\ell = \gamma_\ell \sum_{\kappa=1}^{\ell} 2^\kappa \prod_{i=\ell-\kappa+1}^{\ell-1} (1-\gamma_i) \ . \tag{38}$$

Lastly, we can sum over all lengths to obtain

$$\begin{aligned}
i_\ell &\coloneqq \sum_{j=1}^{\ell} |Z_j| && (39)\\
&= \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa \prod_{i=j-\kappa+1}^{j} (1-\gamma_i) && (40)\\
&= \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa e^{\sum_{i=j-\kappa+1}^{j} \ln(1-\gamma_i)} && (41)\\
&\le \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa e^{-\sum_{i=j-\kappa+1}^{j} \gamma_i} && \boxed{\ln(1-x) \le -x} \quad (42)\\
&= \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa e^{\sigma_{j-\kappa} - \sigma_j} && (43)
\end{aligned}$$

where $\sigma_\ell \coloneqq \sum_{i=1}^{\ell} \gamma_i$. This expression is bounded by Lemma 1.

Conclusion. Using Lemma 1 let $\hat{\ell} \ge e^{2^{3c_{\mathrm{Kt}}+\hat{c}_\Omega+6}}$ be an arbitrarily large integer such that $i_{\hat{\ell}} \le {}^{2^{\hat{\ell}}}/_{\hat{\ell}\ln(\hat{\ell})}$. Let $z_{i_{\hat{\ell}}} \in \{0,1\}^{\hat{\ell}}$ be the last string of length $\hat{\ell}$ visited by $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$. Because $z_{i_{\hat{\ell}}}$ is the last string of length $\hat{\ell}$ the condition $\mathrm{Kt}(z_{i_{\hat{\ell}}}) \gneq |z_{i_{\hat{\ell}}}| - \hat{c}_\Omega = \hat{\ell} - \hat{c}_\Omega$ in line 6 in $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ must be fulfilled. Moreover, because $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ never halts—according to Claim 2—the violated return condition in line 10 in $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}}$ dictates $\mathrm{Kt}(z_{i_{\hat{\ell}}}) \le c_{\mathrm{Kt}} + \lceil \log_2(t_{\hat{\ell}}) \rceil$. Thus we arrive at the contradiction

$$\begin{aligned}
\hat{\ell} - \hat{c}_\Omega &\lneq \mathrm{Kt}\big(z_{i_{\hat{\ell}}}\big) && (44)\\
&\le c_{\mathrm{Kt}} + \big\lceil \log_2\big(t_{\hat{\ell}}\big) \big\rceil && (45)\\
&\le c_{\mathrm{Kt}} + \left\lceil (2c_{\mathrm{Kt}}+4) + \log_2\left(i_{\hat{\ell}}\hat{\ell} + 2^{\hat{\ell}}/\hat{\ell}\right) \right\rceil && \boxed{\textsf{Claim 3}} \quad (46)\\
&\le c_{\mathrm{Kt}} + \left\lceil (2c_{\mathrm{Kt}}+4) + \log_2\left(2^{\hat{\ell}}/\ln\left(\hat{\ell}\right) + 2^{\hat{\ell}}/\hat{\ell}\right) \right\rceil && \boxed{\textsf{Lemma 1}} \quad (47)\\
&\le c_{\mathrm{Kt}} + 2c_{\mathrm{Kt}} + 6 + \log_2\left(2^{\hat{\ell}}/\ln\left(\hat{\ell}\right)\right) && \boxed{\hat{\ell} \gneq 1} \quad (48)\\
&= c_{\mathrm{Kt}} + 2c_{\mathrm{Kt}} + 6 + \hat{\ell} - \log_2 \ln\left(\hat{\ell}\right) && (49)\\
&\le c_{\mathrm{Kt}} + 2c_{\mathrm{Kt}} + 6 + \hat{\ell} - (3c_{\mathrm{Kt}} + \hat{c}_\Omega + 6) && \boxed{\hat{\ell} \ge e^{2^{3c_{\mathrm{Kt}}+\hat{c}_\Omega+6}}} \quad (50)\\
&= \hat{\ell} - \hat{c}_\Omega \ . && (51)
\end{aligned}$$

□

Remark 1. The proof of Theorem 1 relativizes.

Next, we prove our conditional lower bounds.

Theorem 2. *For each time bound* $\mathsf{t}(n) \ge n$ *at least one of the following is true:*
1. $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathsf{t}]$,

2. $\mathsf{MKtP} \notin \mathsf{Heur}_{\gamma_{\mathsf{fp}},\gamma_{\mathsf{fn}}}\mathsf{DTIME}[\mathcal{O}(n)]$ *with no false positive error* $\gamma_{\mathsf{fp}}(n) \coloneqq 0$ *and false negative error* $\gamma_{\mathsf{fn}}(n) \coloneqq {}^{1}\!/_{2n\mathfrak{t}(2n)} - {}^{2}\!/_{2^n}$*,*

Proof. This proof is a slight modification of the proof of Theorem 1, thus we only include the relevant changes. For contradiction assume $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathfrak{t}]$ (by a TM $\mathit{\Pi}_{\mathrm{Kt}}$) and $\mathsf{MKtP} \notin \mathsf{Heur}_{0,\gamma_{\mathsf{fn}}}\mathsf{DTIME}[\mathcal{O}(n)]$ with false negative error probability $\gamma_{\mathsf{fn}}(n) \coloneqq {}^{1}\!/_{2n\mathfrak{t}(2n)} - {}^{2}\!/_{2^n}$ (by a TM $\widetilde{\mathit{\Pi}}_{\mathrm{Kt}}$). See Fig. 4 for the modified traversal algorithm $\mathrm{TRAVERSE}'_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$. Let $\mathcal{M}'_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$ be a TM implementing the modified $\mathrm{TRAVERSE}'_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$. Clearly, if the analog of Claim 1 holds, then $\mathcal{M}'_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$ does not terminate for the same reason as in Claim 2 (note there that the check in line 14 is an errorless check). Because the definition of the counter variable t_ℓ is identical to $\mathrm{TRAVERSE}_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$ the analog of Claim 3 also holds.

It remains to argue the analog of Claim 1. As before we observe that the TM $\mathcal{M}'_{\hat{c}_\Omega, c_{\mathrm{Kt}}, \mathit{\Pi}_{\mathrm{t}}, \mathit{\Pi}_{\mathrm{Kt}}, \widetilde{\mathit{\Pi}}_{\mathrm{Kt}}}$ takes at most $\widetilde{\mathit{\Delta}}_\ell \coloneqq \widetilde{t}_\ell - \widetilde{t}_{\ell-1}$ actual steps to iterate over the strings Z_ℓ of length ℓ (lines 6 through 17). Though, note here that we incur an additional cost for the exact check using time $\mathfrak{t}(2\ell)$ on at most $2^\ell\gamma_{\mathsf{fn}}(\ell)$ strings of length ℓ, plus one exact check at in line 14. Thus, we see through

$$
\begin{aligned}
\widetilde{\mathit{\Delta}}_\ell &\coloneqq \widetilde{t}_\ell - \widetilde{t}_{\ell-1} && (52)\\
&\le \underbrace{|Z_\ell|\big(2^{c_{\mathrm{Kt}}+1}\ell + 4\ell\big) + 2^\ell\gamma_{\mathsf{fn}}(\ell)\mathfrak{t}(2\ell)}_{\text{steps for } Z_\ell\setminus\{\check{z}_\ell\} \text{ in lines 6,8,19}} + \underbrace{2^{c_{\mathrm{Kt}}+1}\ell + \mathfrak{t}(2\ell) + 4\ell + c_{\mathrm{Kt}}^3\ell^2 + c_{\mathrm{Kt}}\ell + \mathfrak{t}(2\ell) + 4\ell}_{\text{steps for } \check{z}_\ell \text{ in lines 6–17}} && (53)\\
&\le \big(2^{c_{\mathrm{Kt}}+1} + 4\big)|Z_\ell|\ell + \big(2^\ell\gamma_{\mathsf{fn}}(\ell) + 2\big)\mathfrak{t}(2\ell) + \big(2^{c_{\mathrm{Kt}}+1} + 8 + c_{\mathrm{Kt}}^3 + c_{\mathrm{Kt}}\big)\ell^2 && (54)\\
&\le 2^{c_{\mathrm{Kt}}+2}|Z_\ell|\ell + \big(2^\ell\gamma_{\mathsf{fn}}(\ell) + 2\big)\mathfrak{t}(2\ell) + 2^{2c_{\mathrm{Kt}}+\ell-\lceil\log_2(\ell)\rceil} && (55)\\
&\le 2^{c_{\mathrm{Kt}}+2}|Z_\ell|\ell + 2^{2c_{\mathrm{Kt}}+\ell-\lceil\log_2(\ell)\rceil+1} && (56)\\
&= t_\ell - t_{\ell-1} && (57)\\
&\eqqcolon \mathit{\Delta}_\ell && (58)
\end{aligned}
$$

that the variable t_ℓ grows more quickly than $\widetilde{t}_\ell$ and since $t_0 = 0 = \widetilde{t}_0$, it follows that $t_\ell \ge \widetilde{t}_\ell$ for any $\ell \in \mathbb{N}$ which establishes Claim 3. The concluding part of the proof works exactly as in the proof of Theorem 1.

The reason why our proof can tolerate the additional runtime cost caused by the exact Kt solver $\mathit{\Pi}_{\mathrm{Kt}}$ is because the safety margin that we add to the counter in line 13 is more than we actually need for Theorem 1.

□

6 Proof of Lemma 1

Because Lemma 1 is essential to our main result we include a rigorous proof in the main body of this work.

Lemma 1 (Infinitely-often bound). *For any sequence* $(\gamma_j)_{j\in\mathbb{N}}$ *with* $\gamma_j \in [0,1]$ *and* $\sigma_\ell \coloneqq \sum_{i=1}^{\ell}\gamma_i$ *it holds that infinitely often* $\sum_{j=1}^{\ell}\gamma_j\sum_{\kappa=1}^{j}2^\kappa e^{\sigma_{j-\kappa}-\sigma_j} \le_{\mathsf{io}} {}^{2^\ell}\!/_{\ell\ln(\ell)}$*.*

Proof. The proof of this claim is quite technical and somewhat tedious although it fundamentally only requires analytic Riemann integration bounds (see Appendix A). A high-level intuition for our bound may best be explained by looking at the double sum

$$\mathfrak{s}(\ell) \coloneqq \sum_{j=1}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad (59)$$

```
$\text{TRAVERSE}'_{\hat{c}_\Omega, c_{\text{Kt}}, \Pi_{\mathfrak{t}}, \Pi_{\text{Kt}}, \widetilde{\Pi}_{\text{Kt}}}$
$z_1 \coloneqq 0 \in \{0,1\}^*$
$t_0 \coloneqq 0 \in \mathbb{N}$
$\ell \coloneqq 1 \in \mathbb{N}$
$\hat{z}_1 \coloneqq 1 \in \{0,1\}^*$
for $i \in \mathbb{N}_{\geq 1}$
    $b \coloneqq \widetilde{\Pi}_{\text{Kt}}(z_i, \ell - \hat{c}_\Omega)$   // in $2^{c'+1}\ell$ steps / quick error-prone check
    if $\mathrm{b} = 1$
        $b \coloneqq \Pi_{\text{Kt}}(z_i, \ell - \hat{c}_\Omega)$   // in $\mathfrak{t}(2\ell)$ steps / slower exact check
    fi
    if $\mathrm{b} = 0$   // assert $\text{Kt}(z_i) \gnsim \ell - \hat{c}_\Omega$
        $z_{i+1} \coloneqq z_i || 0 \in \{0,1\}^{\ell+1}$   // in $4\ell$ steps
        $\hat{z}_{\ell+1} \coloneqq z_{i+1} \in \{0,1\}^{\ell+1}$   // store the starting node of length $\ell + 1$
        $t'_\ell \coloneqq t_{\ell-1} + (\text{int}(z_i) - \text{int}(\hat{z}_\ell) + 2)2^{c_{\text{Kt}}+1}\ell + 2^{2c_{\text{Kt}}+\ell-\lceil \log_2(\ell) \rceil+1}$   // in $c_{\text{Kt}}^3\ell^2$ steps / add time spend on length $\ell$ plus safety margin for increasing the counter itself
        if $\Pi_{\text{Kt}}(z_i, c_{\text{Kt}} + \lceil \log_2(t_\ell) \rceil) = 0$   // in $c_{\text{Kt}}\ell + \mathfrak{t}(2\ell)$ steps / assert $\text{Kt}(z_i) \gnsim c_{\text{Kt}} + \lceil \log_2(t_\ell) \rceil$
            return $z_i$
        endif
        $\ell \coloneqq \ell + 1$   // in $4\ell$ steps
    else
        $z_{i+1} \coloneqq \text{next}(z_i) \in \{0,1\}^\ell$   // in $4\ell$
    endif
endfor
```

Fig. 4: Our search algorithm with runtime bounds under the assumption $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathfrak{t}]$ and $\mathsf{MKtP} \notin \mathsf{Heur}_{\gamma_{\mathsf{fp}},0}\mathsf{DTIME}[\mathcal{O}(n)]$ where $\mathfrak{t}$ is assumed to be time-constructible. The parameters $\hat{c}_\Omega, c_{\text{Kt}} \in \mathbb{N}$, the TM $\Pi_{\mathfrak{t}}$ computing $\mathfrak{t}$, the $\mathfrak{t}$-time TM Π_{Kt} and the linear-time TM $\widetilde{\Pi}_{\text{Kt}}$ are hardcoded. Changes to Fig. 3 are marked in gray.

where $\sigma_\ell \coloneqq \sum_{i=1}^{\ell} \gamma_i$. We don't know the exact values of $\gamma_j \in [0,1]$ but we see that the summands of the outer sum depend on γ_j in two ways. The faster γ_j grows the faster the outer summands grow because the j-th summand depends linearly on γ_j. On the other hand, the faster γ_j grows the faster σ_j grows and thus the slower the inner summands grow because of the e^{σ_j} term in the denominator of the κ-th inner summand. So, there is a "sweet spot" for the asymptotic growth rate of γ_j that maximizes the growth rate of $\mathfrak{s}$. The maximal growth rate is close to $\Theta(\sum_{j=1}^{\ell} \frac{1}{j} \sum_{\kappa=1}^{j} 2^\kappa (1 - \frac{\kappa}{j})^\epsilon) = \Theta(2^\ell / \ell^{1+\epsilon})$ for small $\epsilon > 0$ and $\gamma_j = \epsilon/j$, thus $\sigma_j \approx \epsilon \ln(j)$. Thus we cannot hope to prove $\mathfrak{s}(\ell) \in \mathcal{O}(2^\ell/\ell^{1+\epsilon})$ without further restrictions on γ_j. However, we can prove a weaker bound $\mathfrak{s}(\ell) \leq_{\mathsf{io}} \mathcal{O}(2^\ell/\ell \ln(\ell))$. The way we prove this bound is by establishing increasingly stronger lower bounds for the sum σ_ℓ. The first bound will be of the rough form $\sigma_\ell \in \Omega(\ln\ln(\ell))$, the second one $\sigma_\ell \in \Omega(\ln(\ell)^{1/17})$ and the third one $\sigma_\ell \in \Omega(\ln(\ell)^3)$. The last bound then yields a contradiction to the counter assumption $2^\ell/\ell \ln(\ell) \leq_{\mathsf{abf}} \mathfrak{s}(\ell)$.

Let us proceed with the formal proof. In long equations we highlight changes relative to the previous line with a gray background and give explanations in $\boxed{\text{framed boxes}}$. Also, we use the convention that for any $b < a$ the sum $\sum_{i=a}^{b} \mathfrak{f}(i) \coloneqq 0$. Suppose for contradiction

$$\frac{2^{\ell}}{\ell \ln(\ell)} \le_{\mathsf{abf}} \mathfrak{s}(\ell) \coloneqq \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} \frac{2^{\kappa}}{e^{\sigma_j - \sigma_{j-\kappa}}}\ , \tag{60}$$

then there exists some $\ell_1 \in \mathbb{N}$ such that for all $\ell \ge \ell_1$

$$\begin{aligned}
\frac{1}{\ell \ln(\ell)} &\le \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j - \sigma_{j-\kappa}}} && (61)\\
&\le \sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^{\kappa-\ell} && (62)\\
&\le \sum_{j=1}^{\ell} 2^{j+1-\ell} \gamma_j && (63)
\end{aligned}$$

where Eq. (62) trivially uses $\sigma_j \ge \sigma_{j-k}$ and Eq. (63) uses $\sum_{\kappa=1}^{j} 2^{\kappa} = 2^{j+1} - 2$. For convenience we define a helper variable $\delta_\ell \coloneqq \max(0, \lceil \ln\ln(\ell+1)/8 - \ln\ln(\ell_1)/4 \rceil) \le \ell$. Note that $\delta_\ell \ge \log_2 \ln(\ell)/16$ for $\ell \ge \ell_1$ if ℓ_1 is sufficiently large (which is without loss of generality). Using Riemann integration on the sum of Eq. (61) from ℓ_1 to ℓ yields

$$\begin{aligned}
\frac{1}{4}\ln\ln(\ell+1) - \frac{1}{4}\ln\ln(\ell_1) &= \frac{1}{2}\int_{\ell_1}^{\ell+1} \frac{1}{2x\ln(x)} \mathrm{d}x && (64)\\
&\le \frac{1}{2}\sum_{i=\ell_1}^{\ell} \frac{1}{2i\ln(i)} && \boxed{\begin{array}{c}\text{Fact 2}\\ \text{Riemann integration}\end{array}} \quad (65)\\
&\le \frac{1}{2}\sum_{i=\ell_1}^{\ell} \sum_{j=1}^{i} 2^{j-i}\gamma_j && \boxed{\begin{array}{c}\text{Eq. (63)}\\ \frac{1}{2i\ln(i)} \le \sum_{j=1}^{i} 2^{j-i}\gamma_j\end{array}} \quad (66)\\
&\le \frac{1}{2}\sum_{i=1}^{\ell} \sum_{j=1}^{i} 2^{j-i}\gamma_j && \boxed{\ell_1 \ge 1} \quad (67)\\
&= \frac{1}{2}\sum_{i=1}^{\ell} 2^{-i} \sum_{j=1}^{i} 2^{j}\gamma_j && (68)\\
&= \frac{1}{2}\sum_{j=1}^{\ell} 2^{j}\gamma_j \sum_{i=j}^{\ell} 2^{-i} && \boxed{\begin{array}{c}\text{Lemma 3}\\ \text{sum switching}\end{array}} \quad (69)\\
&\le \frac{1}{2}\sum_{j=1}^{\ell} 2^{j}\gamma_j \sum_{i=j}^{\infty} 2^{-i} && \boxed{\ell < \infty \text{ and } 2^{-i} \ge 0} \quad (70)\\
&= \sum_{j=1}^{\ell} \gamma_j && \boxed{\textstyle\sum_{i=j}^{\infty} 2^{-i} = 2^{1-j}} \quad (71)\\
&= \sigma_\ell && (72)\\
&\le \sigma_\ell - \sigma_{\delta_\ell} + \delta_\ell && \boxed{\sigma_{\delta_\ell} \le \delta_\ell}\ . \quad (73)
\end{aligned}$$

Reordering the terms yields

$$\sigma_\ell - \sigma_{\delta_\ell} \geq \ln\ln(\ell+1)/8 \geq \ln\ln(\ell)/16 \tag{74}$$

for all $\ell \geq \ell_1$. To get his bound we started Eq. (61) off with the trivial bound $\sigma_j \geq \sigma_{j-\kappa}$. Now, we can use our new nontrivial bound for σ_j repeat the previous procedure and obtain an even better bound.

Plugging Eq. (74) back into a weighted sum of Eq. (61) gives the better bound on σ_ℓ for $\ell \geq \ell_1$, i.e.,

$$\ln(\ell+1) - \ln(\ell_1) \tag{75}$$

$$= \int_{\ell_1}^{\ell+1} \frac{1}{x}\mathrm{d}x \tag{76}$$

$$\leq \sum_{i=\ell_1}^{\ell} \frac{1}{i} \qquad \boxed{\begin{array}{c}\textsf{Fact 2}\\ \text{Riemann integration}\end{array}} \tag{77}$$

$$= \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{i\ln(i)} \tag{78}$$

$$\leq \sum_{i=\ell_1}^{\ell} \ln(i) \sum_{j=1}^{i} \gamma_j \sum_{\kappa=1}^{j} \frac{2^{\kappa-i}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\textsf{Eq. (61)}} \tag{79}$$

$$= \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \sum_{\kappa=1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \tag{80}$$

$$\leq \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \sum_{\kappa=1}^{\max(j,\ell_1-1)} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{j \leq \max(j,\ell_1-1)} \tag{81}$$

$$= \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \left(\sum_{\kappa=1}^{\ell_1-1} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} + \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\begin{array}{c}\text{split sum}\\ \text{if } j \lneq \ell_1 \text{ then}\\ \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} = 0\end{array}} \tag{82}$$

$$\leq \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \left(2^{\ell_1} + \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\begin{array}{c}\sigma_j \geq \sigma_{j-\kappa} \text{ and}\\ \sum_{\kappa=1}^{\ell_1-1} 2^{\kappa} = 2^{\ell_1} - 2\end{array}} \tag{83}$$

$$= \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \left(2^{\ell_1}\sigma_i + \sum_{j=1}^{i} \gamma_j \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \tag{84}$$

$$\leq 2^{\ell_1} \sum_{i=\ell_1}^{\ell} \frac{i\ \ln(i)}{2^i} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ \ell_1}^{i} \gamma_j \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\gamma_j \leq 1 \implies \sigma_i \leq i} \tag{85}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \sum_{\kappa=\ell_1}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\textstyle\sum_{i=1}^{\infty} \frac{i\ln(i)}{2^i} \leq 2} \tag{86}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(\sum_{\kappa=\ell_1}^{j-1-\delta_j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^{\kappa}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\text{split sum}} \tag{87}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(\sum_{\kappa=\ell_1}^{j-1-\delta_j} 2^\kappa + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \right) \qquad \boxed{\sigma_j \geq \sigma_{j-\kappa}} \tag{88}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(2^{j-\delta_j} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \right) \qquad \boxed{\textstyle\sum_{\kappa=\ell_1}^{j-1-\delta_j} 2^\kappa \leq 2^{j-\delta_j}} \tag{89}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(2^{j-\delta_j} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{\delta_j}}} \right) \qquad \boxed{\begin{array}{c} \kappa \geq j - \delta_j \\ \implies \delta_j \geq j - \kappa \end{array}} \tag{90}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(2^{j-\delta_j} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{\ln(j)^{1/16}} \right) \qquad \boxed{\begin{array}{c} \sigma_j - \sigma_{\delta_j} \geq \ln\ln(j)/16 \\ \text{for all } j \geq \ell_1 \text{ by Eq. (74)} \end{array}} \tag{91}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(\frac{2^j}{\ln(j)^{1/16}} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{\ln(j)^{1/16}} \right) \qquad \boxed{\delta_j \geq \log_2 \ln(j)/16} \tag{92}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \left(\frac{2^{j+1}}{\ln(j)^{1/16}} + \sum_{\kappa=j-\delta_j}^{j} \frac{2^\kappa}{\ln(j)^{1/16}} \right) \tag{93}$$

$$= 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \sum_{\kappa=j-\delta_j}^{j+1} \frac{2^\kappa}{\ln(j)^{1/16}} \tag{94}$$

$$\leq 2^{\ell_1+1} + \sum_{i=\ell_1}^{\ell} \frac{\ln(i)}{2^i} \sum_{j=\ell_1}^{i} \gamma_j \cdot \frac{2^{j+2}}{\ln(j)^{1/16}} \qquad \boxed{\textstyle\sum_{\kappa=j-\delta_j}^{j+1} 2^\kappa \leq 2^{j+2}} \tag{95}$$

$$= 2^{\ell_1+1} + 4 \sum_{j=\ell_1}^{\ell} \gamma_j \cdot \frac{2^j}{\ln(j)^{1/16}} \sum_{i=j}^{\ell} \frac{\ln(i)}{2^i} \qquad \boxed{\begin{array}{c} \text{Lemma 3} \\ \text{sum switching} \end{array}} \tag{96}$$

$$\leq 2^{\ell_1+1} + 4 \sum_{j=\ell_1}^{\ell} \gamma_j \cdot \frac{2^j}{\ln(j)^{1/16}} \sum_{i=j}^{\infty} \frac{\ln(i)}{2^i} \qquad \boxed{\ell < \infty} \tag{97}$$

$$\leq 2^{\ell_1+1} + 16 \sum_{j=\ell_1}^{\ell} \gamma_j \cdot \frac{2^j}{\ln(j)^{1/16}} \frac{\ln(j)}{2^j} \qquad \boxed{\begin{array}{c} \text{Lemma 4 with} \\ \nu = 0, j \geq \ell_1 \geq 4 \end{array}} \tag{98}$$

$$= 2^{\ell_1+1} + 16 \sum_{j=\ell_1}^{\ell} \gamma_j \ln(j)^{15/16} \tag{99}$$

$$\leq 2^{\ell_1+1} + 16 \sum_{j=\ell_1}^{\ell} \gamma_j \ln(\ell)^{15/16} \qquad \boxed{\begin{array}{c} \ln(x)^{15/16} \\ \text{is non-decreasing} \end{array}} \tag{100}$$

$$\leq 2^{\ell_1+1} + 16\, \sigma_\ell \ln(\ell)^{15/16} \,. \tag{101}$$

Let $\delta'_\ell := \lceil \log_2(e)\ln(\ell)^{1/17}\rceil$. Thus there exists some sufficiently large $\ell_2 \in \mathbb{N}$ s.t. for all $\ell \geq \ell_2$ it holds that

$$\begin{aligned}
\sigma_\ell - \sigma_{\delta'_\ell} &\geq \sigma_\ell - \delta'_\ell \qquad (102)\\
&\geq \left(\ln(\ell+1) - \ln(\ell_1) - 2^{\ell_1+1}\right)/\left(16\ln(\ell)^{15/16}\right) - \log_2(e)\ln(\ell)^{1/17} - 1 \qquad (103)\\
&\geq \ln(\ell)^{1/17}\ . \qquad (104)
\end{aligned}$$

Now, we repeat the previous strategy for a third time to reach the final sufficient bound $\sigma_\ell \in \Omega(\ln(\ell)^3)$. Plugging Eq. (102) back into a weighted sum of Eq. (61) gives the better bound on σ_ℓ for $\ell \geq \ell_2$

$$2(\ell+1)^{1/2} - 2\ell_2^{1/2} \tag{105}$$

$$= \int_{\ell_2}^{\ell+1} \frac{x^{1/2}}{x}\mathrm{d}x \tag{106}$$

$$\leq \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}}{i} \qquad \boxed{\begin{array}{c}\textsf{Fact 2}\\ \text{Riemann integration}\end{array}} \tag{107}$$

$$= \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{i\ln(i)} \tag{108}$$

$$\leq \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \sum_{\kappa=1}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \qquad \boxed{\textsf{Eq. (61)}} \tag{109}$$

$$\leq \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \sum_{\kappa=1}^{\max(j,\ell_2-1)} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \qquad \boxed{j \leq \max(j, \ell_1 - 1)} \tag{110}$$

$$\leq \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \left(\sum_{\kappa=1}^{\ell_2-1} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} + \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \right) \qquad \boxed{\begin{array}{c}\text{split sum}\\ \text{if } j \lneq \ell_2 \text{ then}\\ \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} = 0\end{array}} \tag{111}$$

$$\leq \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=1}^{i} \gamma_j \left(2^{\ell_2} + \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \right) \qquad \boxed{\begin{array}{c}\sigma_j \geq \sigma_{j-\kappa} \text{ and}\\ \sum_{\kappa=1}^{\ell_2-1} 2^\kappa = 2^{\ell_2} - 2\end{array}} \tag{112}$$

$$= \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \left(2^{\ell_2}\sigma_i + \sum_{j=1}^{i} \gamma_j \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \right) \tag{113}$$

$$\leq 2^{\ell_2} \sum_{i=\ell_2}^{\ell} \frac{i^{3/2}\ln(i)}{2^i} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j - \sigma_{j-\kappa}}} \qquad \boxed{\gamma_i \leq 1 \implies \sigma_i \leq i} \tag{114}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \sum_{\kappa=\ell_2}^{j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\textstyle\sum_{i=\ell_2}^{\ell} \frac{i^{3/2}\ln(i)}{2^i} \leq 2} \tag{115}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(\sum_{\kappa=\ell_2}^{j-1-\delta'_j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\text{split sum}} \tag{116}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(\sum_{\kappa=\ell_2}^{j-1-\delta'_j} 2^\kappa + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\sigma_j \geq \sigma_{j-\kappa}} \tag{117}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(2^{j-\delta'_j} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\textstyle\sum_{\kappa=\ell_2}^{j-1-\delta_j} 2^\kappa \leq 2^{j-\delta_j}} \tag{118}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(2^{j-\delta'_j} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\sigma_j-\sigma_{\delta'_j}}} \right) \qquad \boxed{\begin{array}{c}\kappa \geq j-\delta'_j \\ \implies \delta'_j \geq j-\kappa\end{array}} \tag{119}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(2^{j-\delta'_j} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\ln(j)^{1/17}}} \right) \qquad \boxed{\begin{array}{c}\sigma_j - \sigma_{\delta'_j} \geq \ln(j)^{1/17} \\ \text{for all } j \geq \ell_2 \text{ by Eq. (102)}\end{array}} \tag{120}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(\frac{2^j}{e^{\ln(j)^{1/17}}} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\ln(j)^{1/17}}} \right) \qquad \boxed{\delta'_j \geq \log_2(e)\ln(j)^{1/17}} \tag{121}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \left(\frac{2^{j+1}}{e^{\ln(j)^{1/17}}} + \sum_{\kappa=j-\delta'_j}^{j} \frac{2^\kappa}{e^{\ln(j)^{1/17}}} \right) \tag{122}$$

$$= 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \sum_{\kappa=j-\delta'_j}^{j+1} \frac{2^\kappa}{e^{\ln(j)^{1/17}}} \tag{123}$$

$$\leq 2^{\ell_2+1} + \sum_{i=\ell_2}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \sum_{j=\ell_2}^{i} \gamma_j \frac{2^{j+2}}{e^{\ln(j)^{1/17}}} \qquad \boxed{\textstyle\sum_{\kappa=j-\delta'_j}^{j+1} 2^\kappa \leq 2^{j+2}} \tag{124}$$

$$= 2^{\ell_2+1} + 4 \sum_{j=\ell_2}^{\ell} \gamma_j \cdot \frac{2^j}{e^{\ln(j)^{1/17}}} \sum_{i=j}^{\ell} \frac{i^{1/2}\ln(i)}{2^i} \qquad \boxed{\begin{array}{c}\text{Lemma 3} \\ \text{sum switching}\end{array}} \tag{125}$$

$$\le 2^{\ell_2+1} + 4\sum_{j=\ell_2}^{\ell}\gamma_j\cdot\frac{2^j}{e^{\ln(j)^{1/17}}}\sum_{i=j}^{\infty}\frac{i^{1/2}\ln(i)}{2^i} \qquad \boxed{\ell<\infty} \tag{126}$$

$$\le 2^{\ell_2+1} + 16\sum_{j=\ell_2}^{\ell}\gamma_j\cdot\frac{2^j}{e^{\ln(j)^{1/17}}}\,\frac{j^{1/2}\ln(j)}{2^j} \qquad \boxed{\begin{array}{c}\text{Lemma 4 with}\\ \nu=1/2, j\ge\ell_2\ge 4\end{array}} \tag{127}$$

$$= 2^{\ell_2+1} + 16\sum_{j=\ell_2}^{\ell}\gamma_j\frac{j^{1/2}\ln(j)}{e^{\ln(j)^{1/17}}} \tag{128}$$

$$\le 2^{\ell_2+1} + 16\sum_{j=\ell_2}^{\ell}\gamma_j\,\frac{\ell^{1/2}\ln(\ell)}{e^{\ln(\ell)^{1/17}}} \qquad \boxed{\begin{array}{c}x^{1/2}\ln(x)/e^{\ln(x)^{1/17}}\\ \text{is non-decreasing}\end{array}} \tag{129}$$

$$\le 2^{\ell_2+1} + 16\,\sigma_\ell\,\frac{\ell^{1/2}\ln(\ell)}{e^{\ln(\ell)^{1/17}}}\ . \tag{130}$$

Let $\delta''_\ell \coloneqq \lceil\log_2(e)\ln(\ell)^3\rceil$. Thus there exists some sufficiently large $\ell_3\in\mathbb{N}$ s.t. for all $\ell\ge\ell_3$ it holds that

$$\sigma_\ell - \sigma_{\delta''_\ell} \ge \sigma_\ell - \delta''_\ell \ge \Big((\ell+1)^{1/2} - (\ell_2)^{1/2} - 2^{\ell_2+1}\Big)\frac{e^{\ln(\ell)^{1/17}}}{16\ell^{1/2}\ln(\ell)} - \log_2(e)\ln(\ell)^3 - 1 \ge \ln(\ell)^3\ . \tag{131}$$

Finally, we can use our last bound to obtain a contradiction. Plugging Eq. (131) into Eq. (61) yields

$$\frac{1}{\ell\ln(\ell)} \le \sum_{j=1}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \tag{132}$$

$$\le \sum_{j=1}^{\ell_3-1}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} + \sum_{j=\ell_3}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\text{split sum}} \tag{133}$$

$$\le \sum_{j=1}^{\ell_3-1}\gamma_j\sum_{\kappa=1}^{j}2^{\kappa-\ell} + \sum_{j=\ell_3}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\sigma_j-\sigma_{j-\kappa}\ge 0} \tag{134}$$

$$\le \sum_{j=1}^{\ell_3-1}\gamma_j\,2^{j+1-\ell} + \sum_{j=\ell_3}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \tag{135}$$

$$\le 2^{\ell_3+1-\ell} + \sum_{j=\ell_3}^{\ell}\gamma_j\sum_{\kappa=1}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \tag{136}$$

$$\le 2^{\ell_3+1-\ell} + \sum_{j=\ell_3}^{\ell}\gamma_j\left(\sum_{\kappa=1}^{j-1-\delta''_j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} + \sum_{\kappa=j-\delta''_j}^{j}\frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}}\right) \qquad \boxed{\text{split sum}} \tag{137}$$

$$\leq 2^{\ell_3+1-\ell} + \sum_{j=\ell_3}^{\ell} \gamma_j \left(\sum_{\kappa=1}^{j-1-\delta''_j} 2^{\kappa-\ell} + \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\sigma_j - \sigma_{j-\kappa} \geq 0} \tag{138}$$

$$\leq 2^{\ell_3+1-\ell} + \sum_{j=\ell_3}^{\ell} \gamma_j \left(2^{j-\delta''_j-\ell} + \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\textstyle\sum_{\kappa=1}^{j-1-\delta''_j} 2^{\kappa} \leq 2^{j-\delta''_j}} \tag{139}$$

$$\leq 2^{\ell_3+1-\ell} + \sum_{j=\ell_3}^{\ell} \gamma_j \left(\frac{2^{j-\ell}}{e^{\ln(\ell)^3}} + \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \right) \qquad \boxed{\delta''_j \geq \log_2(e)\ln(j)^3} \tag{140}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \sum_{j=\ell_3}^{\ell} \gamma_j \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{j-\kappa}}} \qquad \boxed{\gamma_j \leq 1} \tag{141}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \sum_{j=\ell_3}^{\ell} \gamma_j \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\sigma_j-\sigma_{\delta''_j}}} \qquad \boxed{\begin{array}{c}\kappa \geq j-\delta''_j \\ \implies \delta''_j \geq j-\kappa\end{array}} \tag{142}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \sum_{j=\ell_3}^{\ell} \gamma_j \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\kappa-\ell}}{e^{\ln(j)^3}} \qquad \boxed{\begin{array}{c}\sigma_j - \sigma_{\delta''_j} \geq \ln(j)^3 \\ \text{for all } j \geq \ell_3 \text{ by } \textsf{Eq. (131)}\end{array}} \tag{143}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \sum_{j=\ell_3}^{\ell} \gamma_j \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{j-\ell}}{e^{\ln(j)^3}} \qquad \boxed{\kappa \leq j} \tag{144}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \sum_{j=\ell_3}^{\ell} \gamma_j \sum_{\kappa=j-\delta''_j}^{j} \frac{2^{\ell-\ell}}{e^{\ln(\ell)^3}} \qquad \boxed{\begin{array}{c}2^x/e^{\ln(x)^3} \\ \text{is non-decreasing}\end{array}} \tag{145}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell}{e^{\ln(\ell)^3}} + \frac{\ell^2}{e^{\ln(\ell)^3}} \qquad \boxed{\gamma_j \leq 1} \tag{146}$$

$$\leq 2^{\ell_3+1-\ell} + \frac{2\ell^2}{e^{\ln(\ell)^3}} \qquad \boxed{\ell \geq 2} \tag{147}$$

or equivalently the contradiction

$$1 \leq \frac{\ell\ln(\ell)}{2^{\ell-\ell_3-1}} + \frac{2\ell^3\ln(\ell)}{e^{\ln(\ell)^3}} \to 0 \tag{148}$$

for $\ell \to \infty$. □

To the valiant reader that has retraced the full proof of Lemma 1 we want to put the proposition that the proof can be carried out so long as the right-hand side of the lemma has the form ${}^{2^\ell}/_{\prod_{i=0}^k \ln^{(i)}(\ell)}$ for some fixed $k \in \mathbb{N}$ where $\ln^{(i)}$ is the i-th times iterated logarithm. Towards this, we assume a slight simplification of the form $\sum_{j=1}^{\ell} \gamma_j \sum_{\kappa=1}^{j} 2^\kappa / e^{\sigma_j - \sigma_{j-\kappa}} \approx \sum_{j=1}^{\ell} 2^j \gamma_j / e^{\sigma_j} \leq_{\mathsf{io}} {}^{2^\ell}/_{\prod_{i=0}^k \ln^{(i)}(\ell)}$. We sketch a proof by induction where we go from a bound $\sigma_\ell \in \Omega(\ln^{(k+1)}(\ell))$ to $\sigma_\ell \in \Omega(\ln^{(k)}(\ell))$.
Starting out with the counter assumption $\sum_{j=1}^{\ell} 2^j \gamma_j / e^{\sigma_j} \geq_{\mathsf{abf}} {}^{2^\ell}/_{\prod_{i=0}^k \ln^{(i)}(\ell)}$ we find that the first repetition of Eq. (64) is of the form $\Theta(\ln^{(k+1)}(\ell)) = \Theta(\int {}^{1}/_{\prod_{i=0}^k \ln^{(i)}(\ell)} \mathrm{d}\ell) \leq \Theta(\sigma_\ell)$. Inserting this bound into the counter assumption gives

$$\sum_{j=1}^{\ell} 2^j \gamma_j / e^{\ln^{(k+1)}(j) \cdot \Theta(1)} = \sum_{j=1}^{\ell} 2^j \gamma_j / \ln^{(k)}(j)^{\Theta(1)} \geq {}^{2^\ell}/_{\prod_{i=0}^k \ln^{(i)}(\ell)} \tag{149}$$

$$\implies \sigma_\ell \geq {}^{\ln^{(k)}(j)^{\Theta(1)}}/_{\prod_{i=0}^k \ln^{(i)}(\ell)} \tag{150}$$

The second repetition of Eq. (64) takes the form

$$\begin{aligned}
\Theta\Big(\ln^{(k)}(\ell)\Big) &= \Theta\left(\int \frac{\ln^{(k)}(\ell)}{\prod_{i=0}^k \ln^{(i)}(\ell)} \mathrm{d}\ell\right) && (151)\\
&\leq \Theta\left(\sum_{\ell'=1}^{\ell} \ln^{(k)}(\ell') \sum_{j=1}^{\ell'} 2^{j-\ell} \gamma_j / \ln^{(k)}(j)^{\Theta(1)}\right) && (152)\\
&\leq \Theta\Big(\ln^{(k)}(\ell)^{1-\Theta(1)} \sigma_\ell\Big) && (153)\\
\implies \quad \sigma_\ell &\in \Omega\Big(\ln^{(k)}(\ell)^{\Theta(1)}\Big) && (154)
\end{aligned}$$

which is already a better bound than from the first repetition, although not quite $\Theta(\sigma_\ell) \geq \Theta(\ln^{(k)}(\ell))$.
The third repetition of Eq. (64) takes the form

$$\begin{aligned}
\Theta\Big(\ln^{(k)}(\ell)\Big) &= \Theta\left(\int \frac{\ln^{(k)}(\ell)}{\prod_{i=0}^k \ln^{(i)}(\ell)} \mathrm{d}\ell\right) && (155)\\
&\leq \Theta\left(\sum_{\ell'=1}^{\ell} \ln^{(k)}(\ell') \sum_{j=1}^{\ell'} 2^{j-\ell} \gamma_j / e^{\ln^{(k)}(j)^{\Theta(1)}}\right) && (156)\\
&\leq \Theta\Big(\ln^{(k)}(\ell) / e^{\ln^{(k)}(j)^{\Theta(1)}} \cdot \sigma_\ell\Big) && (157)\\
\implies \quad \sigma_\ell &\in \Omega\Big(e^{\ln^{(k)}(\ell)^{\Theta(1)}}\Big) \geq \Theta\Big(\ln^{(k)}(\ell)\Big) && (158)
\end{aligned}$$

which concludes the induction step.

References

[AB18] B. Aydinlioğlu and E. Bach. Affine relativization: unifying the algebrization and relativization barriers. *ACM Trans. Comput. Theory*, 10(1), January 2018.

[ABK$^+$02] E. Allender, H. Buhrman, M. Koucký, D. van Melkebeek, and D. Ronneburger. Power from random strings. In *43rd FOCS*, pages 669–678. IEEE Computer Society Press, November 2002.

[ABK$^+$06] E. Allender, H. Buhrman, M. Koucký, D. van Melkebeek, and D. Ronneburger. Power from random strings. *SIAM Journal on Computing*, 35(6):1467–1493, 2006. eprint: https://doi.org/10.1137/050628994.

[ACM$^+$21] E. Allender, M. Cheraghchi, D. Myrisiotis, H. Tirumala, and I. Volkovich. One-Way Functions and a Conditional Variant of MKTP. In M. Bojańczy and C. Chekuri, editors, *41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2021)*, volume 213 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 7:1–7:19, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021.

[AIV21] E. Allender, R. Ilango, and N. Vafa. The non-hardness of approximating circuit size. *Theory of Computing Systems*, 65(3):559–578, April 2021.

[AKR$^+$11] E. Allender, M. Koucký, D. Ronneburger, and S. Roy. The pervasive reach of resource-bounded kolmogorov complexity in computational complexity theory. *Journal of Computer and System Sciences*, 77(1):14–40, 2011. Celebrating Karp's Kyoto Prize.

[All01] E. Allender. When worlds collide: derandomization, lower bounds, and kolmogorov complexity. In R. Hariharan, V. Vinay, and M. Mukund, editors, *FST TCS 2001: Foundations of Software Technology and Theoretical Computer Science*, pages 1–15, Berlin, Heidelberg. Springer Berlin Heidelberg, 2001.

[All17] E. Allender. *The complexity of complexity.* In volume 10010. January 2017, pages 79–94.

[All20] E. Allender. The new complexity landscape around circuit minimization. In A. Leporati, C. Martín-Vide, D. Shapira, and C. Zandron, editors, *Language and Automata Theory and Applications*, pages 3–16, Cham. Springer International Publishing, 2020.

[All21] E. Allender. Vaughan jones, kolmogorov complexity, and the new complexity landscape around circuit minimization. *New Zealand Journal of Mathematics*, 52:585–604, September 2021.

[AW08] S. Aaronson and A. Wigderson. Algebrization: a new barrier in complexity theory. In R. E. Ladner and C. Dwork, editors, *40th ACM STOC*, pages 731–740. ACM Press, May 2008.

[BGS75] T. Baker, J. Gill, and R. Solovay. Relativizations of the $\mathcal{P} =?\mathcal{NP}$ question. *SIAM Journal on Computing*, 4(4):431–442, 1975. eprint: https://doi.org/10.1137/0204037.

[BM82] M. Blum and S. Micali. How to generate cryptographically strong sequences of pseudo random bits. In *23rd FOCS*, pages 112–117. IEEE Computer Society Press, November 1982.

[Cha66] G. J. Chaitin. On the length of programs for computing finite binary sequences. *J. ACM*, 13(4):547–569, October 1966.

[Cha69] G. J. Chaitin. On the simplicity and speed of programs for computing infinite sets of natural numbers. *J. ACM*, 16(3):407–422, July 1969.

[Cha75] G. J. Chaitin. A theory of program size formally identical to information theory. *J. ACM*, 22(3):329–340, July 1975.

[DH10] R. Downey and D. Hirschfeldt. *1 Algorithmic Randomness and Complexity.* 2010.

[GGM84] O. Goldreich, S. Goldwasser, and S. Micali. On the cryptographic applications of random functions. In G. R. Blakley and D. Chaum, editors, *CRYPTO'84*, volume 196 of *LNCS*, pages 276–288. Springer, Heidelberg, August 1984.

[GGS$^+$13] S. Garg, C. Gentry, A. Sahai, and B. Waters. Witness encryption and its applications. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *45th ACM STOC*, pages 467–476. ACM Press, June 2013.

[GII$^+$19] A. Golovnev, R. Ilango, R. Impagliazzo, V. Kabanets, A. Kolokolova, and A. Tal. $\mathsf{AC}^0[p]$ lower bounds against MCSP via the coin problem. In C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, editors, *ICALP 2019*, volume 132 of *LIPIcs*, 66:1–66:15. Schloss Dagstuhl, July 2019.

[GKL$^+$22] H. Goldberg, V. Kabanets, Z. Lu, and I. C. Oliveira. Probabilistic Kolmogorov Complexity with Applications to Average-Case Complexity. In S. Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 16:1–16:60, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[GM84] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[Har83] J. Hartmanis. Generalized Kolmogorov complexity and the structure of feasible computations (preliminary report). In *24th FOCS*, pages 439–445. IEEE Computer Society Press, November 1983.

[HIL$^+$99] J. Håstad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[Hir18] S. Hirahara. Non-black-box worst-case to average-case reductions within NP. In M. Thorup, editor, *59th FOCS*, pages 247–258. IEEE Computer Society Press, October 2018.

[Hir20a] S. Hirahara. Characterizing average-case complexity of PH by worst-case meta-complexity. In *61st FOCS*, pages 50–60. IEEE Computer Society Press, November 2020.

[Hir20b] S. Hirahara. Unexpected hardness results for Kolmogorov complexity under uniform reductions. In K. Makarychev, Y. Makarychev, M. Tulsiani, G. Kamath, and J. Chuzhoy, editors, *52nd ACM STOC*, pages 1038–1051. ACM Press, June 2020.

[Hir20c] S. Hirahara. Unexpected power of random strings. In T. Vidick, editor, *ITCS 2020*, volume 151, 41:1–41:13. LIPIcs, January 2020.

[Hir22a] S. Hirahara. NP-hardness of learning programs and partial MCSP. In *63rd FOCS*, pages 968–979. IEEE Computer Society Press, October 2022.

[Hir22b] S. Hirahara. Symmetry of Information from Meta-Complexity. In S. Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 26:1–26:41, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[HIR23] Y. Huang, R. Ilango, and H. Ren. Np-hardness of approximating meta-complexity: a cryptographic approach. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, pages 1067–1075, New York, NY, USA. Association for Computing Machinery, 2023.

[Hir23] S. Hirahara. Capturing one-way functions via np-hardness of meta-complexity. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, STOC 2023, pages 1027–1038, New York, NY, USA. Association for Computing Machinery, 2023.

[HN22] S. Hirahara and M. Nanashima. On worst-case learning in relativized heuristica. In *62nd FOCS*, pages 751–758. IEEE Computer Society Press, February 2022.

[IKK09] R. Impagliazzo, V. Kabanets, and A. Kolokolova. An axiomatic approach to algebrization. In M. Mitzenmacher, editor, *41st ACM STOC*, pages 695–704. ACM Press, May 2009.

[Ila20a] R. Ilango. Approaching MCSP from above and below: hardness for a conditional variant and $\mathsf{AC}^0[p]$. In T. Vidick, editor, *ITCS 2020*, volume 151, 34:1–34:26. LIPIcs, January 2020.

[Ila20b] R. Ilango. Constant depth formula and partial function versions of MCSP are hard. In *61st FOCS*, pages 424–433. IEEE Computer Society Press, November 2020.

[Imp95] R. Impagliazzo. A personal view of average-case complexity. In *Proceedings of Structure in Complexity Theory. Tenth Annual IEEE Conference*, pages 134–147, 1995.

[KC00] V. Kabanets and J. Cai. Circuit minimization problem. In *32nd ACM STOC*, pages 73–79. ACM Press, May 2000.

[Ko86] K.-I. Ko. On the notion of infinite pseudorandom sequences. *Theoretical Computer Science*, 48:9–33, 1986.

[Kol63] A. N. Kolmogorov. On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A (1961-2002)*, 25(4):369–376, 1963.

[Kol65] A. Kolmogorov. Three approaches to the quantitative definition of information. *Problemy Peredachi Informatsii*, 1:3–11, 1965.

[Lev73] L. A. Levin. Universal sequential search problems. *Problemy peredachi informatsii*, 9(3):115–116, 1973.

[Lev74] L. A. Levin. Laws of information conservation (nongrowth) and aspects of the foundation of probability theory. *Problemy Peredachi Informatsii*, 10(3):30–35, 1974.

[Lev84] L. A. Levin. Randomness conservation inequalities; information and independence in mathematical theories. *Information and Control*, 61(1):15–37, 1984.

[LOS21] Z. Lu, I. C. Oliveira, and R. Santhanam. Pseudodeterministic algorithms and the structure of probabilistic time. In S. Khuller and V. V. Williams, editors, *53rd ACM STOC*, pages 303–316. ACM Press, June 2021.

[LP20] Y. Liu and R. Pass. On one-way functions and kolmogorov complexity. In *61st FOCS*, pages 1243–1254. IEEE Computer Society Press, November 2020.

[LP21a] Y. Liu and R. Pass. Cryptography from sublinear-time average-case hardness of time-bounded kolmogorov complexity. In S. Khuller and V. V. Williams, editors, *53rd ACM STOC*, pages 722–735. ACM Press, June 2021.

[LP21b] Y. Liu and R. Pass. On the possibility of basing cryptography on $\mathsf{EXP} \neq \mathsf{BPP}$. In T. Malkin and C. Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 11–40, Virtual Event. Springer, Heidelberg, August 2021.

[LP23a] Y. Liu and R. Pass. On one-way functions and the worst-case hardness of time-bounded kolmogorov complexity. Cryptology ePrint Archive, Paper 2023/1086, 2023. https://eprint.iacr.org/2023/1086.

[LP23b] Y. Liu and R. Pass. One-way functions and the hardness of (probabilistic) time-bounded Kolmogorov complexity w.r.t. samplable distributions. In H. Handschuh and A. Lysyanskaya, editors, *CRYPTO 2023, Part II*, volume 14082 of *LNCS*, pages 645–673. Springer, Heidelberg, August 2023.

[LV08] M. Li and P. Vitányi. *An Introduction to Kolmogorov Complexity and Its Applications*. Springer New York, New York, NY, 2008.

[Mar66] P. Martin-Löf. The definition of random sequences. *Information and Control*, 9(6):602–619, 1966.

[MP24] N. Mazor and R. Pass. The Non-Uniform Perebor Conjecture for Time-Bounded Kolmogorov Complexity Is False. In V. Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, volume 287 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 80:1–80:20, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024.

[Nao91] M. Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, January 1991.

[Oli19] I. C. Oliveira. Randomness and intractability in Kolmogorov complexity. In C. Baier, I. Chatzigiannakis, P. Flocchini, and S. Leonardi, editors, *ICALP 2019*, volume 132 of *LIPIcs*, 32:1–32:14. Schloss Dagstuhl, July 2019.

[OPS19] I. C. Oliveira, J. Pich, and R. Santhanam. Hardness magnification near state-of-the-art lower bounds. In A. Shpilka, editor, *34th Computational Complexity Conference (CCC 2019)*, volume 137 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 27:1–27:29, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019.

[Rom90] J. Rompel. One-way functions are necessary and sufficient for secure signatures. In *22nd ACM STOC*, pages 387–394. ACM Press, May 1990.

[RR97] A. A. Razborov and S. Rudich. Natural proofs. *Journal of Computer and System Sciences*, 55(1):24–35, 1997.

[RS21] H. Ren and R. Santhanam. Hardness of kt characterizes parallel cryptography. In *Proceedings of the 36th Computational Complexity Conference*, CCC '21, Dagstuhl, DEU. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2021.

[RS22] H. Ren and R. Santhanam. A Relativization Perspective on Meta-Complexity. In P. Berenbrink and B. Monmege, editors, *39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022)*, volume 219 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 54:1–54:13, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022.

[Sip83] M. Sipser. A complexity theoretic approach to randomness. In *15th ACM STOC*, pages 330–335. ACM Press, April 1983.

[Sol60] R. Solomonoff. *A Preliminary Report on a General Theory of Inductive Inference*. AFOSR TN-60-1459. United States Air Force, Office of Scientific Research, 1960.

[Sol64a] R. Solomonoff. A formal theory of inductive inference. part i. *Information and Control*, 7(1):1–22, 1964.

[Sol64b] R. Solomonoff. A formal theory of inductive inference. part ii. *Information and Control*, 7(2):224–254, 1964.

[Tra84] B. Trakhtenbrot. A survey of russian approaches to perebor (brute-force searches) algorithms. *Annals of the History of Computing*, 6(4):384–400, 1984.

[Wil13] R. Williams. Improving exhaustive search implies superpolynomial lower bounds. *SIAM Journal on Computing*, 42(3):1218–1244, 2013. eprint: https://doi.org/10.1137/10080703X.

[Wil14] R. Williams. Nonuniform acc circuit lower bounds. *J. ACM*, 61(1), January 2014.

A Technical lemmas

Lemma 3 (Sum-switching). *Let $a, b, n, c \in \mathbb{N}$ be integers. Let $\mathfrak{f}, \mathfrak{g} : \mathbb{N} \to \mathbb{R}$ be functions. Then*

$$\sum_{i=a}^{n} \mathfrak{f}(i) \sum_{j=b}^{i-c} \mathfrak{g}(j) = \sum_{i=a}^{n} \sum_{j=b}^{i-c} \mathfrak{f}(i)\mathfrak{g}(j) = \sum_{j=b}^{n-c} \mathfrak{g}(j) \sum_{i=\max(j+c,a)}^{n} \mathfrak{f}(i) \ . \tag{159}$$

Fact 2 (Riemann integration)**.** *Let $a, n \in \mathbb{N}$ be integers. Let $\mathfrak{f} : \mathbb{R} \to \mathbb{R}$ be a monotonically decreasing integrable function. Then*

$$\int_{a}^{b+1} \mathfrak{f}(x)\mathrm{d}x \leq \sum_{i=a}^{b} \mathfrak{f}(i) \leq \int_{a-1}^{b} \mathfrak{f}(x)\mathrm{d}x \ . \tag{160}$$

Lemma 4 (Riemann bound). *Let $\nu \in [0, 1]$ and $j \in \mathbb{N}_{\geq 4}$.*

$$\sum_{i=j}^{\infty} \frac{i^{\nu} \ln(i)}{2^{i}} \leq 4 \cdot \frac{j^{\nu} \ln(j)}{2^{j}} \tag{161}$$

Proof.

$$-\frac{\partial}{\partial x} \frac{x^{\nu} \ln(x)}{2^{x}} = \frac{x^{\nu} \ln(x) \ln(2)}{2^{x}} - \frac{x^{\nu-1}(\nu \ln(x) - 1)}{2^{x}} \geq \frac{x^{\nu} \ln(x)}{2^{x+1}} \tag{162}$$

for all $x \gneq 0$. Because $\dfrac{x^{\nu} \ln(x)}{2^{x}}$ is monotonically decreasing for $x \geq 3$ it follows

$$\begin{aligned}
\sum_{i=j}^{\infty} \frac{i^{\nu} \ln(i)}{2^{i}} &\leq \int_{j-1}^{\infty} \frac{x^{\nu} \ln(x)}{2^{x}} \mathrm{d}x && (163) \\
&= 2 \int_{j-1}^{\infty} \frac{x^{\nu} \ln(x)}{2^{x+1}} \mathrm{d}x && (164) \\
&\leq 2 \int_{j-1}^{\infty} \left(-\frac{\partial}{\partial x} \frac{x^{\nu} \ln(x)}{2^{x}} \right) \mathrm{d}x && (165) \\
&= 2 \left[\frac{x^{\nu} \ln(x)}{2^{x}} \right]_{\infty}^{j-1} && (166) \\
&= 2 \frac{\ln(j-1)}{2^{j-1}} && (167) \\
&\leq 4 \frac{\ln(j)}{2^{j}} \ . && (168)
\end{aligned}$$

□