Superlinear Lower Bounds Based on ETH

András Z. Salamon
University of St Andrews

Michael Wehar
Swarthmore College

Presentation for STACS 2022

March 2022
Limited nondeterminism refers to the restriction or bounding of the amount of nondeterminism in a computation.

We introduce a notion of limited nondeterminism $\text{DTIWI}(t(n), w(n))$.

Informally, a language $L$ is in $\text{DTIWI}(t(n), w(n))$ if there exists a verification language $V$ for $L$ such that:

- $V \in \text{DTIME}(t(n))$
- witnesses are binary with length at most $w(n)$
Here are a few important points about this model of limited nondeterminism.

- we consider verifiers and witnesses instead of nondeterministic machines and guesses
- the verifiers are deterministic
- the witnesses are binary
- constant factors matter for witness length
To check if an input string $x$ is in $L$, we can check whether $x$ combined with any witness string $y$ is in the verification language $V$.

There is one important property of our model for limited nondeterminism that distinguishes it from others.

**Key Property:** Combining $x$ with a witness string $y$ results in a string with the same length as $x$.

In other words, input strings $x$ have dummy space built into them to be filled in by witness bits from $y$. 
The key property is important because it means that checking if a string of length \( n \) is in \( L \) only requires us to check if strings of length \( n \) are in \( V \).

Other notions of limited nondeterminism have verifiers that will read in larger strings which makes comparing runtime complexity between \( L \) and its verification language \( V \) complicated and messy!
Our primary question is, can we beat brute force search for problems solvable efficiently using limited nondeterminism?

In particular, we consider the following question.

-is $\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^\alpha)$ for some $\alpha < 2$?
We were unable to resolve this question so we investigated the consequences that would follow if such an $\alpha$ exists.

We might expect that:

$$\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^{\alpha}) \rightarrow \text{DTIWI}(n, 2\log(n)) \subseteq \text{DTIME}(n^{2\alpha})$$

However, we actually get a better consequence. In particular, we get that $\text{DTIWI}(n, 2\log(n)) \subseteq \text{DTIME}(n^{\beta})$ where $\beta < 2\alpha$ assuming that $\alpha < 2$.

How do we accomplish this? **Speed-up Theorems!**
We prove the following Speed-up Theorems.

**Theorem (First Speed-up Theorem)**

Let $\alpha$ such that $1 \leq \alpha < 2$ be given. If

$$\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^\alpha),$$

then for all $k \in \mathbb{N}$, $\text{DTIWI}(n, (\sum_{i=0}^{k} \alpha^i) \log(n)) \subseteq \text{DTIME}(n^{\alpha^{k+1}})$.

Notice that

$$\lim_{k \to \infty} \frac{\alpha^{k+1}}{\sum_{i=0}^{k} \alpha^i} = (\alpha - 1) \cdot \lim_{k \to \infty} \frac{\alpha^{k+1}}{\alpha^{k+1} - 1} = \alpha - 1 < 1.$$
Suppose that $\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^\alpha)$.

Consider the following.

$\text{DTIWI}(n, (1 + \alpha) \log(n)) \subseteq \text{DTIWI}(n^\alpha, \alpha \log(n))$ (1)

$\subseteq \text{DTIME}(n^{\alpha^2})$ (2)

Notice that:

$\text{DTIWI}(n, 2 \log(n)) \subseteq \text{DTIWI}(n, (1 + \alpha) \log(n))$ (3)

$\subseteq \text{DTIME}(n^{\alpha^2})$ (4)

$\not\subseteq \text{DTIME}(n^{2\alpha})$ (5)
Further, consider:

\[ \text{DTIWI}(n, (1 + \alpha + \alpha^2) \log(n)) \subseteq \text{DTIWI}(n^\alpha, (\alpha + \alpha^2) \log(n)) \]  
\[ \subseteq \text{DTIWI}(n^{\alpha^2}, \alpha^2 \log(n)) \]  
\[ \subseteq \text{DTIME}(n^{\alpha^3}) \]  

Why does this work? Because we prove two technical lemmas showing the following.

If \( \text{DTIWI}(t(n), w(n)) \subseteq \text{DTIME}(t'(n)) \), then:

\[ \text{DTIWI}(t(n), w(n) + w'(n)) \subseteq \text{DTIWI}(t'(n), w'(n)) \] [translating witness into time]
\[ \text{DTIWI}(t(f(n)), w(f(n))) \subseteq \text{DTIME}(t'(f(n))) \] [padding]
Theorem (Second Speed-up Theorem)

Let well-computable $g(n)$ and $\alpha$ such that $1 < \alpha < 2$ be given. If

$$\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^\alpha),$$

then

$$(\forall \varepsilon > 0) \text{DTIWI}(\text{poly}(n), g(n)) \subseteq \text{DTIME}(2^{(1+\varepsilon)\cdot(\alpha-1)\cdot g(n)}).$$

This follows from the First Speed-up Theorem combined with the padding lemma.
Theorem (Third Speed-up Theorem)

If for all $\alpha > 1$,

$$\text{DTIWI}(n, \log(n)) \subseteq \text{DTIME}(n^\alpha),$$

then for all $c \in \mathbb{N}$ and all $\alpha > 1$,

$$\text{DTIWI}(n, c \cdot \log(n)) \subseteq \text{DTIME}(n^\alpha).$$

This follows from the First Speed-up Theorem by choosing sufficiently small $\alpha$ and sufficiently large $k$. 
Conclusion

We investigated whether we can beat brute force search for problems solvable efficiently using limited nondeterminism.

Our Speed-up Theorems showed that slightly faster algorithms for problems with small witnesses can translate to significantly faster algorithms for problems with larger witnesses.

In our paper, we further connect these results to log-CircuitSAT, $k$-Clique, and the Exponential Time Hypothesis (ETH).
Thank you!