Lower Bounds for Levin–Kolmogorov Complexity

Nicholas Brandt

Department of Computer Science
ETH Zurich
Zurich, Switzerland

December 3, 2024

Preliminaries

Preliminaries

- **Kolmogorov complexity** $\mathsf{K}(x) := \min_{\Pi}\{|\Pi| \mid \Pi(\varepsilon) = x\}$

Preliminaries

Preliminaries

- **Kolmogorov complexity** $\mathsf{K}(x) := \min_{\Pi}\{|\Pi| \mid \Pi(\varepsilon) = x\}$
- **Levin's notion** [Lev84]
  $\mathsf{Kt}(x) := \min_{\Pi}(|\Pi| + \lceil \log_2(t) \rceil \mid \Pi(\varepsilon) = x \text{ in } t \text{ steps})$

Preliminaries

Preliminaries

- **Kolmogorov complexity** $\mathsf{K}(x) := \min_{\Pi}\{|\Pi| \mid \Pi(\varepsilon) = x\}$
- **Levin's notion** [Lev84]
  $\mathsf{Kt}(x) := \min_{\Pi}(|\Pi| + \lceil \log_2(t) \rceil \mid \Pi(\varepsilon) = x \text{ in } t \text{ steps})$
- **Decisional problem** $\mathsf{MKtP} := \{(x, k) \mid \mathsf{Kt}(x) \leq k\}$

Hardness of Kt

Hardness of Kt

Hardness of Kt

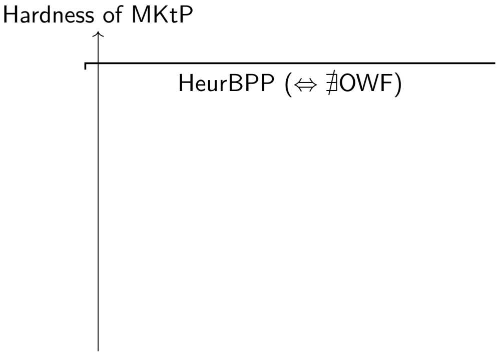
Results about the hardness of Kt [Hirahara; LiuPass]

Hardness of MKtP

Hardness of Kt

Hardness of Kt

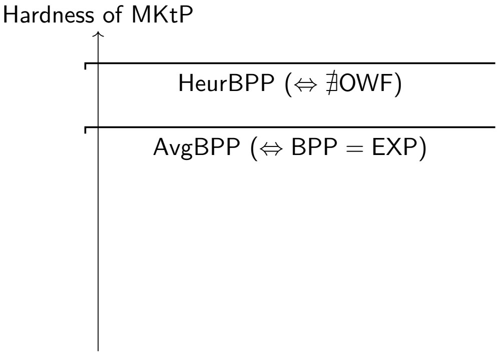
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

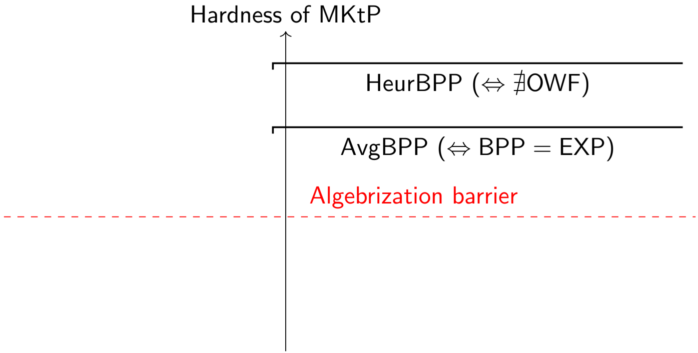
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

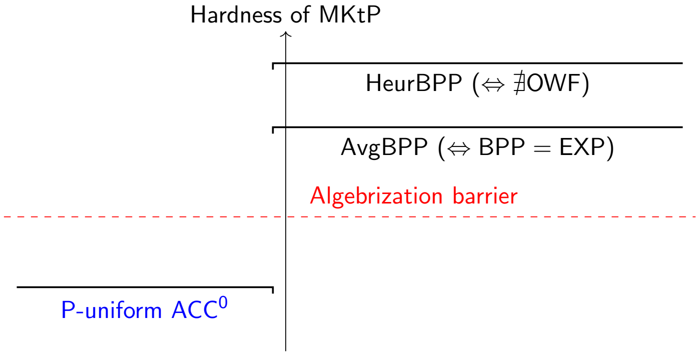
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

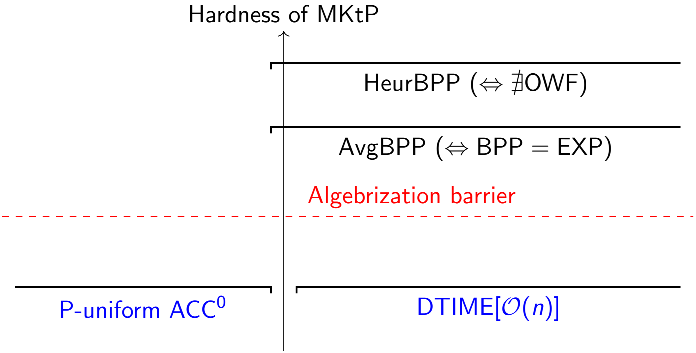
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

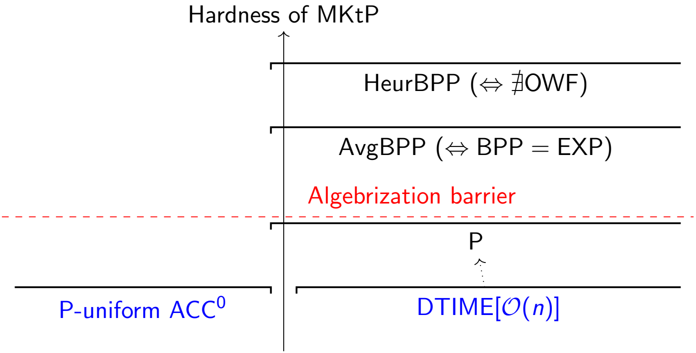
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

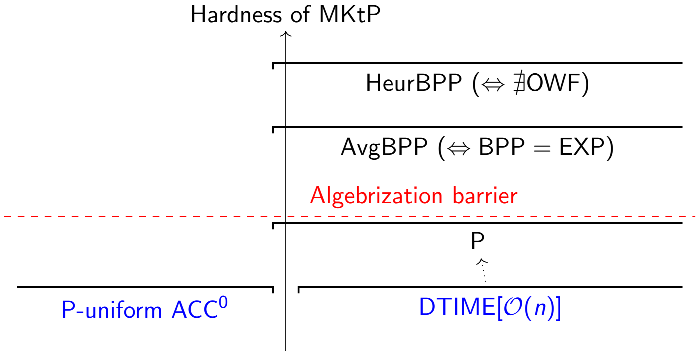
Results about the hardness of Kt [Hirahara; LiuPass]



Hardness of Kt

Hardness of Kt

Results about the hardness of Kt [Hirahara; LiuPass]



[RenSanthanam] oracle: approx. to with $(1+\epsilon)$ and BPP = EXP

Diagonalization Approach

Simple Diagonalization for K

Diagonalization Approach

Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.

Diagonalization Approach

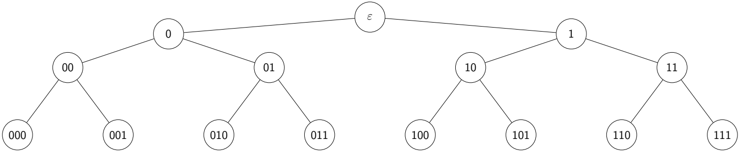
Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_K = \{x \mid K(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_K$ of length $|x| \gneq c$, output x.

Diagonalization Approach

Simple Diagonalization for K

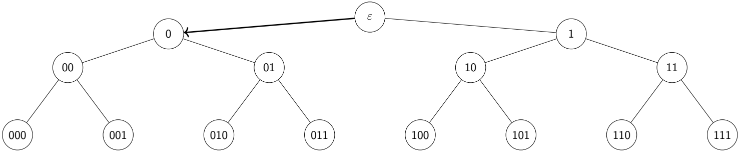
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

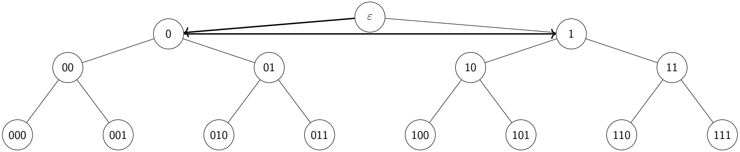
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

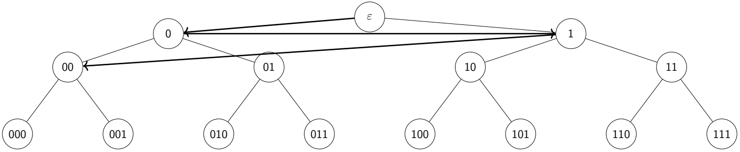
- Suppose TM (of size $c \leq 100$) decides $R_\mathsf{K} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_\mathsf{K}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

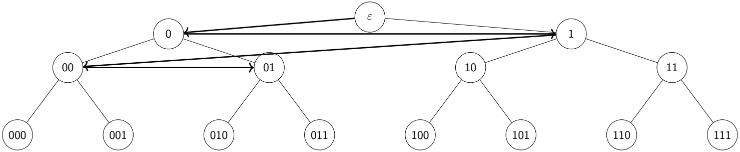
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

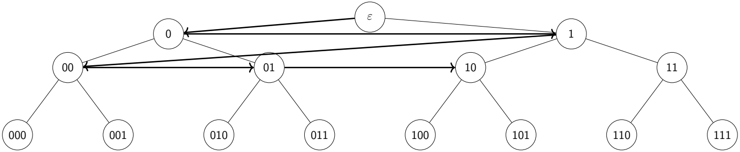
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

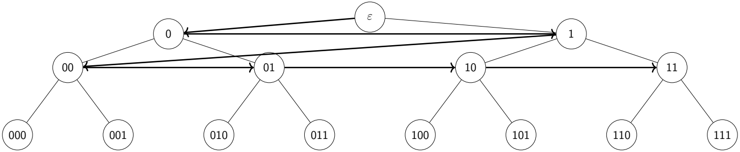
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

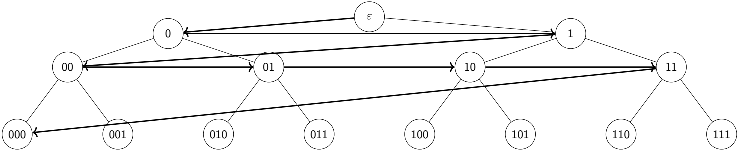
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

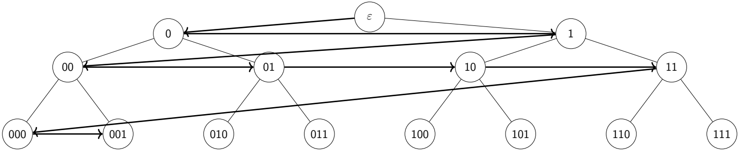
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

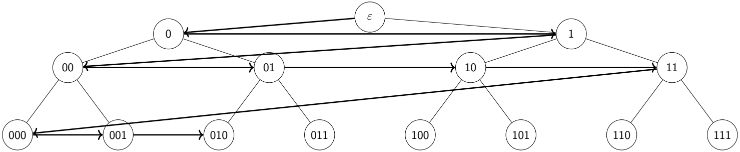
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

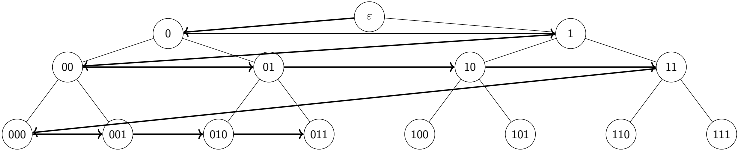
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

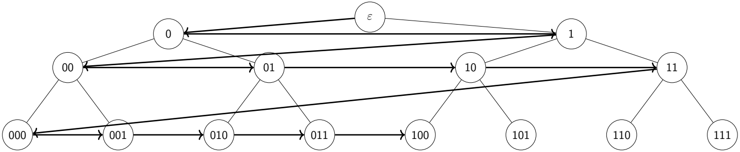
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

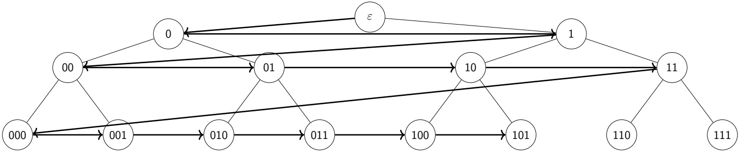
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

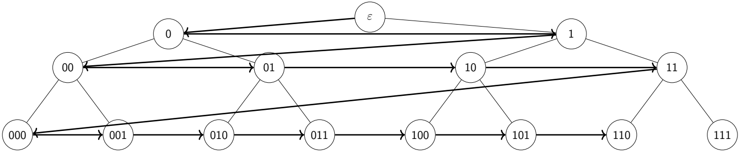
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

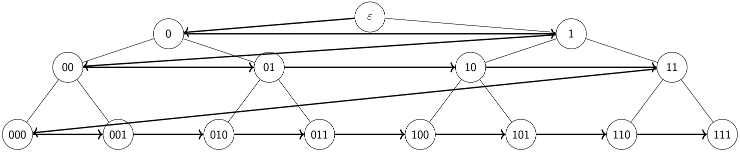
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \ngeq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

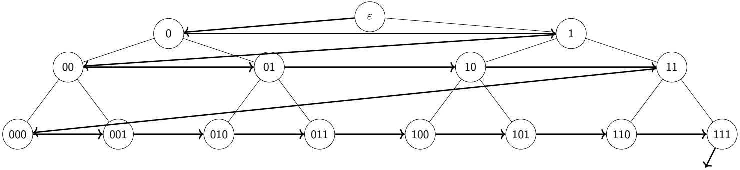
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

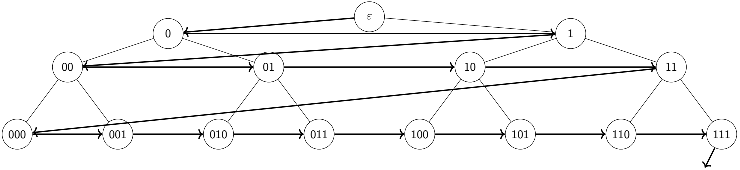
- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



Diagonalization Approach

Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.

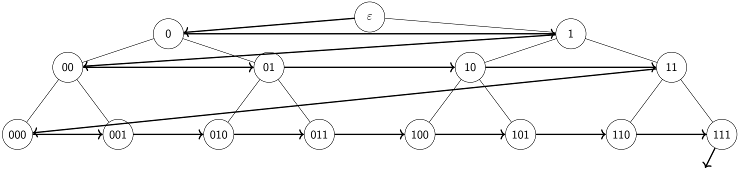


- Contradiction: $|x| \leq \mathsf{K}(x)$

Diagonalization Approach

Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gnsim c$, output x.

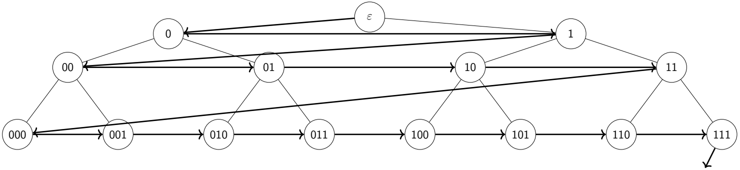


- Contradiction: $|x| \leq \mathsf{K}(x) \leq c$

Diagonalization Approach

Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.

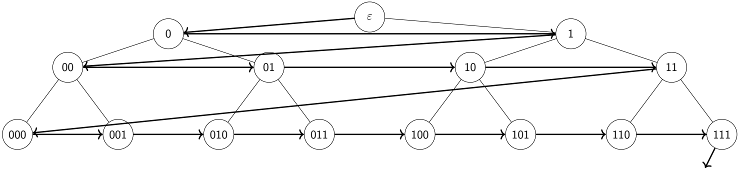


- Contradiction: $|x| \leq \mathsf{K}(x) \leq c \lneq |x|$.

Diagonalization Approach

Simple Diagonalization for K

- Suppose TM (of size $c \leq 100$) decides $R_{\mathsf{K}} = \{x \mid \mathsf{K}(x) \geq |x|\}$.
- Strategy: Compute own size c, iterate over all strings. On K-random string $x \in R_{\mathsf{K}}$ of length $|x| \gneq c$, output x.



- Contradiction: $|x| \leq \mathsf{K}(x) \leq c \lneq |x|$.
- Need structure of R_{K}: infinitely many K-random strings exist.

Diagonalization Approach

Simple Diagonalization fails for Kt

- Suppose linear-time TM (of size $c \geq 2$) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.

Diagonalization Approach

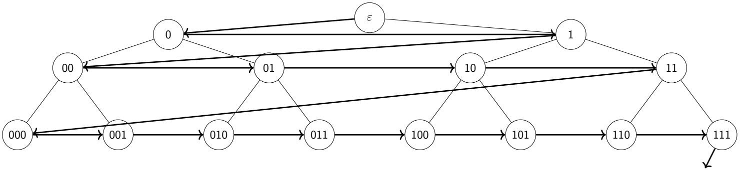
Simple Diagonalization fails for Kt

- Suppose linear-time TM (of size $c \geq 2$) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, iterate over all strings.
  On Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

Diagonalization Approach

Simple Diagonalization fails for Kt

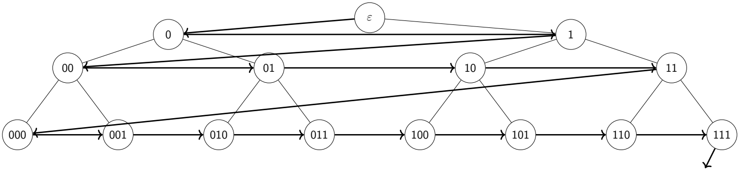
- Suppose linear-time TM (of size $c \geq 2$) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, iterate over all strings.
  On Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \not\geq c + \lceil \log(t) \rceil$, output x.



Diagonalization Approach

Simple Diagonalization fails for Kt

- Suppose linear-time TM (of size $c \geq 2$) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, iterate over all strings.
  On Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

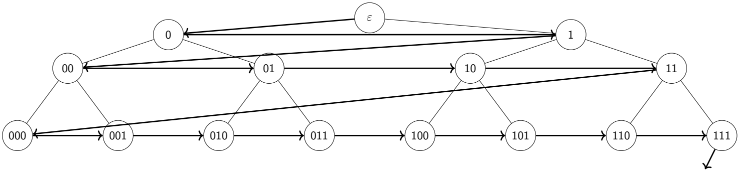


- Level ℓ costs $t \geq 2^{\ell}\ell$ time.

Diagonalization Approach

Simple Diagonalization fails for Kt

- Suppose linear-time TM (of size $c \geq 2$) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, iterate over all strings.
  On Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.



- Level ℓ costs $t \geq 2^\ell \ell$ time. Never output any string $x \in \{0,1\}^{\ell+1}$ because $|x| + \lceil \log |x| \rceil \leq c + \lceil \ell + \log(\ell) \rceil \leq c + \lceil \log(t) \rceil$.

Diagonalization Approach

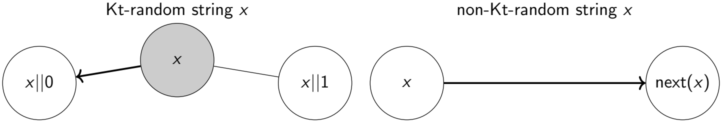
Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathrm{Kt}} = \{x \mid \mathrm{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathrm{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

Diagonalization Approach

Our Diagonalization

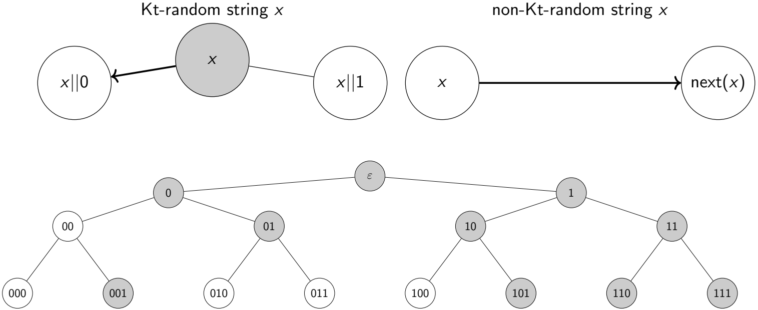
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.



Diagonalization Approach

Our Diagonalization

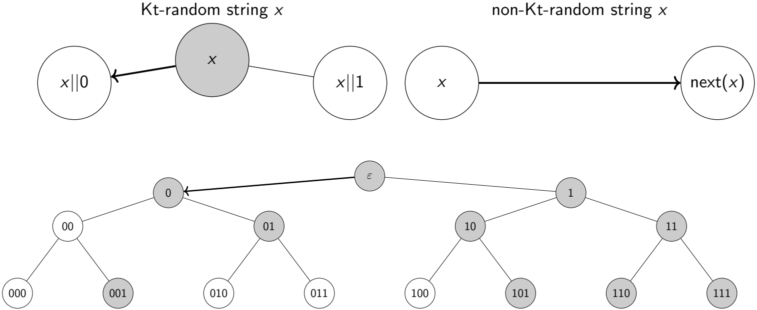
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.



Diagonalization Approach

Our Diagonalization

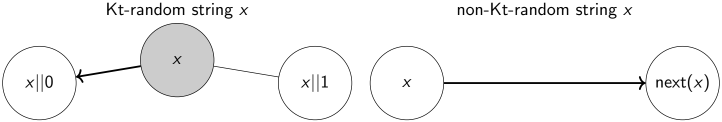
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

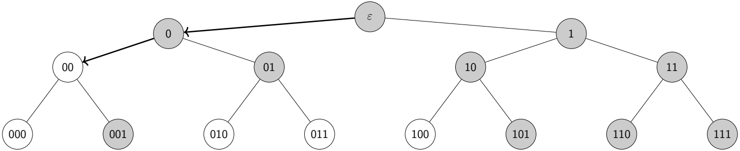


Diagonalization Approach

Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathrm{Kt}} = \{x \mid \mathrm{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathrm{Kt}}$ if $|x| + \lceil \log |x| \rceil \gnsim c + \lceil \log(t) \rceil$, output x.

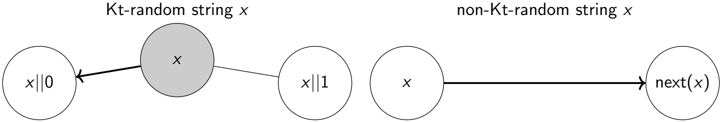


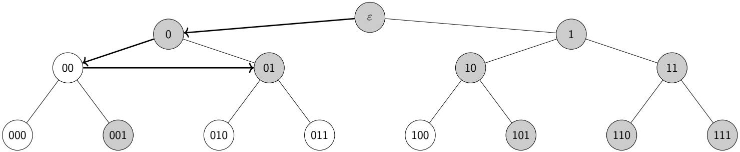


Diagonalization Approach

Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

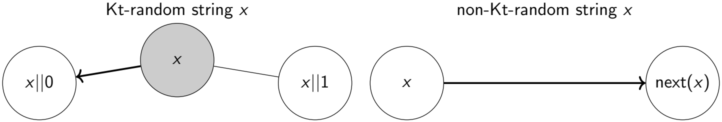


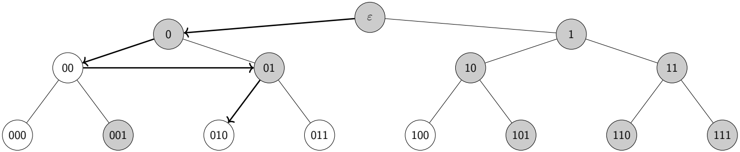


Diagonalization Approach

Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

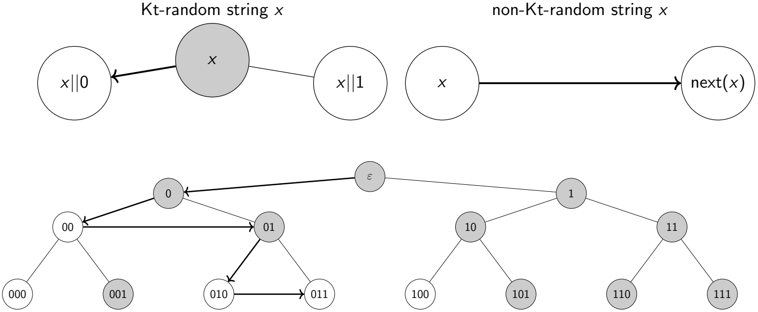




Diagonalization Approach

Our Diagonalization

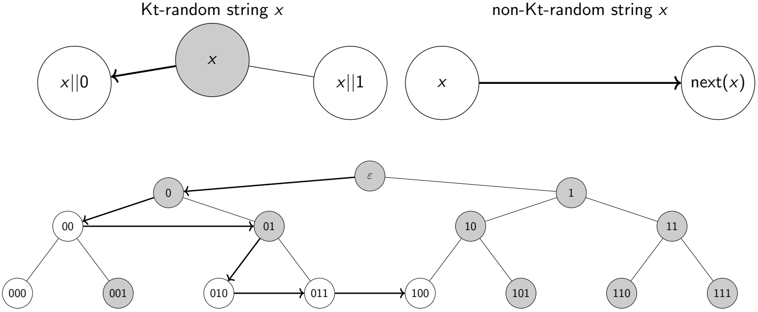
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gtrneqq c + \lceil \log(t) \rceil$, output x.



Diagonalization Approach

Our Diagonalization

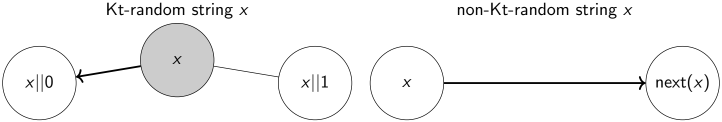
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

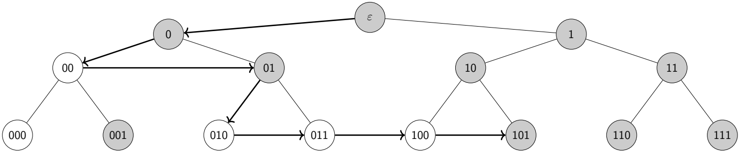


Diagonalization Approach

Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathrm{Kt}} = \{x \mid \mathrm{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathrm{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

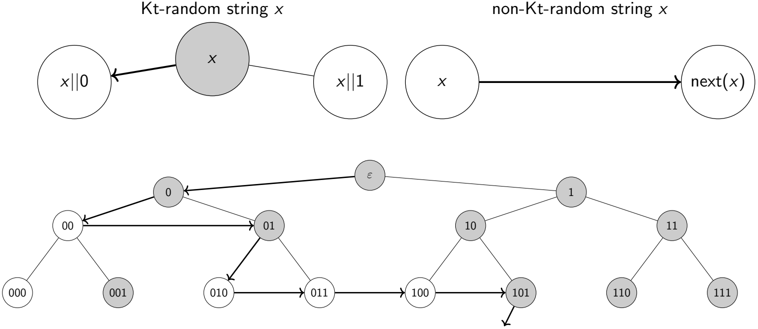




Diagonalization Approach

Our Diagonalization

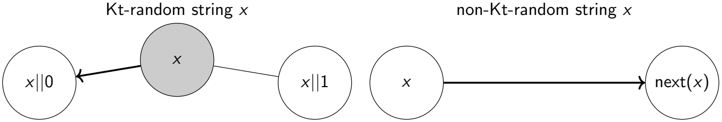
- Suppose linear-time TM (of size c) decides $R_{\mathsf{Kt}} = \{x \mid \mathsf{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathsf{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.

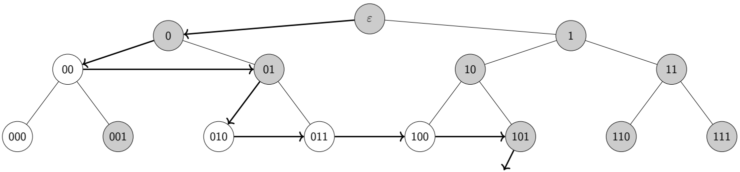


Diagonalization Approach

Our Diagonalization

- Suppose linear-time TM (of size c) decides $R_{\mathrm{Kt}} = \{x \mid \mathrm{Kt}(x) \geq |x| + \lceil \log |x| \rceil\}$.
- Strategy: Compute own size c, on Kt-random string $x \in R_{\mathrm{Kt}}$ if $|x| + \lceil \log |x| \rceil \gneq c + \lceil \log(t) \rceil$, output x.





- Need structure of R_{Kt}: a 1-Kt-random string (Chaitin's constant Ω [Cha75]); here 101...

Results

1

Results

Unconditional Results

- $\mathsf{MKtP} \notin \mathsf{DTIME}[\mathcal{O}(n)]$,

Results

Unconditional Results

- MKtP $\notin$ DTIME[$\mathcal{O}(n)$],
- MKtP $\notin$ DTIME[$\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))$] for each $k \in \mathbb{N}$.

Results

Unconditional Results

- MKtP $\notin$ DTIME[$\mathcal{O}(n)$],
- MKtP $\notin$ DTIME[$\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))$] for each $k \in \mathbb{N}$.
- MKtP $\notin$ DTIME[$\mathcal{O}(n)$] even with $\mathcal{O}(1/n^2)$ false-positive error,

---
1

Results

Unconditional Results

- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$,
- MKtP $\notin$ DTIME$[\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))]$ for each $k \in \mathbb{N}$.
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/n^2)$ false-positive error,

Conditional Result

- MKtP $\notin$ DTIME$[t(n)]$, or

1

Results

Unconditional Results

- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$,
- MKtP $\notin$ DTIME$[\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))]$ for each $k \in \mathbb{N}$.
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/n^2)$ false-positive error,

Conditional Result

- MKtP $\notin$ DTIME$[t(n)]$, <u>or</u>
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/t(n))$ false-negative error.

1

Results

Unconditional Results

- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$,
- MKtP $\notin$ DTIME$[\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))]$ for each $k \in \mathbb{N}$.
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/n^2)$ false-positive error,

Conditional Result

- MKtP $\notin$ DTIME$[t(n)]$, <u>or</u>
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/t(n))$ false-negative error.

Thus, MKtP $\in$ P $\implies$ MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ with $1/\text{poly}_{\exists}(n)$ two-sided error

---

1

Results

Unconditional Results

- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$,
- MKtP $\notin$ DTIME$[\mathcal{O}(n \log(n) \cdots \log^{(k)}(n))]$ for each $k \in \mathbb{N}$.
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/n^2)$ false-positive error,

Conditional Result

- MKtP $\notin$ DTIME$[t(n)]$, or
- MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ even with $\mathcal{O}(1/t(n))$ false-negative error.

Thus, MKtP $\in$ P $\implies$ MKtP $\notin$ DTIME$[\mathcal{O}(n)]$ with $1/\text{poly}_{\exists}(n)$ two-sided error which implies a very weak[1] form of OWF against *deterministic* inverters.

[1] OWF is computable in poly-time, and linear-time inversion is correct or $\perp$.

References I

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

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.

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

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

References II

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.