Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 186 (2026)

Almost Optimal FPT Inapproximability for k-SetCover*

Reported by

Venkatesan Guruswami
venkatg@berkeley.edu
Departments of EECS & Mathematics
UC Berkeley

Xuandi Ren
xuandi_ren@berkeley.edu
Department of EECS
UC Berkeley

Abstract

We show that $\left(\frac{\log n}{\log\log n}\right)$-approximate parameterized k-SetCover is W[1]-hard, and has no $n^{o(k/\log k)}$-time algorithms under ETH. This improves upon the previous best factors $\left(\frac{\log n}{\log\log n}\right)^{1/k}$ in (Lin, 2019) and $(\log n)^{1/\operatorname{poly}(k)}$ in (Karthik, Laekhanukit, and Manurangsi, 2019). Here k is the yes-case guarantee and n is the number of candidate sets. While the best approximation ratio is still $O(\log n)$ via the greedy algorithm, closing this $1/k$ gap in the exponent has been a longstanding open problem; we remove this loss via a simple direct reduction.

The construction is self-contained and does not rely on the parameterized inapproximability hypothesis (PIH). Starting with sparse parameterized 2-CSP instances (Karthik, Marx, Pilipczuk, and Souza, 2024), we build a monotone CNF formula, which is equivalent to a Set-Cover instance. To obtain a k-versus-h gap, the reduction enumerates all hash functions from Σ to $[2h]$ and all unsatisfiable 2-CSP instances on the same constraint graph with alphabet $[2h]$. For each such instance, it asks for a certificate that the hashed label pairs are not all contained in that instance. Perfect hashing makes this enumeration efficient for $h = \log n/\log\log n$.

AI Disclosure. The proof was discovered by GPT-6 Astra in response to the authors' prompts about the parameterized Minimum Monotone Satisfying Assignment (MMSA) problem. The authors checked the proofs and wrote the paper; they take responsibility for the correctness of the results but claim no intellectual credit for them.

*Work supported in part by NSF grant CCF-2211972 and DOD Advanced Research Projects Agency grant HR0011262E031.

ISSN 1433-8092

1 Introduction

SetCover is a fundamental problem in combinatorial optimization. It is one of Karp's first 21 NP-complete problems [Kar72]. Given a collection of sets over a universe U, the goal is to find a smallest subcollection whose union is U. The simple formulation and the tight connection to other combinatorial problems have made SetCover a central example for studying hardness of approximation. The polynomial-time approximability of SetCover is by now well understood. The greedy algorithm achieves a $(1+\ln|U|)$-approximation [Joh74, Chv79]. On the hardness side, Feige [Fei98] ruled out $(1-\varepsilon)\ln|U|$-approximation in polynomial time for every fixed $\varepsilon \in (0,1)$, assuming NP is not in quasi-polynomial time. Later works established NP-hardness for the same approximation ratio [Mos15, DS14]. Thus the logarithmic approximation guarantee is essentially optimal for polynomial-time algorithms, unless $\mathsf{P} = \mathsf{NP}$.

Parameterized complexity offers a complementary approach to NP-hard problems by identifying a parameter k and seeking algorithms whose running time depends arbitrarily on k but only polynomially on the input size n. An algorithm is said to be fixed-parameter tractable (FPT) if it runs in time $f(k)n^{O(1)}$, where f can be any computable function. Parameterized by the desired number of sets, k-SetCover is a canonical W[2]-complete problem [DF99], so it has no FPT algorithm unless $\mathsf{W[2]} = \mathsf{FPT}$. It is then natural to ask how good an approximation algorithm can be when FPT running time is allowed.

A long line of work has pursued this direction. Chen and Lin [CL19] ruled out any constant-factor FPT approximation for k-SetCover under $\mathsf{W[1]} \neq \mathsf{FPT}$, and a $(\log k)^{1/4-\varepsilon}$-factor FPT approximation for every fixed $\varepsilon \in (0, 1/4)$ under ETH. Chalermsook, Cygan, Kortsarz, Laekhanukit, Manurangsi, Nanongkai, and Trevisan [CCK+20] then established *total FPT inapproximability* for k-SetCover by ruling out any $g(k)$-factor approximation even in $f(k)n^{o(k)}$ time under Gap-ETH. Karthik, Laekhanukit, and Manurangsi [KLM19] and Lin [Lin19] used two different approaches to weaken the Gap-ETH assumption to gap-free hypotheses, giving $(\log n)^{o(1)}$-factor[1] FPT inapproximability under $\mathsf{W[1]} \neq \mathsf{FPT}$, and $(\log n)^{1/\operatorname{poly}(k)}$ and $(\log n/\log\log n)^{1/k}$ factors, respectively, in $f(k)n^{o(k)}$ time under ETH. Lin, Ren, Sun, and Wang [LRSW23] later showed the W[2]-hardness of approximating k-SetCover within any constant factor. While these hardness results seem satisfactory, we remark that the best FPT approximation algorithm is still the classical greedy one, which is an $O(\log n)$-approximation. There is still a gap in the exponent between the algorithm and the hardness results: 1 versus $o(1)$ under $\mathsf{W[1]} \neq \mathsf{FPT}$ and 1 versus $1/k$ under ETH. Closing this gap has been listed as an important open problem in the literature [Lin19, Section 4; FKLM20, Open Question 4; LRSW23, Question 2].

Open Question 1 ([FKLM20, Open Question 4])**.** Is there a $(\log n)^{1-o(1)}$ factor approximation algorithm for k-SetCover running in time $n^{k-0.1}$?

Our results. In this paper, we remove the aforementioned gap in the exponent and establish FPT inapproximability for k-SetCover with a factor of $\frac{\log n}{\log\log n}$. Our reduction gives W[1]-hardness and, under ETH, excludes $f(k)n^{o(k/\log k)}$-time algorithms for every computable function f. See Theorems 1.1 and 1.2 below and Table 1 for a comparison between our results and the previous state of the art.

[1] The factor stated in [KLM19, Theorem 1.3] is $(\log n)^{1/\operatorname{poly}(k)}$. When only FPT running time is required, padding strengthens this to $(\log n)^{\alpha(k)}$ for every computable positive function $\alpha(k) = o(1)$; see [GLRZ26, Lemma 2.8].

Theorem 1.1. *Assuming* $\mathsf{W[1]} \neq \mathsf{FPT}$*, there is no FPT algorithm that approximates* k-SetCover *within a* $\frac{\log n}{\log \log n}$ *factor.*

Theorem 1.2. *Assuming* ETH*, for any computable function* f*, there is no* $f(k)n^{o(k/\log k)}$*-time algorithm that approximates* k-SetCover *within a* $\frac{\log n}{\log \log n}$ *factor.*

We remark that the near-logarithmic FPT inapproximability for k-SetCover was obtained independently and almost simultaneously by Lin and Zheng [LZ26], and Karthik [K26], using potentially different reductions. They also obtain tight running-time lower bounds under ETH, SETH, and the k-SUM hypothesis.

Assumption	Inapproximability Factor	Runtime Lower Bound	Reference
$\mathsf{W[1]} \neq \mathsf{FPT}$	$(\log n)^{o(1)}$	$f(k)n^{O(1)}$	[Lin19, Theorem 4]
			[KLM19, Theorem 1.3]
	$\frac{\log n}{\log \log n}$		Theorem 1.1
ETH	$\left(\frac{\log n}{\log \log n}\right)^{1/k}$	$f(k)n^{o(k)}$	[Lin19, Theorem 2]
	$(\log n)^{1/\operatorname{poly}(k)}$		[KLM19, Theorem 1.4]
	$\frac{\log n}{\log \log n}$	$f(k)n^{o(k/\log k)}$	Theorem 1.2

Table 1: A summary of FPT inapproximability results for k-SetCover.

Connection to minimum monotone satisfying assignment. SetCover is equivalent to the depth-two Minimum Monotone Satisfying Assignment (MMSA) problem [ABMP01, DS04], in which we are given a monotone CNF formula: each Boolean variable corresponds to a candidate set and each clause corresponds to an element of the universe. The goal is to minimize the number of Boolean variables set to 1 while satisfying the formula. This correspondence preserves the optimum. Parameterized inapproximability of k-MMSA and connections between variants of different depths were studied in [Mar13, GLRZ26]. In the following, we use the above CNF view to describe our constructions for ease of presentation.

Proof overview. As a proof of concept, we present a direct reduction from k-Clique, which suffices for the W[1]-hardness. Let $G = (V, E)$ be a k-Clique instance. Put $m = \binom{k}{2}$, and fix an integer $h \geq m$. Introduce one Boolean variable x_{uv} for each edge $uv \in E$, so the total number of Boolean variables is $n = |E|$. An edge set $X \subseteq E$ specifies an assignment by setting $x_{uv} = 1$ whenever $uv \in X$. Take a perfect hash family Φ from V to $[2h]$: for every $A \subseteq V$ of size at most $2h$, some $\phi \in \Phi$ is injective on A. Let $\mathcal{H}$ be the collection of all *bad graphs*: simple graphs on $[2h]$ with at most h edges and no k-clique.

For each $\phi \in \Phi$ and $H \in \mathcal{H}$, form the clause

$$C_{\phi,H} = \bigvee_{\substack{uv \in E \\ \phi(u)=\phi(v)}} x_{uv} \vee \bigvee_{\substack{uv \in E \\ \phi(u)\neq\phi(v) \\ \{\phi(u),\phi(v)\}\notin E(H)}} x_{uv},$$

$$C = \bigwedge_{\substack{\phi \in \Phi \\ H \in \mathcal{H}}} C_{\phi,H}.$$

Each clause asks for a certificate: an edge in X whose endpoints either collide under ϕ or map to a nonedge of H.

For completeness, let X consist of the m edges of a k-clique in G. Fix any $\phi \in \Phi$ and $H \in \mathcal{H}$. If two clique vertices collide under ϕ, the edge between them satisfies the first disjunction. Otherwise, their images form a k-clique. Since H has no k-clique, some edge of X maps to a nonedge of H, satisfying the second disjunction. Thus every clause is satisfied, and $\mathrm{OPT}(C) \leq m$.

For soundness, suppose G has no k-clique and take any edge set $X \subseteq E$ with $|X| \leq h$. Let A be the set of endpoints of edges in X, so $|A| \leq 2h$. Choose $\phi \in \Phi$ injective on A, and let H be the graph on $[2h]$ with edge set

$$E(H) = \{\{\phi(u), \phi(v)\} : uv \in X\}.$$

This is a simple graph with at most h edges. By injectivity, any k-clique in H would lift to one in G, so $H \in \mathcal{H}$. No edge in X has colliding endpoints, and every edge in X maps into $E(H)$. Both disjunctions in $C_{\phi,H}$ are therefore false. Thus $\mathrm{OPT}(C) > h$, giving an (m, h)-gap SetCover instance.

There are at most $(h+1)\binom{2h}{2}^h = 2^{O(h \log h)}$ graphs to consider. Each can be tested for a k-clique in $h^{O(k)}$ time by enumerating k-subsets of $[2h]$. Together with the fact that the perfect hash family can be constructed in time $2^{O(h)}|V|\log(2|V|)$, this gives total construction time $(|V|+n)^{O(1)}2^{O(h \log h)}$. Taking $h = \lfloor \log n / \log\log n \rfloor$ makes this polynomial. This proves Theorem 1.1 and rules out $f(k)n^{o(\sqrt{k})}$-time algorithms under ETH. To avoid the quadratic parameter blow-up, we instead reduce from sparse parameterized 2-CSP, as described in Section 3.

2 Preliminaries

Definition 2.1 (SetCover)**.** A SetCover instance is a bipartite graph $I = (\mathcal{S}, U, E)$ with a family $\mathcal{S}$ of candidate sets, a universe U, and incidences $E \subseteq \mathcal{S} \times U$. A subfamily $X \subseteq \mathcal{S}$ is a *cover* if every element of U has a neighbor in X. Define

$$\mathrm{OPT}(I) = \min\{\, |X| : X \subseteq \mathcal{S} \text{ covers } U \,\},$$

with $\mathrm{OPT}(I) = +\infty$ if no cover exists. The k-SetCover problem asks whether U can be covered by at most k sets from $\mathcal{S}$. For $h \geq k$, the (k, h)-gap SetCover problem asks us to distinguish between the case $\mathrm{OPT}(I) \leq k$ and the case $\mathrm{OPT}(I) > h$.

Definition 2.2 (Minimum monotone satisfying assignment at depth two)**.** An MMSA_2 instance is a monotone CNF C on n Boolean variables: an AND of clauses, each of which is an OR of unnegated Boolean variables. For a set X of Boolean variables, write $C(X)$ for the value of C when precisely the Boolean variables in X are set to 1. Define

$$\mathrm{OPT}(C) = \min\{\, |X| : C(X) = 1 \,\},$$

with $\mathrm{OPT}(C) = +\infty$ if no satisfying assignment exists. The k-MMSA_2 problem asks whether C has a satisfying assignment with at most k Boolean variables set to 1.

SetCover is equivalent to MMSA_2, as pointed out in [GM99]: each Boolean variable corresponds to a candidate set, and each clause corresponds to an element of the universe. This correspondence preserves the optimum. Thus, in the rest of this paper, we use CNF notation to describe instances of k-SetCover.

Definition 2.3 (k-Clique)**.** An instance of k-Clique is a simple undirected graph $G = (V, E)$. The goal is to decide whether there exist k distinct vertices $v_1, \ldots, v_k \in V$ such that $v_i v_j \in E$ for every $i < j$.

Definition 2.4 (Parameterized 2-CSP [GLRSW25])**.** An instance of parameterized 2-CSP consists of an undirected constraint graph $G = (V, E)$, a finite alphabet Σ, and an allowed relation $R_i \subseteq \Sigma^2$ for each constraint $i \in [m]$, where $m = |E|$. Fix an ordering (u_i, v_i) of the endpoints of each constraint i. An assignment $a : V \to \Sigma$ satisfies the instance if

$$(a(u_i), a(v_i)) \in R_i \qquad \text{for every } i \in [m].$$

We write $k = |V|$ for the number of variables and $n = |\Sigma|$ for the alphabet size. We further assume that the constraint graph is simple and connected.

Lemma 2.5. *Parameterized 2-*CSP *is* W[1]*-hard. Furthermore, it has no $f(k)n^{o(k/\log k)}$-time algorithm for any computable function f under* ETH*, even when the constraint graph is 3-regular.*

Proof. The W[1]-hardness follows from a reduction from k-Clique. Given a k-Clique instance G with $k \geq 2$, introduce k variables, each choosing a vertex from a copy of $V(G)$. Let the alphabet be $\Sigma = V(G)$ and the constraint graph be K_k. For each $i < j$, let

$$R_{ij} = \{(u, v) \in \Sigma^2 : uv \in E(G)\}.$$

It's easy to see that a satisfying assignment exists if and only if G contains a k-clique.

The ETH lower bound follows from [KMPS24, Theorem 1.2], whose construction uses 3-regular expanders, which are connected, as constraint graphs. □

Lemma 2.6 (Perfect hash families [AYZ95])**.** *For any nonempty finite set V and $\Sigma = [h]$, there is a family $\Phi(V, \Sigma)$ of maps $\phi : V \to \Sigma$ such that*

$$\forall A \subseteq V,\ |A| \leq h, \qquad \exists \phi \in \Phi(V, \Sigma) \text{ injective on } A.$$

This family has size $2^{O(h)} \log(2|V|)$ and can be constructed deterministically in $2^{O(h)}|V| \log(2|V|)$ time.

3 A Reduction from Parameterized 2-CSP to k-SetCover

Let Γ be an instance of parameterized 2-CSP with constraint graph $G = (V, E)$, alphabet Σ, and relations $R_1, \ldots, R_m$.

The construction. Introduce one Boolean variable $x_{i,a,b}$ for each $(a,b) \in R_i$, giving a total of

$$N = \sum_{i=1}^{m} |R_i|$$

Boolean variables. Setting $x_{i,a,b} = 1$ means selecting the label pair (a,b) for the i-th constraint.

Fix an integer $h \geq m$. Take the perfect hash family $\Phi = \Phi(\Sigma, [2h])$ as constructed in Lemma 2.6. Define a *template* to be a family of relations $H = (H_i)_{i\in[m]}$, where $H_i \subseteq [2h]^2$ and $\sum_{i=1}^{m} |H_i| \leq h$. It specifies a 2-CSP on the same constraint graph G, with alphabet $[2h]$ and relation H_i on (u_i, v_i). The template is *bad* if this 2-CSP is unsatisfiable, i.e., if there is no assignment $z : V(G) \to [2h]$ such that

$$(z(u_i), z(v_i)) \in H_i \qquad \text{for every } i \in [m].$$

For each $\phi \in \Phi$ and each bad template H, form the clause

$$C_{\phi,H} = \bigvee_{\substack{i\in[m] \\ (a,b)\in R_i \\ (\phi(a),\phi(b))\notin H_i}} x_{i,a,b}, \tag{1}$$

which asks for a selected label pair (a,b) such that $(\phi(a), \phi(b)) \notin H_i$.

Let the final formula be

$$C = \bigwedge_{\substack{\phi\in\Phi \\ H \text{ bad}}} C_{\phi,H}.$$

In words, for every hash function $\phi \in \Phi$ and every bad template H, we ask for a certificate that the hashed label pairs are *not* contained in H.

Completeness. Let $\alpha : V(G) \to \Sigma$ be a satisfying assignment of Γ, and let

$$X_\alpha = \{x_{i,\alpha(u_i),\alpha(v_i)} : i \in [m]\}.$$

Set the m Boolean variables in X_α to 1 and all others to 0. Fix any $\phi \in \Phi$ and any bad template H. Since H is unsatisfiable, the assignment $z = \phi \circ \alpha$ violates some relation H_i. Hence $(\phi(\alpha(u_i)), \phi(\alpha(v_i))) \notin H_i$, so the variable $x_{i,\alpha(u_i),\alpha(v_i)}$, which is set to 1, satisfies $C_{\phi,H}$. This holds for every ϕ, H, so $C(X_\alpha) = 1$ and $\mathrm{OPT}(C) \leq m$.

Soundness. Suppose Γ is not satisfiable. Let X be a set of size at most h Boolean variables and let

$$A = \{a, b : x_{i,a,b} \in X\} \subseteq \Sigma$$

be the collection of selected labels in X, we have $|A| \leq 2h$. Choose $\phi \in \Phi$ injective on A. Define the template $H = (H_i)_{i\in[m]}$ where

$$H_i = \{(\phi(a), \phi(b)) : x_{i,a,b} \in X\} \qquad \text{for each } i \in [m].$$

Then $\sum_i |H_i| \leq |X| \leq h$. If H were satisfiable, let $z : V \to [2h]$ be a satisfying assignment. Since every vertex is incident to a constraint and z satisfies H, we have $z(v) \in \phi(A)$ for every $v \in V$. Define $\alpha : V \to \Sigma$ by setting $\alpha(v) = (\phi|_A)^{-1}(z(v))$ for every $v \in V$.

Injectivity guarantees that α is well-defined, and that for every $i \in [m]$, $x_{i,\alpha(u_i),\alpha(v_i)} \in X$. Thus α satisfies Γ, a contradiction. Therefore, H is bad and $C_{\phi,H}(X) = 0$ by our definition of H. This holds for every X of size at most h, so $\mathrm{OPT}(C) > h$.

Construction time. The number of Boolean variables is at most

$$N = \sum_{i=1}^{m} |R_i| \leq m|\Sigma|^2.$$

Let $M = m(2h)^2$, the number of templates is at most

$$\sum_{j=0}^{h} \binom{M}{j} \leq (h+1)M^h = 2^{O(h \log h)}.$$

To check whether a template is bad, we enumerate one pair from each H_i and check whether these pairs agree on shared variables. Since $\prod_i |H_i| \leq 2^{\sum_i |H_i|} \leq 2^h$, this takes $2^{O(h)}$ time per template.

By Lemma 2.6, the perfect hash family Φ has size $2^{O(h)} \log(2|\Sigma|)$ and can be constructed in $2^{O(h)}|\Sigma| \log(2|\Sigma|)$ time. Therefore, the entire circuit has construction time and size at most

$$N^{O(1)} 2^{O(h \log h)}. \tag{2}$$

For $N \geq 16$, set

$$h = \left\lfloor \frac{\log N}{\log \log N} \right\rfloor. \tag{3}$$

Since $h \log h \leq \log N$, the construction is in polynomial-time.

Plugging the W[1]-hardness and the ETH lower bound for parameterized 2-CSP from Lemma 2.5 into the reduction, we obtain the desired hardness of k-SetCover.

Proof of Theorem 1.1. Apply the reduction to the instance obtained from k-Clique as in the proof of Lemma 2.5. Thus the (m, h)-gap SetCover problem is W[1]-hard, where $m = \binom{k}{2}$. □

Proof of Theorem 1.2. Apply the reduction to the 3-regular instances in Lemma 2.5. The target parameter $m = 3k/2$ due to the 3-regularity and the number of Boolean variables is $N \leq n^{O(1)}$. Thus, an $f(m)N^{o(m/\log m)}$-time algorithm for (m, h)-gap SetCover would solve the source parameterized 2-CSP instances in $g(k)n^{o(k/\log k)}$ time for some computable function g, contradicting Lemma 2.5. □

References

[ABMP01] Michael Alekhnovich, Samuel R. Buss, Shlomo Moran, and Toniann Pitassi. Minimum propositional proof length is NP-hard to linearly approximate. *The Journal of Symbolic Logic*, 66(1):171–191, 2001. doi:10.2307/2694916.

[AYZ95] Noga Alon, Raphael Yuster, and Uri Zwick. Color-coding. *Journal of the ACM*, 42(4):844–856, 1995. doi:10.1145/210332.210337.

[CCK+20] Parinya Chalermsook, Marek Cygan, Guy Kortsarz, Bundit Laekhanukit, Pasin Manurangsi, Danupon Nanongkai, and Luca Trevisan. From gap-exponential time hypothesis to fixed parameter tractable inapproximability: Clique, dominating set, and more. *SIAM Journal on Computing*, 49(4):772–810, 2020. doi:10.1137/18M1166869.

[CHKX06] Jianer Chen, Xiuzhen Huang, Iyad A. Kanj, and Ge Xia. Strong computational lower bounds via parameterized complexity. *Journal of Computer and System Sciences*, 72(8):1346–1367, 2006. doi:10.1016/j.jcss.2006.04.007.

[Chv79] Vašek Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979. doi:10.1287/moor.4.3.233.

[CL19] Yijia Chen and Bingkai Lin. The constant inapproximability of the parameterized Dominating Set problem. *SIAM Journal on Computing*, 48(2):513–533, 2019. doi:10.1137/17M1127211.

[CMV23] Eden Chlamtáč, Yury Makarychev, and Ali Vakilian. Approximating Red-Blue Set Cover and Minimum Monotone Satisfying Assignment. In *Proceedings of APPROX/RANDOM*, volume 275 of LIPIcs, pages 11:1–11:19, 2023. doi:10.4230/LIPIcs.APPROX/RANDOM.2023.11.

[DF99] Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity*. Springer, 1999. doi:10.1007/978-1-4612-0515-9.

[DFMR08] Rodney G. Downey, Michael R. Fellows, Catherine McCartin, and Frances Rosamond. Parameterized approximation of dominating set problems. *Information Processing Letters*, 109(1):68–70, 2008. doi:10.1016/j.ipl.2008.09.017.

[DS04] Irit Dinur and Shmuel Safra. On the hardness of approximating label-cover. *Information Processing Letters*, 89(5):247–254, 2004. doi:10.1016/j.ipl.2003.11.007.

[DS14] Irit Dinur and David Steurer. Analytical approach to parallel repetition. In *Proceedings of STOC*, pages 624–633, 2014. doi:10.1145/2591796.2591884.

[Fei98] Uriel Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998. doi:10.1145/285055.285059.

[FKLM20] Andreas Emil Feldmann, Karthik C. S., Euiwoong Lee, and Pasin Manurangsi. A survey on approximation in parameterized complexity: Hardness and algorithms. *Algorithms*, 13(6), Article 146, 2020. doi:10.3390/a13060146.

[GLRSW25] Venkatesan Guruswami, Bingkai Lin, Xuandi Ren, Yican Sun, and Kewen Wu. Almost optimal time lower bound for approximating parameterized Clique, CSP, and more, under ETH. In *Proceedings of STOC*, pages 2136–2144, 2025. doi:10.1145/3717823.3718130. Full version: arXiv:2404.08870v2.

[GLRZ26] Venkatesan Guruswami, Bingkai Lin, Xuandi Ren, and Xin Zheng. On the Approximability of Parameterized Minimum Monotone Satisfying Assignment. In *Proceedings of APPROX/RANDOM*, volume 392 of LIPIcs, pages 20:1–20:14, 2026. doi:10.4230/LIPIcs.APPROX/RANDOM.2026.20.

[GM99] Michael H. Goldwasser and Rajeev Motwani. Complexity measures for assembly sequences. *International Journal of Computational Geometry & Applications*, 9(4–5):371–417, 1999. doi:10.1142/S0218195999000248.

[Gro07] Martin Grohe. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *Journal of the ACM*, 54(1), Article 1, 2007. doi:10.1145/1206035.1206036.

[IP01] Russell Impagliazzo and Ramamohan Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367–375, 2001. doi:10.1006/jcss.2000.1727.

[IPZ01] Russell Impagliazzo, Ramamohan Paturi, and Francis Zane. Which problems have strongly exponential complexity? *Journal of Computer and System Sciences*, 63(4):512–530, 2001. doi:10.1006/jcss.2001.1774.

[Joh74] David S. Johnson. Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9(3):256–278, 1974. doi:10.1016/S0022-0000(74)80044-9.

[Kar72] Richard M. Karp. Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972. doi:10.1007/978-1-4684-2001-2_9.

[K26] Karthik C. S. Almost-Logarithmic Inapproximability of Parameterized Set Cover. Private communication, 2026.

[KLM19] Karthik C. S., Bundit Laekhanukit, and Pasin Manurangsi. On the parameterized complexity of approximating dominating set. *Journal of the ACM*, 66(5), Article 33, 2019. doi:10.1145/3325116. Full version: arXiv:1711.11029v2.

[KMPS24] Karthik C. S., Dániel Marx, Marcin Pilipczuk, and Uéverton Souza. Conditional lower bounds for sparse parameterized 2-CSP: A streamlined proof. In *Proceedings of SOSA*, pages 383–395, 2024. doi:10.1137/1.9781611977936.35.

[KN21] Karthik C. S. and Inbal Livni-Navon. On hardness of approximation of parameterized Set Cover and Label Cover: Threshold graphs from error correcting codes. In *Proceedings of SOSA*, pages 210–223, 2021. doi:10.1137/1.9781611976496.24.

[Lin19] Bingkai Lin. A simple gap-producing reduction for the parameterized Set Cover problem. In *Proceedings of ICALP*, volume 132 of LIPIcs, pages 81:1–81:15, 2019. doi:10.4230/LIPIcs.ICALP.2019.81.

[LRSW23] Bingkai Lin, Xuandi Ren, Yican Sun, and Xiuhan Wang. Constant approximating parameterized k-SetCover is W[2]-hard. In *Proceedings of SODA*, pages 3305–3316, 2023. doi:10.1137/1.9781611977554.ch126.

[LZ26] Bingkai Lin and Xin Zheng. Near-Logarithmic Inapproximability of Parameterized Set Cover. Private communication, 2026.

[Mar08] Dániel Marx. Parameterized complexity and approximation algorithms. *The Computer Journal*, 51(1):60–78, 2008. doi:10.1093/comjnl/bxm048.

[Mar13] Dániel Marx. Completely inapproximable monotone and antimonotone parameterized problems. *Journal of Computer and System Sciences*, 79(1):144–151, 2013. doi:10.1016/j.jcss.2012.09.001.

[Mos15] Dana Moshkovitz. The Projection Games Conjecture and the NP-Hardness of $\ln n$-Approximating Set-Cover. *Theory of Computing*, 11(7):221–235, 2015. doi:10.4086/toc.2015.v011a007.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il