algorithms

MDPI

Article

The Inapproximability of k-DominatingSet for Parameterized AC^0 Circuits †

Wenxing Lai

Shanghai Key Laboratory of Intelligent Information Processing, School of Computer Science, Fudan University, Shanghai 201203, China; wenxing.lai@outlook.com or wenxinglai.fudan@gmail.com

† This paper is an extended version of our paper published in the Thirteenth International Frontiers of Algorithmics Workshop (FAW 2019), Sanya, China, 29 April–3 May 2019.

Received: 18 July 2019; Accepted: 11 October 2019; Published: 4 November 2019

check for updates

Abstract: Chen and Flum showed that any FPT-approximation of the k-Clique problem is not in para-AC^0 and the k-DominatingSet (k-DomSet) problem could not be computed by para-AC^0 circuits. It is natural to ask whether the $f(k)$-approximation of the k-DomSet problem is in para-AC^0 for some computable function f. Very recently it was proved that assuming $\mathsf{W}[1] \neq \mathsf{FPT}$, the k-DomSet problem cannot be $f(k)$-approximated by FPT algorithms for any computable function f by S., Laekhanukit and Manurangsi and Lin, seperately. We observe that the constructions used in Lin's work can be carried out using constant-depth circuits, and thus we prove that para-AC^0 circuits could not approximate this problem with ratio $f(k)$ for any computable function f. Moreover, under the hypothesis that the 3-CNF-SAT problem cannot be computed by constant-depth circuits of size $2^{\varepsilon n}$ for some $\varepsilon > 0$, we show that constant-depth circuits of size $n^{o(k)}$ cannot distinguish graphs whose dominating numbers are either $\leq k$ or $> \left(\frac{\log n}{3 \log \log n}\right)^{1/k}$. However, we find that the hypothesis may be hard to settle by showing that it implies $\mathsf{NP} \not\subseteq \mathsf{NC}^1$.

Keywords: parameterized AC^0; dominating set; inapproximability

1. Introduction

The dominating set problem is often regarded as one of the most important NP-complete problems in computational complexity. A dominating set in a graph is a set of vertices such that every vertex in the graph is either in the set or adjacent to a vertex in it. The dominating set problem is, given a graph $G = (V, E)$ and a number $k \in \mathbb{N}$, to decide the minimum dominating set of G has a size of at most k. This problem is tightly connected to the set cover problem, which was firstly shown to be NP-complete in Karp's famous NP-completeness paper [1]. Unless $\mathsf{P} = \mathsf{NP}$, we do not expect to solve this problem and its optimization variant in polynomial time. Furthermore, the set cover conjecture asserts that for every fixed $\varepsilon > 0$, no algorithm can solve the set cover problem in time $2^{(1-\varepsilon)n}\mathrm{poly}(m)$, even if set sizes are bounded by $\Delta = \Delta(\varepsilon)$ [2,3]. One way to handle NP-hard problems is to use approximation algorithms. One key measurement of an approximation algorithm for the dominating set problem is its approximation ratio, i.e., the ratio between the size of the solution output by the algorithm and the size of the minimum dominating set. It is known that greedy algorithms can achieve an approximation ratio of $\approx \ln n$ [4–8]. Though this problem has a PTAS (polynomial-time approximation scheme, an algorithm which takes an instance of an optimization problem and a parameter $\varepsilon > 0$ and, in polynomial time, approximate the problem with ratio $1 + \varepsilon$) applied to apex-minor-free graphs for contraction-bidimensional parameters [9], after a long line of works [10–14], the approximation ratio of this problem was matched by the lower bound by Dinur and Steurer [15], who followed the construction presented in Feige's work [12], showing that for every $\varepsilon > 0$, we could not obtain a $(1-\varepsilon)\ln n$-approximation for this problem unless $\mathsf{P} = \mathsf{NP}$. Besides approximation, another widely-considered technique to circumvent the intractability of NP-hard problems is parameterization. If we take the minimum solution size k as a parameter, then the brute-force algorithm can solve the k-DOMINATINGSET (k-DOMSET) problem in $O(n^{k+1})$ time. However, it is recently proved that, assuming $\mathsf{FPT} \neq \mathsf{W}[1]$, for any computable function f, there is no $f(k)$-FPT-approximation algorithm, that is, there is no approximation algorithm running in FPT-time and with a ratio of $f(k)$ [16–18].

Circuit complexity was thought to be a promising direction to solve P vs. NP. Though it has been long known that some problems, like the parity problem, are not in AC^0 [19–21], proving that non-uniform lower bounds for functions in nondeterministic complexity classes such as NP, $\mathsf{NQP} = \mathsf{NTIME}[n^{\log^{O(1)} n}]$, or NEXP is a well-known challenge. After Williams' proving that NEXP does not have $n^{\log^{O(1)} n}$-size $\mathsf{ACC} \circ \mathsf{THR}$ circuits (ACC composed with a layer of linear threshold gates at the bottom) [22,23], Murray and Williams showed that for every k, d, and m there is an e and a problem in $\mathsf{NTIME}[n^{\log^e n}]$ which does not have depth-d $n^{\log^k n}$-size $\mathsf{AC}[m]$ circuits with linear threshold gates at the bottom layer [24].

Rossman showed that the k-CLIQUE problem has no bounded-depth and unbounded fan-in circuits of size $O(n^{k/4})$ [25], which may be viewed as an AC^0 version of $\mathsf{FPT} \neq \mathsf{W}[1]$. Chen and Flum [26] showed that any FPT-approximation of the k-CLIQUE problem is not in para-AC^0. The parameterized circuit complexity class para-AC^0 introduced by Elberfeld, Stockhusen, and Tantau [27] as the AC^0 analog of the class FPT, is the class of parameterized problems computed by constant-depth circuits of size $f(k)\mathrm{poly}(n)$ for some computable function f. In the same paper, based on Rossman's result, they also showed that the k-DOMSET problem could not be computed in para-AC^0. This brings us to the main question addressed in our work: *Is there a computable function f such that the $f(k)$-approximation of k-DOMSET is in* para-AC^0*?* Furthermore, since we could enumerate every k tuple of vertices by depth-3 circuits of size $O(n^{k+1})$ using brute force, we might wonder whether it is possible to have a computable function f such that the $f(k)$-approximation of k-DOMSET could be computed by constant-depth circuits of size $n^{o(k)}$.

Our Work

In this paper, we show that for any computable function f, the $f(k)$-FPT-approximation of the k-DOMSET problem is not in para-AC^0. Furthermore, under the hypothesis that constant-depth circuits of size $2^{o(n)}$ could not compute 3-CNF-SAT (we call it AC^0-*ETH*, the constant-depth version of ETH—the exponential time hypothesis), there is no computable function f such that the $f(k)$-approximation of k-DOMSET could be computed by constant-depth circuits of size $n^{o(k)}$. Theorems 1 and 2 are direct consequences of Theorems 3 and 4, respectively.

Theorem 1. *Given a graph G with n vertices, there is no constant-depth circuits of size $f(k)n^{o(\sqrt{k})}$ for any computable function f which distinguish between:*

- *The size of the minimum dominating set is at most k,*
- *The size of the minimum dominating set is greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$.*

Note that this theorem implies the nonexistence of para-AC^0 circuits which $f(k)$-approximates the k-DOMSET problem for any computable function f. This is because if there is an $f(k)$-approximation para-AC^0 circuit $\mathsf{C}_{n,k}$ whose size is $g(k)\mathrm{poly}(n)$, we can construct a constant-depth para-AC^0 circuit $\mathsf{C}'_{n,k}$ to distinguish the size of the minimum dominating set is at most k or greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$ as follows. Compare $f(k)$ and $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$ — if $f(k)$ is smaller, we let $\mathsf{C}'_{n,k}$ be $\mathsf{C}_{n,k}$; otherwise, since $f(k) \geq \left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$, we let $\mathsf{C}'_{n,k}$ be the circuit which, using brute force, computes the size of a minimum dominating set with the depth-3 circuit of size $O(n^{k+1})$. Since $f(k) \geq \left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$, we have $f(k)^{k^3} \geq \left(\frac{\log n}{\log\log n}\right)^{2k}$; by simple calculations we know that $k \cdot \log^{2k-1} n \geq (k+1)(\log\log n)^{2k}$ for $k \geq 2, n \geq 2$, which implies $k \cdot \left(\frac{\log n}{\log\log n}\right)^{2k} \geq (k+1) \cdot \log n$, that is, $2^{k \cdot \left(\frac{\log n}{\log\log n}\right)^{2k}} \geq n^{k+1}$. Thus, we know $2^{kf(k)^{k^3}} \geq n^{k+1}$, which means the circuit is still a para-AC^0 circuit.

Hypothesis 1 (AC^0-ETH, the constant-depth version of ETH)**.** *There exists $\delta > 0$ such that no constant-depth circuits of size $2^{\delta n}$ can decide whether the* 3-CNF-SAT *instance φ is satisfiable, where n is the number of variables of φ.*

Theorem 2. *Assuming AC^0-ETH, given a graph G with n vertices, there is no constant-depth circuits of size $f(k)n^{o(k)}$ for any computable function f which distinguish between:*

- *The size of the minimum dominating set of G is at most k,*
- *The size of the minimum dominating set of G is greater than $\left(\frac{\log n}{3\log\log n}\right)^{1/k}$.*

Though AC^0-ETH seems much weaker than ETH (ETH implies the nonexistence of uniform circuits of size $2^{\delta n}$ and any depth which could compute the 3-CNF-SAT problem), we show that the hypothesis is hard to settle by proving it implies $\mathsf{NP} \not\subseteq \mathsf{NC}^1$, which, believed to be true, remains open for decades. Moreover, it is still unknown whether the weaker version, $\mathsf{NP} \not\subseteq \mathsf{ACC} \circ \mathsf{THR}$, holds or not.

Since our hard set cover instances can be easily reduced to the instances of the total dominating set problem, the connected dominating set problem and the independent dominating set problem, we can apply our inapproximability results to these variants of the dominating set problem. More discussion of the variants can be found in the work of Downey and Fellows [28] and the work of Chlebík and Chlebíková [29].

Compared with the conference version [30] of this article, the proofs of Lemmas 1–7 are firstly given here; some results are slightly improved by more careful analyses.

2. Preliminaries

We denote by $\mathbb{N}$ the set of nonnegative integers. For each $n, m \in \mathbb{N}$, we define $[n] := \{1, \ldots, n\}$ and $[n, m] := [m] \setminus [n-1]$ for $m > n > 0$. For any set A and $k \in \mathbb{N}$, we let $\binom{A}{k} := \{B \subseteq A \mid |B| = k\}$ be the set of subsets with exactly k elements of A. For a sequence of bits b, we let $b[l]$ be the l-th bit of b.

For a graph G, the set of vertices of G is denoted by V_G and the set of edges is denoted by E_G; for a vertex $v \in V_G$, we let $N_G(v) := \{u \in V_G \mid \{u, v\} \in E_G\}$ be the neighbors of v. Since a graph G is represented using a binary string, we express the bit of the edge $\{u, v\}$ by $\mathsf{bit}_G\{u, v\}$. For a bipartite graph $G = (L, R, E)$, we often tacitly represent G only using $O(|L| \cdot |R|)$ bits.

In this article, logarithms have base 2, and fractions and irrational numbers are rounded up if necessary.

2.1. Problem Definitions

The decision problems studied in this paper are listed below:

- In the k-DOMINATINGSET (k-DOMSET) problem, our goal is to decide if there is a dominating set of size k in the given graph G.
- In the k-SETCOVER problem, we are given a bipartite graph $I = (S, U, E)$ and the goal is to decide whether there is a subset X of S with cardinality k such that for each vertex v in U, there exists a vertex u in X that covers v, i.e., $\{u, v\} \in E$.
- In the k-CLIQUE problem, our goal is to determine if there is a clique of size k in the given graph G.
- In the 3-CNF-SAT problem, we are given a propositional formula φ in which every clause contains at most 3 literals and the goal is to decide whether φ is satisfiable.

We say a set cover instance $I = (S, U, E)$ has set cover number m if the size of a minimum set $X \subseteq S$ such that X could cover U is m. Similarly, we say a graph G has dominating number m if the size of a minimum dominating set of G is m.

As we mentioned, the dominating set problem is tightly connected to the set cover problem. Given a k-DOMSET instance $G = (V, E)$, we can construct a k-SETCOVER instance $I = (S, U, E')$ with $S = V$, $U = V$ and $E' = \bigcup_{\{u,v\} \in E} \{\{u_S, u_U\}, \{v_S, v_U\}, \{u_S, v_V\}, \{v_S, u_U\}\}$; here, for each vertex $v \in V$, we denote the corresponding vertices in S, U by v_S and v_U, respectively. It is quite clear that G has dominating number k if and only if I has set cover number k. Also, given a k-SETCOVER instance $I = (S, U, E)$, we can construct a k-DOMSET instance $G = (D \cup U_1 \cup U_2, C \cup E_1 \cup E_2)$ by letting $D = S$, $U_1 = U_2 = U$, $C = \{\{u, v\} \mid u, v \in D\}$, $E_1 = \{\{s, u\} \mid s \in D, u \in U_1\}$ and $E_2 = \{\{s, u\} \mid s \in D, u \in U_2\}$. It is trivial that I has set cover number k if and only if G has dominating number k. The reductions can also be found in the work of Chlebík and Chlebíková [29].

It is notable that each hard instance with gap reduced from a CLIQUE or 3-CNF-SAT instance satisfies that the size of the sets M is at most $\mathrm{poly}(N)$ where N is the size of the universe. Hence, it is safe to tacitly apply the inapproximability of k-SETCOVER to the k-DOMSET problem.

2.2. Circuit Complexity

For $n, m \in \mathbb{N}$, an n-input, m-output Boolean circuit C is a directed acyclic graph with n vertices with no incoming edges and m vertices with no outgoing edges. All nonsource vertices are called gates and are labeled with one of either $\vee, \wedge$, or $\neg$. The size of C, denoted by $|\mathsf{C}|$, is the number of vertices in it. The depth of C is the length of the longest directed path from an input node to the output node. We often tacitly identify C with the function $\mathsf{C} : \{0,1\}^n \to \{0,1\}^m$ it computes.

All the circuits considered in this paper are non-uniform and with unbounded fan-in $\wedge$ and $\vee$ gates unless otherwise stated.

The classes of AC^0, para-AC^0, and NC^1 are defined as follows:

- AC^0 is the class of problems which can be computed by constant-depth circuit families $(\mathsf{C}_n)_{n \in \mathbb{N}}$ where every C_n has size $\mathrm{poly}(n)$, and whose gates have unbounded fan-in.
- Para-AC^0 is the class of parameterized problems which can be computed by a circuit family $(\mathsf{C}_{n,k})_{n,k \in \mathbb{N}}$ satisfying that there exist $d \in \mathbb{N}$ and a computed function f such that for every $n \in \mathbb{N}, k \in \mathbb{N}$, $\mathsf{C}_{n,k}$ has depth d and size $f(k)\mathrm{poly}(n)$, and whose gates have unbounded fan-in.
- NC^1 is the class of problems which can be computed by a circuit family $(\mathsf{C}_n)_{n \in \mathbb{N}}$ where C_n has depth $O(\log n)$ and size $\mathrm{poly}(n)$, and whose gates have a fan-in of 2.

2.3. Covering Arrays

A covering array $\mathsf{CA}(N; t, p, v)$ is an $N \times p$ array A whose cells take values from a set V of size v and the set of rows of every $N \times t$ subarray of A is the whole set V^t. The smallest number N such that $\mathsf{CA}(N; t, p, v)$ exists is denoted by $\mathsf{CAN}(t, p, v)$. Covering arrays are discussed extensively since the

1990s, as they play an important role in the interaction testing of complex engineered systems. The recent discussion about the upper bounds of the size of covering arrays can be found as presented by Sarkar and Colbourn [31].

In this article, we always assume $V = \{0, 1\}$. It is noted that in Lin's work [18], a covering array $\mathsf{CA}(N; k, n, 2)$ is also called an (n, k)-universal set.

3. Introducing Gap to the k-SetCover Problem

Theorem 1 and Theorem 2 show that the para-AC^0 circuits cannot approximate the k-DomSet problem with ratio $f(k)$ for any computable function f. To achieve this, we need to introduce gaps for k-DomSet instances. In this section, we present the lemmas which allow us to introduce gaps to the k-SetCover problem, using *gap-gadgets* as presented in Lin's work [18]. Lemma 1 gives an upper bound for $\mathsf{CAN}(k, n, 2)$. The next two lemmas also follow the idea from Lin's work [18]. Lemma 2 allows us to construct gap gadgets with $h \leq \frac{\log n}{\log \log n}$ and $k \log \log n \leq \log n$. In Lemma 3, we present the construction which introduces gaps to set cover instances.

Definition 1. *A (k, n, m, ℓ, h)-Gap-Gadget is a bipartite graph $T = (A, B, E)$ satisfying the following conditions.*

(G1) *A is partitioned into $(A_1, \ldots, A_m)$ where $|A_i| = \ell$ for every $i \in [m]$.*
(G2) *B is partitioned into $(B_1, \ldots, B_k)$ where $|B_j| = n$ for every $j \in [k]$.*
(G3) *For each $b_1 \in B_1, \ldots, b_k \in B_k$, there exists $a_1 \in A_1, \ldots, a_m \in A_m$ such that a_i is adjacent to b_j for $i \in [m], j \in [k]$.*
(G4) *For any $X \subseteq B$ and $a_1 \in A_1, \ldots, a_m \in A_m$, if a_i has $k+1$ neighbors in X for $i \in [m]$, then $|X| > h$.*

Lemma 1. $\mathsf{CAN}(k, n, 2) \leq k2^k \log n$ *for $n \geq 5$.*

Proof. We let $M = 2^k \binom{n}{k} (1 - \frac{1}{2^k})^{k2^k \log n}$. Kleitman and Spencer showed that if $2^k \binom{n}{k} (1 - \frac{1}{2^k})^r < 1$, $\mathsf{CAN}(k, n, 2) \leq r$ [32]. Thus, we only need to show that $M < 1$. Since $2^x \log\left(\frac{2^x}{2^x - 1}\right) > \frac{1}{\ln 2}$ and $n > 5$, we have

$$\begin{aligned} \log M &< k + k \log n + k2^k \log n \log\left(1 - \frac{1}{2^k}\right) \\ &< k + k \log n - \frac{1}{\ln 2} k \log n \\ &< 0 \end{aligned}.$$

This implies $M < 1$. $\square$

Lemma 2. *There is a constant-depth circuit family $(\mathsf{C}_{k,n,h})_{k,n,h \in \mathbb{N}}$ which, for sufficiently large n and $k, h \in \mathbb{N}$ with $h \leq \frac{\log n}{\log \log n}$ and $k \log \log n \leq \log n$, given $S = S_1 \cup \cdots \cup S_k$ with $|S_i| = n$ for $i \in [k]$, outputs a $(k, n, n \log h, h^k, h)$-Gap-Gadget $T = (A, B, E)$ with $|A| = h^k n \log h$, $B = S$. Furthermore, $\mathsf{C}_{k,n,h}$ has size at most $kh^k n^2 \log h$ and could output whether $\boldsymbol{a}$ and b are adjacent using $O(1)$ gates, for every $\boldsymbol{a} \in A, b \in B$.*

Proof. Let $m = n \log h$. Note that $\log((h \log h) 2^{h \log h} \log m) \leq (h+2) \log h + \log \log n$, that is, $(h \log h) 2^{h \log h} \log m \leq n \leq n \log h$; by Lemma 1, we know that there exists a covering array $\mathsf{CA}(n \log h; k, n, 2)$, denoted by $\mathcal{S}$.

We partition every row of $\mathcal{S}$ into $n = \frac{m}{\log h}$ blocks so that each block has length $\log h$, interpreted as an integer in $[h]$. From the $m \times n$ numbers of $\mathcal{S}$, we could obtain an $m \times n$ matrix M by setting $M_{r,c}$ to be the c-th integer of the r-th row.

Claim. *For any $C \subseteq [n]$ with $|C| \le h$, there exists $r \in [m]$ such that $|\{M_{r,c} \mid c \in C\}| = |C|$.*

This claim says that for any $C = \{i_1, \dots, i_j\} \subseteq [n]$ for $j \le h$, there is a row r such that the i_1-th, …, i_j-th numbers of r are distinct. This is because we could choose the corresponding bits $C' = \cup_{c \in C}[(c-1)\log h + 1, c \log h]$ (since for each $c \in [n]$, the c-th number of a row is from the $\big((c-1)\log h + 1\big)$-th bit to the $(c \log n)$-th bit) of the row, with $|C'| \le h \log h$; by the property of a $\mathsf{CA}(n \log h; k, n, 2)$ covering array, there must be a row r such that for each $i_{j'} \in C$, $M_{r,i_{j'}} = j'$.

Now we construct a bipartite graph $T = (A, B, E)$ as follows.

- $A = \cup_{i \in [m]} A_i$ with each $A_i = \{\boldsymbol{a} \mid \boldsymbol{a} = (a_1, \dots, a_k), a_j \in [h] \text{ for } j \in [k]\}$;
- $B = \cup_{i \in [k]} B_i$ with $B_i = S_i$ for $i \in [k]$;
- $E = \{\{\boldsymbol{a}, b\} \mid \boldsymbol{a} \in A_i, b \in B_j \text{ and } M_{i,b} = \boldsymbol{a}[j], \text{ for } i \in [m], j \in [k]\}$, that is, for every $i \in [m], j \in [k]$ and every $\boldsymbol{a} \in A_i, b \in B_j$, if $M_{i,b} = \boldsymbol{a}[j]$ then we add an edge between $\boldsymbol{a}$ and b.

We prove that T is a $(k, n, n \log h, h^k, h)$-Gap-Gadget. It is clear that (G1) and (G2) hold for T. For (G3), given any $b_1 \in B_1, \dots, b_k \in B_k$, we know that for each $i \in [m]$, $(M_{i,b_1}, \dots, M_{i,b_k}) \in A_i$, which is adjacent to $b_1, \dots, b_k$.

If T does not satisfy (G4), then there exists $X \subseteq B$ with $|X| \le h$ such that there is $\boldsymbol{a}_1 \in A_1, \dots, \boldsymbol{a}_m \in A_m$ and $\boldsymbol{a}_i$ has at least $k+1$ neighbors in X for each $i \in [m]$. Since $|X| \le h$, we know that there is a row $r \in [m]$ such that $|\{M_{r,c} \mid c \in X\}| = X$. For this r, there exist some $j \in [k]$ such that $\boldsymbol{a}_r$ has at least 2 neighbors $b_1 \neq b_2$ in B_j. However, $\{\boldsymbol{a}_r, b_1\}$ and $\{\boldsymbol{a}_r, b_2\} \in E$ means that $b_1 = b_2 = M_{i,b_1}$. This implies $|\{M_{r,c} \mid c \in X\}| < X$, which is a contradiction.

The $\mathsf{C}_{k,n,h}$ outputs T with $kh^k n^2 \log h$ bits where whether $\boldsymbol{a}$ and b are connected is determined by

$$\mathsf{bit}_T\{\boldsymbol{a}, b\} = \begin{cases} 1 & \text{if } M_{i,b} = \boldsymbol{a}[j] \\ 0 & \text{otherwise} \end{cases}$$

for every $\boldsymbol{a} \in A_i, b \in B_j$. □

Given a set cover instance $I = (S, U, E)$, we construct the gap gadget $T = (A, B, E_T)$ with $B = S$. To use the gap gadget, we construct a new set cover instance $I' = (S', U', E')$ with $S' = S$ such that for every $X \subseteq S'$ which covers U', there exists $a_1 \in A_1, \dots, a_m \in A_m$ witnessing that there is an $X' \subseteq X$ which covers U and each vertex of which is adjacent to a_i for some $i \in [m]$.

In the following lemma, we use the hypercube set system, which is firstly presented in Feige's work [12] and is also used in [16–18]. The set $X^Y = \{f : Y \to X\}$ is considered to be all the functions from Y to X with $|X^Y| = |X|^{|Y|}$.

Lemma 3. *There is a constant-depth circuit family $(\mathsf{C}_{n,k})_{n,k \in \mathbb{N}}$ which, for each $k \in \mathbb{N}$, given a set cover instance $I = (S, U, E)$ where $S = S_1 \cup \dots \cup S_k$ and $|S_i| = n$ for $i \in [k]$ and a (k, n, m, ℓ, h)-Gap-Gadget constructed with S as Lemma 2 describes, outputs a set cover instance $I' = (S', U', E)$ with $S' = S$ and $|U'| = m|U|^{\ell}$ such that*

- *If there exists $s_1 \in S_1, \dots, s_k \in S_k$ which could cover U, then the set cover number of I' is at most k;*
- *If the set cover number of I is larger than k, then the set cover number of I' is greater than h.*

Furthermore, the circuit $\mathsf{C}_{n,k}$ has size at most $knm\ell|U|^{\ell}$ and could output whether s and f are connected with at most $(\ell + 1)$ gates.

Proof. Let $T = (A = \cup_{i\in[m]} A_i, B, E_T)$ be the (k, n, m, ℓ, h)-Gap-Gadget with $B_i = S_i$ for $i \in [k]$. $I' = (S', U', E')$ is defined as follows.

- $S' = S$;
- $U' = \cup_{i\in[m]} U^{A_i}$;
- For every $s \in S'$ and $f \in U^{A_i}$ for each $i \in [m]$, $\{s, f\} \in E'$ if there is an $a \in A_i$ such that $\{s, f(a)\} \in E$ and $\{a, s\} \in E_T$.

If there exist $s_1 \in S_1, \ldots, s_k \in S_k$ that can cover U, then we show that for each $f \in U'$, it is covered by some vertex in $C = \{s_1, \ldots, s_k\}$. Suppose $f \in U^{A_i}$. By (G3) we know that, for $s_1 \in S_1, \ldots, s_k \in S_k$, there exists $a_1 \in A_1, \ldots, a_m \in A_m$ such that a_p is adjacent to s_q for $p \in [m], q \in [k]$. Since C covers U, there must be $s_j \in C$ for some $j \in [k]$ covers $f(a_i)$. That is, we have $\{f(a_i), s_j\} \in E$ and $\{a_i, s_j\} \in E_T$, which means s_j covers f.

If the set cover number of I is greater than k, we show that for every $X \subseteq S'$ that covers U', we must have $|X| > h$.

Claim. *For any $X \subseteq S'$ that covers U', there exist $a_1 \in A_1, \ldots, a_m \in A_m$ that $|N_T(a_i) \cap X| \geq k + 1$ for every $i \in [m]$.*

Otherwise, there is some $i \in [m]$ such that for any $a \in A_i$, we have $|N_T(a) \cap X| \leq k$, which means there is some $u \in U$ not covered by $N_T(a) \cap X$ since the covering number of I is greater than k. For $f \in U^{A_i}$ such that $f(a') = u$ for any $a' \in A_i$, it is covered by S only if it is covered by some $s \in S \setminus N_T(a)$ since u can only be covered by $S \setminus N_T(a)$. However, for any $s \in S \setminus N_T(a)$, s is not a neighbor of a. That is, f is not adjacent to $S \setminus N_T(a)$, either. Hence, f is not covered by X, which is a contradiction.

With the claim, we know that for any $X \subseteq S'$ that covers U', there exist $a_1 \in A_1, \ldots, a_m \in A_m$ that a_i has $k + 1$ neighbors in X for every $i \in [m]$. With (G4), we must have $|X| \geq h$.

The $\mathsf{C}_{n,k}$ outputs I' with $knm\ell|U|^{\ell}$ bits where whether there is an edge between s and f is determined by

$$\mathsf{bit}_{I'}\{s, f\} = \bigvee_{a\in A_i} \mathsf{bit}_T\{s, a\} \wedge \mathsf{bit}_I\{s, f(a)\},$$

for every $s \in S', f \in U^{A_i}$ using at most $(\ell + 1)$ gates. □

4. Inapproximability of k-DOMINATINGSET

In this section, we show the inapproximability of the dominating set problem by proving Theorems 3 and 4. To show Theorem 3, we have Lemmas 4 and 5. Lemma 4 follows the idea in recent papers [17,18], presenting the circuits that output a $\binom{k}{2}$-SETCOVER instance given a k-CLIQUE instance as input. With Lemma 4, Lemma 5 introduces circuits reducing k-CLIQUE instances to set cover instances with gaps. To prove Theorem 4, Lemma 6 (firstly shown by Pătraşcu and Williams [33]) is used to prove the inapproximability of set cover problem using constant-depth $n^{o(k)}$ circuits, assuming Hypothesis 1. At the end of this section, we show that Hypothesis 1 may be hard to settle by showing that it implies $\mathsf{NP} \not\subseteq \mathsf{NC}^1$, which has remained open for decades.

4.1. The Unconditional Inapproximability of k-DOMINATINGSET

Now we give the circuits which reduce k-CLIQUE instances to $\binom{k}{2}$-SETCOVER instances, and introduce gaps to them. Finally we use Rossman's result [25], i.e., the unconditional lower bounds of the size of constant-depth circuits determining the k-CLIQUE problem, to show the inapproximability of the k-DOMSET problem.

Lemma 4. *There is a $(\mathsf{C}_{n,k})_{n,k\in\mathbb{N}}$ circuit family which, given a k-CLIQUE instance G with $|V_G| = n$, outputs a set cover instance $I = (S, U, E)$ with $|U| \leq k^3 \log n$ and $S \leq \binom{k}{2}\binom{n}{2}$ such that G contains a k-clique if and only if the set cover number of I is at most $\binom{k}{2}$. Furthermore, $\mathsf{C}_{n,k}$ has constant depth and size at most $k^5 n^2 \log n$.*

Proof. Firstly, we construct $G' = (V_1 \cup \cdots \cup V_k, E')$, a k-colored version of G as follows. Let each $V^{(i)}$ be a copy of V and for every $v \in V_G$, we call the corresponding vertex in $V^{(i)}$ by $v^{(i)}$; let $E' = \cup_{1\leq i<j\leq k} E_{i,j}$ with $E_{i,j} = \{\{u^{(i)}, v^{(j)}\} \mid \{u, v\} \in E_G\}$. Note that each $V^{(i)}$ is an independent set for $i \in [k]$ and G' contains a k-clique if and only if G contains a k-clique.

Now we construct the the set cover instance $I = (S, U, E)$ according to G' in the following way. Given $v \in G'$, we denote by $\boldsymbol{b}(v)$ the bit representation of v. Note that when i is fixed, every vertex in V_i could be determined using $\log n$ bits.

- $S = E' = \cup_{1\leq i<j\leq k} E_{i,j}$;
- $U = \cup_{i\in[k]} U_i$ with $U_i = \{(f^{(i)}, l) \mid f^{(i)} : \{0, 1\} \to [k] \setminus \{i\}, l \in [\log n]\}$;
- For every $i \in [k]$, we connect every $(f^{(i)}, l) \in U_i$ to each $\{v_i, v_j\} \in S$, with $v_i \in V_i, v_j \in V_j$, such that $f^{(i)}(\boldsymbol{b}(v_i)[l]) = j$.

Suppose that there is a k-clique in G' with vertices $u_1 \in V_1, \ldots, u_k \in V_k$. We claim that $\{\{u_i, u_j\} \mid 1 \leq i < j \leq k\}$ covers U. This is because for any $(f^{(i)}, l) \in U_i$, we have $f^{(i)}(\boldsymbol{b}(u_i)[l]) \in [k] \setminus \{i\}$ and thus, it is covered by $\{u_i, u_j\}$.

If there is $X \subseteq S$ covers U with cardinality at most $\binom{k}{2}$, then we show that there is a k-clique in G'. Firstly, $|X \cap E_{i,j}| = 1$ for $1 \leq i < j \leq k$. Otherwise, let $f^{(i)}(0) = f^{(i)}(1) = j$ and for any $l \in [\log n]$, $(f^{(i)}, l) \in U_i$ is not covered by X.

Now we let X be the vertices $e_{i,j} \in E_{i,j}$ for every $1 \leq i < j \leq k$. For each $i \in [k]$ and distinct $j, j' \in [k] \setminus \{i\}$, we let $e_{i,j} = \{v, u^{(j)}\}$ and $e_{i,j'} = \{v', u^{(j')}\}$. We claim that $v = v'$. Otherwise, there must be a bit $l \in [\log n]$ such that $\boldsymbol{b}(v)[l] \neq \boldsymbol{b}(v')[l]$. Without loss of generality, we assume $\boldsymbol{b}(v)[l] = 0, \boldsymbol{b}(v')[l] = 1$. Now we take $f^{(i)}$ such that $f^{(i)}(0) = j'$ and $f^{(i)}(1) = j$. Then $(f^{(i)}, l)$ is not covered by X, which is a contradiction.

Hence, for every $i \in [k]$, we could safely take the vertex $v^{(i)} \in V_i$ such that $v^{(i)}$ is in the edge $e_{i,j} \in X \cap E_{i,j}$ for arbitrary j as the i-th vertex of the k-clique.

The $\mathsf{C}_{n,k}$ outputs I' with at most $k^5 n^2 \log n$ bits where whether $e_{u^{(i)},v^j j}$ and $(f^{(i)}, l)$ is connected is determined by

$$\mathsf{bit}_I\{e_{u^{(i)},v^{(j)}}, (f^{(i)}, l)\} = \begin{cases} 1, & \text{if } f^{(i)}(\boldsymbol{b}(u^{(i)})[l]) = j \\ 0, & \text{otherwise} \end{cases}$$

for every $1 \leq i < j \leq k$, $e_{u^{(i)},v^{(j)}} \in E_{i,j}$ and every $(f^{(i)}, l) \in U_i$. Hence, $\mathsf{C}_{n,k}$ is with each output gate of depth at most 3 and of size at most $k^5 n^2 \log n$. □

Lemma 5. *There is a $(\mathsf{C}_{n,k})_{n,k\in\mathbb{N}}$ circuit family which, given a k-CLIQUE instance G with $|V_G| = n$, could output a set cover instance $I = (S, U, E)$ with $|U| \leq n^5$ and $S \leq \binom{k}{2}\binom{n}{2}$ such that*

- *If G contains a k-clique, then the set cover number of I is at most $\binom{k}{2}$;*
- *If G contains no k-clique, then the set cover number of I is greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$.*

Furthermore, $\mathsf{C}_{n,k}$ has constant depth and size at most $n^7 \log n$.

Proof. By Lemma 4, we can construct a $\binom{k}{2}$-SetCover instance $I = (S, U, E)$ with $|S| \le \binom{k}{2}\binom{n}{2}$ and $|U| \le k^3 \log n$, using a constant-depth circuit of size at most $k^5 n^2 \log n$. Let $m = \binom{n}{2}$. By Lemma 2, we can construct a $(\binom{k}{2}, m, m \log h, h^{\binom{k}{2}}, h)$-Gap-Gadget T with $h = (\frac{\log m}{\log\log m})^{1/\binom{k}{2}}$ given S, using a constant-depth circuit of size at most $O(n^2 \log^2 n)$. By Lemma 3, we could have a constant-depth circuit of size at most $n^7 \log n$ which computes a set cover instance $I' = (S', U', E')$ with $S' = S$, $|U'| \le m \log h (k^3 \log n)^{h^{\binom{k}{2}}}$ such that

- If G contains a k-clique, then the set cover number of I is at most $\binom{k}{2}$;
- If G contains no k-clique, then the set cover number of I is greater than $(\frac{\log m}{\log\log m})^{1/\binom{k}{2}} \ge \left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$.

Here, since $(k^3 \log n)^{\frac{\log m}{\log\log m}} \le (k^3 \log n)^{\frac{2\log n}{\log\log n}} \le (k^{\frac{\log n}{\log\log n}})^6 \cdot n^2 \le n^{\frac{6k}{\log\log n}} \cdot n^2 \le n^{2+o(1)}$, we can conclude that $|U'| \le m \log h (k^3 \log n)^{h^{\binom{k}{2}}} \le n^2 \log n \cdot n^{2+o(1)} \le n^{4+o(1)} < n^5$. $\square$

Theorem 3. *Given a set cover instance $I = (S, U, E)$ with $n = |S| + |U|$, for $k > 28$, any constant-depth circuit of size $O(n^{\frac{\sqrt{k}}{20}})$ cannot distinguish between*

- *The set cover number of I is at most k, or*
- *The set cover number of I is greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$.*

Proof. Rossman showed that for every $k \in \mathbb{N}$, the k-Clique problem on n-vertex graphs requires constant-depth circuits of size $\omega(n^{\frac{k}{4}})$ [25]. Now if there is a constant-depth circuit $\mathsf{C}_{n,k}$ of size $O(n^{\frac{\sqrt{k}}{20}})$ that could distinguish between the set cover number of I where $|V_I| = n$ is at most k or greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$, then by Lemma 5, given $k \in \mathbb{N}$ and a graph G with $|V_G| = n$, we can construct a set cover instance I with vertex number at most $2n^5$ satisfying that if G has a k-clique then the set cover number of I is at most $\binom{k}{2}$ and otherwise it is greater than $\left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$—we could use $\mathsf{C}_{2n^5,\binom{k}{2}}$ to decide whether the set cover number of I is either $\le \binom{k}{2}$ or $> \left(\frac{\log n}{\log\log n}\right)^{1/\binom{k}{2}}$. The circuits are of size $O((n^5)^{\sqrt{\binom{k}{2}}/20}) + O(n^7 \log n) = O(n^{\frac{k}{4}})$ when $k > 28$, which contradicts the result shown by Rossman [25]. $\square$

Note that Theorem 3 implies Theorem 1 since for every set cover instance $I = (S, U, E)$ we can construct a dominating set instance $I' = (S \cup U, E \cup \{\{u, v\} \mid u, v \in S\})$ simply by adding edges to S so that it becomes a clique. Then the dominating number of I' is the same as the set cover number of I.

4.2. The Inapproximability of k-DominatingSet Assuming AC^0-ETH

Next we show the $f(k)$-inapproximability of the set cover problem for constant-depth circuits of size $n^{o(k)}$ for any computable function f, assuming AC^0-ETH. To achieve this, we use Lemma 6 to reduce 3-CNF-SAT formulas to set cover instances with gaps.

Lemma 6. *There is a circuit family $(\mathsf{C}_{n,k})_{n,k\in\mathbb{N}}$ which for every $k \in \mathbb{N}$, given a* 3-CNF-SAT *instance φ with n variables where n is much larger than k, outputs $N \le 2^{\frac{11n}{2k}}$ and a set cover instance $I = (S, U, E)$ satisfying*

- $|S| + |U| \le N$;
- *If φ is satisfiable, then the set cover number of I is at most k;*
- *If φ is not satisfiable, then the set cover number of I is greater than* $\left(\frac{\log N}{3\log\log N}\right)^{1/k}$*;*

Furthermore, $\mathsf{C}_{n,k}$ *has constant depth and size at most* $2^{\frac{11n}{2k}}$*.*

Proof. Firstly, we construct a set cover instance $I' = (S', U', E')$ whose set cover number is k if and only if φ is satisfiable, as follows.

Partition n variables into k parts and each part has n/k variables. We let $S' = S_1 \cup \cdots \cup S_k$ and for $i \in [k]$, each S_i be the set of all the assignments of the variables from the i-th part. Thus, $|S_i| = 2^{n/k}$ for each $i \in [k]$. Let $U' := \{C \mid C \in \varphi\} \cup \{x_i \mid i \in [k]\}$ be the clauses of φ and vertices $x_1, \ldots, x_k$. We define $E' := \cup_{i\in[k]} \left(\{\{s, C\} \mid s \text{ satisfies } C \text{ for } s \in S_i, C \in U'\} \cup \{\{s, x_i\} \mid s \in S_i\}\right)$ the edges connecting each $s \in S_i$ with x_i and every $C \in U'$ such that s satisfies C, for each $i \in [k]$.

It is clear that if φ is satisfied by assignment σ, then we know that there are $s_1 \in S_1, \ldots, s_k \in S_k$ such that we can combine $s_1, \ldots, s_k$ to get the σ, satisfying φ. Now suppose $s_1 \in S_1, \ldots, s_k \in S_k$ can cover U'. Note that the set cover number of I' cannot be less than k because of the existence of $x_1, \ldots, x_k$. Since the different sets $S_1, ..., S_k$ of variables are pairwise disjoint, we could simply combine the assignments $s_1, \ldots, s_k$, which together satisfy all the clauses, to have the assignment satisfying φ.

We have I' with $|S'| = k2^{n/k}$ and $|U'| \le \binom{n}{3} + k \le 2n^3$ and let $m = 2^{n/k}$. By Lemma 2, there is a constant-depth circuit which can compute a $(k, m, m\log h, h^k, h)$-Gap-Gadget T with $h = \sqrt[k]{\frac{\log m}{\log\log m}} \ge \left(\frac{\log N}{3\log\log N}\right)^{1/k}$ which has size $O(k^2 m^2 h^{2k+1}) = O(k^2 m^2 \log^3 m)$. By Lemma 3, there is a constant-depth circuit C that can construct a set cover instance $I = (S, U, E)$ which, given I' and T such that

- If φ is satisfiable, then the set cover number of I is at most k;
- If φ is unsatisfiable, then the set cover number of I is greater than h;
- $S = S'$, $|U| = m\log h |U'|^{h^k} \le \frac{1}{k} m(\log\log m - \log\log\log m)(2n^3)^{\frac{\log m}{\log\log m}} \le m \cdot \log\log m \cdot (2^{\frac{\log m}{\log\log m}} + n^{\frac{3n}{k(\log n - \log k)}}) \le m \cdot \log\log m \cdot (2^{\frac{\log m}{\log\log m}} + n^{\frac{3n}{k}}) = m^{4+o(1)}$.

Thus, $|S| + |U| = km + m^{4+o(1)} \le 2^{\frac{11n}{2k}} = N$. Furthermore, C has size at most $(h^k + 1)(km \cdot m^{4+o(1)}) = O(m^{5+o(1)}) = 2^{\frac{11n}{2k}}$. □

Theorem 4. *Assuming* AC^0*-ETH, there is* $\varepsilon > 0$ *such that, given a set cover instance* $I = (S, U, E)$ *with* $n = |S| + |U|$*, any constant-depth Boolean circuit of size* $n^{\varepsilon k}$ *cannot distinguish between*

- *The set cover number of I is at most k, or*
- *The set cover number of I is greater than* $\left(\frac{\log n}{3\log\log n}\right)^{1/k}$*.*

Proof. By AC^0-ETH, there exists $\delta > 0$ such that no constant-depth circuits of size $2^{\delta n}$ can decide whether the 3-CNF-SAT instance φ is satisfiable where n is the number of variables of φ. For every 3-CNF-SAT formula φ, there is a constant-depth circuit $\mathsf{C}_{n,k}$ of size $2^{\frac{11n}{2k}}$ which, given φ, computes a set cover instance $I = (S, U, E)$ with $|S| + |U| \le N$ for $N \le 2^{\frac{11n}{2k}}$ whose set cover number is either at most k or greater than $\left(\frac{\log N}{3\log\log N}\right)^{1/k}$ by Lemma 6. Now take $\varepsilon = \delta/12$.

If a constant-depth circuit C_n of size $n^{\varepsilon k}$ could distinguish between the set cover number of I being at most k or greater than $\left(\frac{\log n}{3\log\log n}\right)^{1/k}$ where n is the vertex number of the given set cover instance I, then we could use $\mathsf{C}_{\binom{N}{2}}$ to determine the set cover number of I is whether at most k, i.e., to decide if φ is satisfiable. The used circuits have size at most $2^{\frac{11n}{2k}} + (N^2)^{\varepsilon k} \leq 2^{12\varepsilon n} = 2^{\delta n}$, which contradicts AC^0-ETH. □

Using the same trick for Theorem 1, we know that Theorem 4 implies Theorem 2.

4.3. The Difficulty of Proving AC^0-ETH

Though AC^0-ETH seems much weaker than ETH, we find that it is still very hard to settle by showing Theorem 5, i.e., AC^0-ETH implies $\mathsf{NP} \not\subseteq \mathsf{NC}^1$. Firstly, we show the trade off between depth compression and size expansion when simulating NC^1 circuits using constant-depth ones. Then we prove the Theorem 5 by showing that AC^0-ETH implies that 3-CNF-SAT $\notin \mathsf{NC}^1$.

Lemma 7. *For every $L \in \mathsf{NC}^1$, i.e., there exists $c \in \mathbb{N}$ such that L could be computed by a family of circuits $(\mathsf{C}_n)_{n\in\mathbb{N}}$ such that C_n has size at most n^c and depth at most $c\log n$, there exists $d \in \mathbb{N}$ such that there is a family of circuits $(\mathsf{C}'_n)_{n\in\mathbb{N}}$ which satisfies*

- *$s \in L$ if and only if $\mathsf{C}'_{|s|}$ outputs 1;*
- *C'_n has depth d and size at most $n^{3c/2}(2^{n^{2c/d}+1}+1)$.*

Proof. We show that for every $n \in \mathbb{N}$, C_n could be simulated by a circuit $\mathsf{C}_n{}'$ that has depth d and size $n^{3c/2}(2^{n^{2c/d}+1}+1)$. Suppose C_n has size n^c and depth $c\log n$ (otherwise, we could add dummy gates to C_n). For every gate σ of depth $\frac{c\log n}{\frac{1}{2}d}$, let f_σ be the Boolean function computed by σ. Note that f_σ has at most $n^{2c/d}$ input bits, denoted by $b_1,\ldots,b_{n^{2c/d}}$ since C_n is of fan-in 2. Now we could replace σ using brute force by

$$\bigvee_{f_\sigma(\ell)=1,\ell\in\{0,1\}^{n^{2c/d}}} \bigwedge_{i\in[n^{2c/d}]} \beta_i$$

where $\beta_i = b_i$ if $\ell[i] = 1$ and $\beta_i = \neg b_i$ if $\ell[i] = 0$. That is, σ could be simulated by a 2-depth circuit which has size at most $2^{n^{2c/d}+1}+1$.

Assume for every $l \in [d/2-1]$, every gate σ of depth $\frac{l\cdot 2c\log n}{d}$ could be replaced by a $2l$-depth circuit $\mathsf{C}^{(\sigma)}$ which has size $n^{\frac{(l-1)\cdot 3c}{d}}(2^{n^{2c/d}+1}+1)$. Now we could simulate each gate γ of depth $\frac{(l+1)\cdot 2c\log n}{d}$, whose output is determined by the gates $\sigma_1,\ldots,\sigma_{n^{2c/d}}$ from the $\frac{l\cdot 2c\log n}{d}$-th layer, in the similar way. That is, σ is replaced by

$$\bigvee_{f_\sigma(\ell)=1,\ell\in\{0,1\}^{n^{2c/d}}} \bigwedge_{i\in[n^{2c/d}]} \mathsf{C}^{(i)}$$

where $\mathsf{C}^{(i)} = \mathsf{C}^{(\sigma_i)}$ if $\ell[i] = 1$ and $\mathsf{C}^{(i)} = \neg\mathsf{C}^{(\sigma_i)}$ if $\ell[i] = 0$. Now, $\mathsf{C}^{(\gamma)}$ has depth $2(l+1)$ and size at most $(2^{n^{2c/d}+1}+1) + n^{2c/d}\cdot n^{\frac{(l-1)\cdot 3c}{d}}(2^{n^{2c/d}+1}+1) \leq n^{\frac{l\cdot 3c}{d}}(2^{n^{2c/d}+1}+1)$.

Thus, the output gate of C_n could be simulated by a depth d circuit whose size is at most $n^{3c/2}(2^{n^{2c/d}+1}+1)$. □

Theorem 5. AC^0*-ETH implies* $\mathsf{NP} \not\subseteq \mathsf{NC}^1$.

Proof. We show that AC^0-ETH implies 3-CNF-SAT $\notin \mathsf{NC}^1$. If there exists $c \in \mathbb{N}$ such that 3-CNF-SAT could be computed by a family of circuits $(\mathsf{C}_n)_{n\in\mathbb{N}}$ satisfying C_n has size at most n^c and depth at most

$c \log n$. By Lemma 7, 3-CNF-SAT could be computed by $5c$-depth, size $n^{3c/2}(2^{n^{2/5}+1}+1) = O(2^{\sqrt{n}})$ circuits for sufficiently large n, which contradicts AC^0-ETH. □

5. Conclusions and Open Questions

We have presented that para-AC^0 circuits could not approximate the k-DOMSET problem with ratio $f(k)$ for any computable function f. With the hypothesis that the 3-CNF-SAT problem cannot be computed by constant-depth circuits of size $2^{\delta n}$ for some $\delta > 0$, we could show that constant-depth circuits of size $n^{o(k)}$ cannot distinguish graphs whose dominating numbers are either $\leq k$ or $> \left(\frac{\log n}{3 \log \log n}\right)^{1/k}$.

A natural question is to settle the hypothesis, which may be hard since we show that it implies $\mathsf{NP} \not\subseteq \mathsf{NC}^1$. Another question is to ask: Are constant-depth circuits of size $n^{o(k)}$ unable to approximate dominating number with ratio $f(k)$ for any computable function f *without assuming* AC^0-*ETH*? Sparsification is one of the key techniques when ETH is involved. Could sparsification be implemented using constant-depth circuits with size $2^{\varepsilon n}$ for any $\varepsilon > 0$? Moreover, we could have more results assuming the set cover conjecture [3]. Can we prove the inapproximability of the set cover problem based on this conjecture?

Funding: This research was funded by National Natural Science Foundation of China under grant 61872092.

Acknowledgments: I thank Yijia Chen for his valuable advice and help when preparing this work. I am grateful to Bundit Laekhanukit and Chao Liao for many helpful discussions and comments. I also thank the anonymous referees for their detailed comments.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Karp, R.M. Reducibility among Combinatorial Problems. In *Complexity of Computer Computations*; Miller, R.E.; Thatcher, J.W.; Bohlinger, J.D., Eds.; The IBM Research Symposia Series; Springer: Boston, MA, USA, 1972; pp. 85–103.
2. Cygan, M.; Dell, H.; Lokshtanov, D.; Marx, D.; Nederlof, J.; Okamoto, Y.; Paturi, R.; Saurabh, S.; Wahlström, M. On Problems As Hard As CNF-SAT. *ACM Trans. Algorithms* **2016**, *12*, 41:1–41:24. [CrossRef]
3. Krauthgamer, R.; Trabelsi, O. The Set Cover Conjecture and Subgraph Isomorphism with a Tree Pattern. In Proceedings of the 36th International Symposium on Theoretical Aspects of Computer Science (STACS 2019), Berlin, Germany, 13–16 March 2019; pp. 45:1–45:15.
4. Johnson, D.S. Approximation Algorithms for Combinatorial Problems. *J. Comput. Syst. Sci.* **1974**, *9*, 256–278. [CrossRef]
5. Stein, S.K. Two Combinatorial Covering Theorems. *J. Comb. Theory Ser. A* **1974**, *16*, 391–397. [CrossRef]
6. Lovász, L. On the Ratio of Optimal Integral and Fractional Covers. *Discret. Math.* **1975**, *13*, 383–390. [CrossRef]
7. Chvatal, V. A Greedy Heuristic for the Set-Covering Problem. *Math. Oper. Res.* **1979**, *4*, 233–235. [CrossRef]
8. Slavı́k, P. A Tight Analysis of the Greedy Algorithm for Set Cover. *J. Algorithms* **1997**, *25*, 237–254. [CrossRef]
9. Demaine, E.D.; Hajiaghayi, M. The Bidimensionality Theory and Its Algorithmic Applications 1. *Comput. J.* **2008**, *51*, 292–302. [CrossRef]
10. Lund, C.; Yannakakis, M.; Yannakakis, M. On the Hardness of Approximating Minimization Problems. *J. ACM* **1994**, *41*, 960–981. [CrossRef]
11. Raz, R.; Safra, S. A Sub-Constant Error-Probability Low-Degree Test, and a Sub-Constant Error-Probability PCP Characterization of NP. In Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing (STOC '97), El Paso, TX, USA, 4–6 May 1997; ACM: New York, NY, USA, 1997; pp. 475–484.
12. Feige, U. A Threshold of $\ln n$ for Approximating Set Cover. *J. ACM* **1998**, *45*, 634–652. [CrossRef]
13. Alon, N.; Moshkovitz, D.; Safra, S. Algorithmic Construction of Sets for k-restrictions. *ACM Trans. Algorithms* **2006**, *2*, 153–177. [CrossRef]

14. Moshkovitz, D. The Projection Games Conjecture and the NP-Hardness of ln n-Approximating Set-Cover. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*; Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B., et al., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7408, pp. 276–287.
15. Dinur, I.; Steurer, D. Analytical Approach to Parallel Repetition. In Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing (STOC '14), New York, NY, USA, 31 May–3 June 2014; ACM: New York, NY, USA, 2014; pp. 624–633.
16. Chalermsook, P.; Cygan, M.; Kortsarz, G.; Laekhanukit, B.; Manurangsi, P.; Nanongkai, D.; Trevisan, L. From Gap-ETH to FPT-Inapproximability: Clique, Dominating Set, and More. In Proceedings of the 2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS), Berkeley, CA, USA, 15–17 October 2017; pp. 743–754.
17. Laekhanukit, B.; Manurangsi, P. On the Parameterized Complexity of Approximating Dominating Set. In Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2018), Los Angeles, CA, USA, 25–29 June 2018; ACM Press: Los Angeles, CA, USA, 2018; pp. 1283–1296.
18. Lin, B. A Simple Gap-Producing Reduction for the Parameterized Set Cover Problem. In Proceedings of the 46th International Colloquium on Automata, Languages, and Programming (ICALP 2019), Patras, Greece, 8–12 July 2019; pp. 81:1–81:15.
19. Furst, M.; Saxe, J.B.; Sipser, M. Parity, Circuits, and the Polynomial-Time Hierarchy. *Math. Syst. Theory* **1984**, *17*, 13–27. [CrossRef]
20. Ajtai, M. Σ_1^1-Formulae on finite structures. *Ann. Pure Appl. Log.* **1983**, *24*, 1–48. [CrossRef]
21. Håstad, J. Almost Optimal Lower Bounds for Small Depth Circuits. In *Leibniz International Proceedings in Informatics (LIPIcs), Proceedings of the 41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016)*; Faliszewski, P., Muscholl, A., Niedermeier, R., Eds.; Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik: Dagstuhl, Germany, 2016; Volume 58, pp. 1–14.
22. Williams, R. New Algorithms and Lower Bounds for Circuits with Linear Threshold Gates. In Proceedings of the Forty-Sixth Annual ACM Symposium on Theory of Computing (STOC '14), New York, NY, USA, 31 May–3 June 2014; ACM: New York, NY, USA, 2014; pp. 194–202.
23. Williams, R. Nonuniform ACC Circuit Lower Bounds. *J. ACM* **2014**, *61*, 2:1–2:32. [CrossRef]
24. Murray, C.; Williams, R. Circuit Lower Bounds for Nondeterministic Quasi-Polytime: An Easy Witness Lemma for NP and NQP. In Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing (STOC '18), Los Angeles, CA, USA, 25–29 June 2018; ACM: New York, NY, USA, 2018; pp. 890–901.
25. Rossman, B. On the Constant-depth Complexity of k-clique. In Proceedings of the Fortieth Annual ACM Symposium on Theory of Computing (STOC '08), Victoria, BC, Canada, 17–20 May 2008; ACM: New York, NY, USA, 2008; pp. 721–730.
26. Chen, Y.; Flum, J. Some Lower Bounds in Parameterized AC^0. In *Leibniz International Proceedings in Informatics (LIPIcs), Proceedings of the 41st International Symposium on Mathematical Foundations of Computer Science (MFCS 2016)*; Faliszewski, P., Muscholl, A., Niedermeier, R., Eds.; Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik: Dagstuhl, Germany, 2016; Volume 58, pp. 1–14.
27. Elberfeld, M.; Stockhusen, C.; Tantau, T. On the Space and Circuit Complexity of Parameterized Problems: Classes and Completeness. *Algorithmica* **2015**, *71*, 661–701. [CrossRef]
28. Downey, R.G.; Fellows, M.R. *Parameterized Complexity*; Monographs in Computer Science; Springer: New York, NY, USA, 1999.
29. Chlebík, M.; Chlebíková, J. Approximation Hardness of Dominating Set Problems in Bounded Degree Graphs. *Inf. Comput.* **2008**, *206*, 1264–1275. [CrossRef]
30. Lai, W. The Inapproximability of k-DominatingSet for Parameterized AC^0 Circuits. In Proceedings of the Thirteenth International Frontiers of Algorithmics Workshop (FAW 2019), Sanya, China, 29 April–3 May 2019; pp. 133–143.
31. Sarkar, K.; Colbourn, C.J. Upper Bounds on the Size of Covering Arrays. *SIAM J. Discret. Math.* **2017**, *31*, 1277–1293. [CrossRef]

32. Kleitman, D.J.; Spencer, J. Families of k-independent sets. *Discret. Math.* **1973**, *6*, 255–262. [CrossRef]
33. Pătraşcu, M.; Williams, R. On the Possibility of Faster SAT Algorithms. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms*; Charikar, M., Ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2010; pp. 1065–1075.