Faster Algorithms for k-Orthogonal Vectors in Low Dimension

Anita Dürr
ETH Zürich, Switzerland

Evangelos Kipouridis
Max Planck Institute for Informatics, Saarbrücken, Germany

Michael Lampis
Université Paris Dauphine, Paris, France

Karol Węgrzycki
Max Planck Institute for Informatics, Saarbrücken, Germany

Abstract

In the Orthogonal Vectors problem (OV), we are given two families A, B of subsets of $\{1, \ldots, d\}$, each of size n, and the task is to decide whether there exists a pair $\mathbf{a} \in A$ and $\mathbf{b} \in B$ such that $\mathbf{a} \cap \mathbf{b} = \emptyset$. Straightforward algorithms for this problem run in $\mathcal{O}(n^2 \cdot d)$ or $\mathcal{O}(2^d \cdot n)$ time, and assuming SETH, there is no $2^{o(d)} \cdot n^{2-\varepsilon}$ time algorithm that solves this problem for any constant $\varepsilon > 0$.

Williams (FOCS 2024) presented a $\tilde{\mathcal{O}}(1.35^d \cdot n)$-time algorithm for the problem, based on the succinct equality-rank decomposition of the disjointness matrix. In this paper, we present a combinatorial algorithm that runs in randomized time $\tilde{\mathcal{O}}(1.25^d \cdot n)$. This can be improved to $\mathcal{O}(1.16^d \cdot n)$ using computer-aided evaluations.

We also consider a more general k-Orthogonal Vectors problem, where given k families $A_1, \ldots, A_k$ of subsets of $\{1, \ldots, d\}$, each of size n, the task is to find elements $\mathbf{a}_i \in A_i$ for every $i \in \{1, \ldots, k\}$ such that $\mathbf{a}_1 \cap \mathbf{a}_2 \cap \ldots \cap \mathbf{a}_k = \emptyset$. We show that for every fixed $k \geqslant 2$, there exists $\varepsilon_k > 0$ such that the k-OV problem can be solved in time $\mathcal{O}(2^{(1-\varepsilon_k)\cdot d} \cdot n)$. We also show that, asymptotically, this is the best we can hope for: for any $\varepsilon > 0$ there exists a $k \geqslant 2$ such that $2^{(1-\varepsilon)\cdot d} \cdot n^{\mathcal{O}(1)}$ time algorithm for k-Orthogonal Vectors would contradict the Set Cover Conjecture.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms; Theory of computation → Computational complexity and cryptography

Keywords and phrases Orthogonal Vectors, Fine-grained Complexity, Exact Algorithms, Set Cover

Digital Object Identifier 10.4230/LIPIcs.ICALP.2026.85

Category Track A: Algorithms, Complexity and Games

Related Version *Extended Version*: `https://arxiv.org/abs/2507.11098`

Funding *Anita Dürr*: Part of this work was done while affiliated to Saarland University and Max Planck Institute for Informatics, Saarbrücken, Germany, where this work was part of the project TIPEA that has received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (grant agreement No. 850979).
Karol Węgrzycki: Supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) grant number 559177164.

1 Introduction

In the Orthogonal Vectors problem, the task is to find a disjoint pair of vectors in a given collection of vectors. We view vectors in $\{0, 1\}^d$ as subsets of $\{1, 2, \ldots, d\}$ for some dimension d.

▶ **Definition 1.1** (Orthogonal Vectors (2-OV)). *Given two families A, B of subsets of $\{1, \ldots, d\}$ with $|A| = |B| = n$, the Orthogonal Vectors problem asks whether there exist $\mathbf{a} \in A$ and $\mathbf{b} \in B$ such that $\mathbf{a} \cap \mathbf{b} = \emptyset$.*

The naive algorithm for the Orthogonal Vectors problem runs in $\mathcal{O}(n^2 \cdot d)$ time. This quadratic-time algorithm has been slightly improved to $n^{2-1/\mathcal{O}(\log(d/\log n))}$[6, 1], but no $\mathcal{O}(n^{2-\varepsilon})$-time algorithm is currently known for any $\varepsilon > 0$. In fact, the conjecture that there is no such $\varepsilon > 0$ for which Orthogonal Vectors can be solved in $\mathcal{O}(n^{2-\varepsilon})$ time for $d = \omega(\log n)$ is one of the central hypotheses in the field of fine-grained complexity [19]. The Orthogonal Vectors problem and its connections to other problems have been thoroughly investigated [7, 12], and a $2^{o(d)} \cdot n^{2-\varepsilon}$-time algorithm for any $\varepsilon > 0$ would contradict the Strong Exponential Time Hypothesis [17].

In this paper we focus on the low-dimensional regime when $d = c \log n$ for some small constant $c > 0$. In that regime, the SETH lower bound does not preclude the possibility of subquadratic algorithms. In fact, folklore algorithms with running times $\mathcal{O}(d \cdot 2^d \cdot n)$ or even $\widetilde{\mathcal{O}}(n + 2^d)$ are known [4][1].

Inspired by connections to non-uniform circuit lower bounds, Williams [18] investigated whether the dependence on d can be improved, and designed an $\widetilde{\mathcal{O}}(1.35^d \cdot n)$-time algorithm for Orthogonal Vectors. Williams [18] exploited the structure of the problem by presenting a systematic approach based on constant-sized decompositions of the disjointness matrix.

Our first contribution is a new combinatorial algorithm for the problem:

▶ **Theorem 1.1.** *The Orthogonal Vectors problem can be solved in $\widetilde{\mathcal{O}}(1.25^d \cdot n)$ time by a randomized, one-sided error algorithm that succeeds with probability $1 - 2^{-d^{\Omega(1)}}$. Using computer-aided evaluation, the running time can be improved to $\mathcal{O}(1.16^d \cdot n)$.*

This result was independently discovered by Alman and Li [2]. It is worth noting that the technique of Williams [18] also allows one to deterministically count the number of solutions to the Orthogonal Vectors problem in $\widetilde{\mathcal{O}}(1.38^d \cdot n)$ time, whereas ours cannot count and is inherently randomized. Our approach is based on an application of the representation method [11]. In a nutshell, we observe that if $\mathbf{a} \cap \mathbf{b} = \emptyset$, then there exists a certificate set $\mathbf{c}$ such that: (i) $\mathbf{a} \subseteq \mathbf{c}$, and (ii) $\mathbf{b} \cap \mathbf{c} = \emptyset$. Our algorithm samples an appropriate number of such sets and constructs a data structure that certifies conditions (i) and (ii).

We remark that, using computer-aided evaluation, the running time of our algorithm can be improved to $\mathcal{O}(1.16^d \cdot n)$. This constant is very close to the barrier achievable with current techniques when the size of $\mathbf{a}$ and $\mathbf{b}$ is $d/4$: in that setting, a similar approach was used to design a $\mathcal{O}(1.14^d \cdot n)$ time algorithm for Orthogonal Vectors. Any improvement on that running time would improve known algorithms for Subset Sum [13].

Next, we consider a more general k-Orthogonal Vectors problem:

▶ **Definition 1.2** (k-Orthogonal Vectors (k-OV)). *Given families $A_1, \ldots, A_k$ of subsets of $\{1, \ldots, d\}$, each of size n, the k-Orthogonal Vectors problem asks whether there exist $\mathbf{a}_1 \in A_1, \ldots, \mathbf{a}_k \in A_k$ such that $\mathbf{a}_1 \cap \mathbf{a}_2 \cap \ldots \cap \mathbf{a}_k = \emptyset$.*

Naive algorithms for the k-Orthogonal Vectors problem run in $\mathcal{O}(d \cdot k \cdot n^k)$ time, and any $2^{o(d)} \cdot n^{k-\varepsilon}$ time algorithm for $\varepsilon > 0$ would contradict SETH. As in the 2-OV case, for every fixed $k \geqslant 2$, a folklore algorithm solves the k-Orthogonal Vectors problem in $\mathcal{O}(d \cdot 2^d \cdot n)$ time. We show that the exponential dependence on d can be improved:

[1] $\widetilde{\mathcal{O}}(\cdot)$ hides polylogarithmic factors, while $\mathcal{O}^\star(\cdot)$ hides polynomial factors in the input. Since $d = c \log n$ for some constant $c > 0$ throughout this paper, $\widetilde{\mathcal{O}}$ also hides polynomial factors in d.

▶ **Theorem 1.2.** *For every $k \geqslant 2$, there exists $\varepsilon_k > 0$ such that the k-Orthogonal Vectors problem can be solved deterministically in time $\mathcal{O}(2^{(1-\varepsilon_k)d} \cdot n)$.*

To prove the theorem, we observe that the algorithm of Björklund et al. [4] can be generalized to give a $\mathcal{O}\big(d \cdot (|{\downarrow}A_1| + \ldots + |{\downarrow}A_k|)\big)$-time algorithm for the k-Orthogonal Vectors problem, where ${\downarrow}F := \{\mathbf{x} \subseteq \mathbf{s} \mid \mathbf{s} \in F\}$ denotes the down-closure of the family F. First, guess the cardinalities of the solution $1 \leqslant \alpha_1 d \leqslant \ldots \leqslant \alpha_k d \leqslant d$ and assume that the family A_i contains only sets of cardinality $\alpha_i d$. If $\alpha_k < (1-\varepsilon)$ for some $\varepsilon > 0$, then $|{\downarrow}A_i| \leqslant 2^{(1-\varepsilon)d}|A_i|$ for each $i \in \{1, \ldots, k\}$ and thus the above algorithm runs in time $\mathcal{O}(2^{(1-\varepsilon)d} \cdot n)$. Therefore, consider the case where $\alpha_k > (1-\varepsilon)$. Then $|A_k| \leqslant \binom{d}{\varepsilon d}$. Hence, we can afford to guess a set $\mathbf{a}_k \in A_k$ and recurse on the $(k-1)$-Orthogonal Vectors problem with universe $\mathbf{a}_k$. We continue this process until we reach the case $k = 2$, at which point we can use a deterministic algorithm for $k = 2$.

Finally, we observe that Theorem 1.2 is asymptotically optimal: having a $2^{(1-\varepsilon)d} \cdot n^{\mathcal{O}(1)}$-time algorithm that solves k-Orthogonal Vectors for every $k \geqslant 2$ would drastically improve the currently best algorithms for Set Cover.

▶ **Theorem 1.3.** *For every $c > 0$ and $\varepsilon > 0$, there exists $k \in \mathbb{N}$ such that the k-Orthogonal Vectors problem is not solvable in $\mathcal{O}(2^{(1-\varepsilon)d} \cdot n^c)$ time, assuming the Set Cover Conjecture.*

Recently, it has been shown that the *Set Cover Conjecture* and the *Asymptotic Rank Conjecture* cannot both be true [5, 14]. We show that these techniques can also be used to conditionally improve algorithms for k-Orthogonal Vectors:

▶ **Theorem 1.4.** *Assuming the Asymptotic Rank Conjecture, there exist $c > 0$ and $\varepsilon > 0$, such that any k-Orthogonal Vectors instance can be solved in $\mathcal{O}(2^{(1-\varepsilon)\cdot d} \cdot n^c)$ time for every $k \in \mathbb{N}$.*

As these results are relatively straightforward corollaries of the known connections, we include them in Section 5.

Related Work. As mentioned before, techniques used in this paper are inspired by [13] which in turn is based on the more general framework of Fomin et al. [11]. These techniques have also been used to give a faster algorithm for Subset Balancing Problems [15]. Similar techniques have also been used by Chukhin et al. [8] in the context of monotone circuits and matrix rigidity.

Organization. After introducing notations and discussing a folklore algorithm in Section 2, we give our algorithm for the Orthogonal Vectors problem in Section 3. In Section 4 we give our algorithm for the k-Orthogonal Vectors problem. Equivalence between k-Orthogonal Vectors and Set Cover is provided in Section 5.

2 Preliminaries

We use the shorthand notation $[n] := \{1, 2, \ldots, n\}$ for any $n \in \mathbb{N}$. For a universe set $\mathbf{u}$ and a set family $F \subseteq 2^{\mathbf{u}}$, we denote the down-closure of F as ${\downarrow}F := \{\mathbf{x} \subseteq \mathbf{s} \mid \mathbf{s} \in F\}$. We use $\mathbf{s} \subseteq_R F$ to denote that $\mathbf{s}$ is a subset of F chosen uniformly at random among all such subsets. For a non-negative integer $i \leqslant |\mathbf{u}|$, the set of subsets of $\mathbf{u}$ of size i is denoted $\binom{\mathbf{u}}{i}$. We use $\mathbf{x} \sqcup \mathbf{y}$ to denote a disjoint partition. We use the Iverson bracket notation for the characteristic function; i.e. for any logical expression b, the value of $[\![b]\!]$ is 1 if b is true and 0 otherwise.

Binomial Coefficients. The binary entropy function is defined as $h(\alpha) := -\alpha \log_2(\alpha) - (1-\alpha)\log_2(1-\alpha)$ for every $\alpha \in (0,1)$, and $h(\alpha) = 0$ for $\alpha \in \{0,1\}$. We use it to approximate binomial coefficients with the following inequalities [16]:

$$2^{h(\alpha)n} \geqslant \binom{n}{\alpha n} \geqslant \Omega\left(2^{h(\alpha)n} \cdot n^{-1/2}\right), \tag{1}$$

for every $\alpha \in (0,1)$. The binary entropy function generalizes to the multinomial form. For $\alpha_1, \ldots, \alpha_k \in (0,1)$ with $\sum_{i=1}^{k} \alpha_i = 1$ it is defined as:

$$h(\alpha_1, \ldots, \alpha_k) := -\sum_{i=1}^{k} \alpha_i \log_2 \alpha_i.$$

Note that for $\alpha \in (0,1)$, we use the shorthand notation $h(\alpha) := h(\alpha, 1-\alpha)$. The multinomial coefficient can be approximated with the function h as follows:

▶ **Lemma 2.1** ([9], Lemma 2.2)**.**

$$2^{h(\alpha_1,\ldots,\alpha_k)n} \cdot n^{-\mathcal{O}(k)} \leqslant \binom{n}{\alpha_1 n, \ldots, \alpha_k n} \leqslant 2^{h(\alpha_1,\ldots\alpha_k)n}.$$

Orthogonal Vectors. Throughout the paper, we denote by n the number of vectors given in each family of an OV instance and by d the dimension of the vectors. Vectors in $\{0,1\}^d$ are often interpreted as subsets of $[d]$, and we use these terms interchangeably. We always assume that $d = c\log n$ for some constant $c > 0$. For completeness, we include a simple deterministic algorithm for 2-OV. Notice that [18] offers a faster deterministic algorithm. However, the running time proven below is enough for the base case of the recursive algorithm in Theorem 1.2.

▶ **Lemma 2.2.** *The Orthogonal Vectors problem can be solved deterministically in $\widetilde{\mathcal{O}}(2^{d/2} \cdot n)$ time.*

Proof. Partition the universe $[d]$ into the low order bits $\mathbf{u}_L = \{1, \ldots, \lfloor d/2 \rfloor\}$ and high order bits $\mathbf{u}_H = \{\lfloor d/2 \rfloor + 1, \ldots, d\}$. For any subset $\mathbf{x} \subseteq [d]$, we write $\mathbf{x} = \mathbf{x}_L \uplus \mathbf{x}_H$ for $\mathbf{x}_L = \mathbf{x} \cap \mathbf{u}_L$ and $\mathbf{x}_H = \mathbf{x} \cap \mathbf{u}_H$. Consider the following sets of vectors:

$$S_A := \{\mathbf{a}_L \uplus \mathbf{x}_H \subseteq [d] \mid \mathbf{a}_L \uplus \mathbf{a}_H \in A,\, \mathbf{x}_H \subseteq \mathbf{u}_H,\, \mathbf{x}_H \cap \mathbf{a}_H = \emptyset\},$$
$$S_B := \{\mathbf{x}_L \uplus \mathbf{b}_H \subseteq [d] \mid \mathbf{b}_L \uplus \mathbf{b}_H \in B,\, \mathbf{x}_L \subseteq \mathbf{u}_L,\, \mathbf{x}_L \cap \mathbf{b}_L = \emptyset\}.$$

We claim that $S_A \cap S_B \neq \emptyset$ if and only if an orthogonal pair exists. Assume that there exist orthogonal vectors $\mathbf{a}_L \uplus \mathbf{a}_H \in A$ and $\mathbf{b}_L \uplus \mathbf{b}_H \in B$. Then $\mathbf{a}_L \uplus \mathbf{b}_H$ is contained in both S_A and S_B. For the converse, assume that $\mathbf{x}_L \uplus \mathbf{x}_H \in S_A \cap S_B$. Since $\mathbf{x}_L \uplus \mathbf{x}_H \in S_A$, there exists $\mathbf{a}_L \uplus \mathbf{a}_H \in A$ such that $\mathbf{x}_L = \mathbf{a}_L$ and $\mathbf{x}_H \cap \mathbf{a}_H = \emptyset$. Similarly, since $\mathbf{x}_L \uplus \mathbf{x}_H \in S_B$, there exists $\mathbf{b}_L \uplus \mathbf{b}_H \in B$ such that $\mathbf{b}_H = \mathbf{x}_H$ and $\mathbf{b}_L \cap \mathbf{x}_L = \emptyset$. Hence, $\mathbf{a}_L \cap \mathbf{b}_L = \emptyset$ and $\mathbf{a}_H \cap \mathbf{b}_H = \emptyset$.

Observe that both $|S_A|$ and $|S_B|$ are at most $2^{d/2} \cdot (|A| + |B|)$, so in $\widetilde{\mathcal{O}}(2^{d/2} \cdot n)$ time we can construct S_A and S_B, and check whether $S_A \cap S_B \neq \emptyset$. ◀

3 Algorithm for Orthogonal Vectors (2-OV)

▶ **Theorem 1.1.** *The Orthogonal Vectors problem can be solved in $\widetilde{\mathcal{O}}(1.25^d \cdot n)$ time by a randomized, one-sided error algorithm that succeeds with probability $1 - 2^{-d^{\Omega(1)}}$. Using computer-aided evaluation, the running time can be improved to $\mathcal{O}(1.16^d \cdot n)$.*

The algorithm of Theorem 1.1 first randomly partitions the universe $\mathbf{u} = [d]$ into ℓ sets $\mathbf{u}_1, \ldots, \mathbf{u}_\ell$ of equal size and samples random families C_i of subsets of $\mathbf{u}_i$. We will show that if there exists $\mathbf{a} \in A$ and $\mathbf{b} \in B$ such that $\mathbf{a} \cap \mathbf{b} = \emptyset$, then with sufficiently high probability the families $C_1, \ldots, C_\ell$ contain a witness of the existence of $\mathbf{a}$ and $\mathbf{b}$. Finally, we introduce auxiliary families $L^i \subseteq C_i$ and $R^i \subseteq C_i$ that allow us to efficiently find this witness, if it exists.

In the following, we assume that there exist $\mathbf{a} \in A$ and $\mathbf{b} \in B$ with $\mathbf{a} \cap \mathbf{b} = \emptyset$. Let $\alpha, \beta \in [0,1]$ be real numbers such that $|\mathbf{a}| = \alpha \cdot d$ and $|\mathbf{b}| = \beta \cdot d$. Note that we can guess the values of α and β in $\mathrm{poly}(d)$ time. Hence, from now on, we assume that we know the values of α and β precisely. Moreover, we can assume without loss of generality that $\alpha \leqslant \beta$ (by swapping A and B) and $\alpha + \beta \leqslant 1$ (as otherwise, the answer is trivially negative).

3.1 Partition of the universe

Let ℓ be a sufficiently large constant (for the sake of presentation we use $\ell = 100$). By padding, we can assume that d,αd and βd are multiples of ℓ. Consider a random partition of the universe $\mathbf{u} = [d]$ into ℓ parts $\mathbf{u}_1, \ldots, \mathbf{u}_\ell$ of equal cardinality such that $[d] = \mathbf{u}_1 \sqcup \ldots \sqcup \mathbf{u}_\ell$ and $|\mathbf{u}_i| = d/\ell$ for every $i \in [\ell]$. We observe that with sufficiently high probability, the solution is partitioned equally among $\mathbf{u}_i$.

▶ **Observation 3.1.** *Let $\mathbf{a} \in A$ and $\mathbf{b} \in B$ with $|\mathbf{a}| = \alpha d$ and $|\mathbf{b}| = \beta d$. With probability at least $d^{-\mathcal{O}(\ell)}$, for every $i \in [\ell]$ it holds that:*

$$|\mathbf{u}_i \cap \mathbf{a}| = \frac{\alpha d}{\ell} \quad \text{and} \quad |\mathbf{u}_i \cap \mathbf{b}| = \frac{\beta d}{\ell}.$$

Proof. We focus on bounding the probability that $|\mathbf{u}_i \cap \mathbf{a}| = \frac{\alpha d}{\ell}$ for every $i \in [\ell]$, as the reasoning for $\mathbf{b}$ is symmetric. For fixed integers $k_1, \ldots, k_\ell \geqslant 0$, the number of partitions of $\mathbf{u}$ into ℓ equal size sets such that $|\mathbf{u}_i \cap \mathbf{a}| = k_i$ for every $i \in [\ell]$ is

$$\binom{|\mathbf{a}|}{k_1, \ldots, k_\ell} \cdot \binom{|\mathbf{u} \setminus \mathbf{a}|}{d/\ell - k_1, \ldots, d/\ell - k_\ell}.$$

Since $|\mathbf{a}| = \alpha d$, the number of partitions of $\mathbf{u}$ into ℓ equal size sets such that $|\mathbf{u}_i \cap \mathbf{a}| = \frac{\alpha d}{\ell}$ for every $i \in [\ell]$ is

$$\binom{\alpha d}{\alpha d/\ell, \ldots, \alpha d/\ell} \cdot \binom{(1-\alpha) d}{(1-\alpha)d/\ell, \ldots, (1-\alpha)d/\ell},$$

which can be lower bounded using Lemma 2.1 by

$$2^{h(1/\ell, \ldots, 1/\ell) d} \cdot d^{-\mathcal{O}(\ell)} = 2^{(\log_2 \ell) \cdot d} \cdot d^{-\mathcal{O}(\ell)} = \ell^d d^{-\mathcal{O}(\ell)}.$$

On the other hand, the number of partitions of $\mathbf{u}$ into ℓ equal size sets is at most ℓ^d. Hence

$$\mathbb{P}\left[|\mathbf{u}_i \cap \mathbf{a}| = \frac{\alpha d}{\ell} \text{ for every } i \in [\ell]\right] \geqslant \ell^d d^{-\mathcal{O}(\ell)} \cdot \ell^{-d} = d^{-\mathcal{O}(\ell)}. \qquad \blacktriangleleft$$

3.2 Certificate of orthogonality

Let $\lambda = \lambda(\alpha, \beta) \in [0,1]$ and $\kappa = \kappa(\alpha, \beta)$ be parameters to be tuned later that only depend on α and β. For every $i \in [\ell]$, we randomly draw a family C_i of $2^{\kappa |\mathbf{u}_i|} \cdot d^c$ subsets of $\mathbf{u}_i$ of size $\lambda |\mathbf{u}_i|$, for some sufficiently large constant c, i.e. we let

$$C_i \subseteq_R \binom{\mathbf{u}_i}{\lambda \cdot |\mathbf{u}_i|} \text{ such that } |C_i| = 2^{\kappa \cdot |\mathbf{u}_i|} \cdot d^c. \tag{2}$$

▶ **Definition 3.2** (Certificate of Orthogonality). *A tuple* $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in C_1 \times \cdots \times C_\ell$ *is a* certificate of orthogonality *if*

(a) *There exists* $\mathbf{a} \in A$ *such that for every* $i \in [\ell]$ *it holds that* $\mathbf{a} \cap \mathbf{u}_i \subseteq \mathbf{c}_i$*, and*

(b) *There exists* $\mathbf{b} \in B$ *such that for every* $i \in [\ell]$ *it holds that* $\mathbf{b} \cap \mathbf{c}_i = \emptyset$.

Clearly, if there exists a certificate of orthogonality, then there exists $\mathbf{a} \in A$ and $\mathbf{b} \in B$ with $\mathbf{a} \cap \mathbf{b} = \emptyset$. However, the converse is not necessarily true. We prove that for sufficiently many sets in each C_i (i.e. large enough parameter κ), if there exists $\mathbf{a} \in A$ and $\mathbf{b} \in B$ with $\mathbf{a} \cap \mathbf{b} = \emptyset$, then with sufficiently high probability there exists a certificate of orthogonality in $(C_1, \ldots, C_\ell)$.

▶ **Lemma 3.3.** *Assume that there exist disjoint* $\mathbf{a} \in A$ *and* $\mathbf{b} \in B$ *with* $|\mathbf{a}| = \alpha d$ *and* $|\mathbf{b}| = \beta d$*. If*

$$\lambda \in [\alpha, 1-\beta] \text{ and}$$

$$\kappa \geqslant h(\lambda) - (1-\alpha-\beta) \cdot h\left(\frac{\lambda - \alpha}{1-\alpha-\beta}\right),$$

then with probability at least $d^{-\mathcal{O}(\ell)}$ *there exists a certificate of orthogonality in* $C_1 \times \cdots \times C_\ell$.

Proof. We condition on the event that $|\mathbf{a} \cap \mathbf{u}_i| = \alpha|\mathbf{u}_i|$ and $|\mathbf{b} \cap \mathbf{u}_i| = \beta|\mathbf{u}_i|$ for all $i \in [\ell]$. By Observation 3.1, this happens with probability at least $d^{-\mathcal{O}(\ell)}$.

Fix $i \in [\ell]$. First, we bound the probability that a set $\mathbf{c}$ sampled uniformly at random from $\binom{\mathbf{u}_i}{\lambda|\mathbf{u}_i|}$ satisfies $\mathbf{a} \cap \mathbf{u}_i \subseteq \mathbf{c}$ and $\mathbf{b} \cap \mathbf{c} = \emptyset$. The number of sets $\mathbf{c} \subseteq \binom{\mathbf{u}_i}{\lambda|\mathbf{u}_i|}$ that satisfy both properties is

$$\binom{(1-\beta-\alpha)|\mathbf{u}_i|}{(\lambda-\alpha)|\mathbf{u}_i|}.$$

Hence, since $|\mathbf{u}_i| = d/\ell$ and using Equation (1), the probability that $\mathbf{c} \subseteq_R \binom{\mathbf{u}_i}{\lambda|\mathbf{u}_i|}$ satisfies properties both properties is at least

$$\binom{(1-\alpha-\beta)|\mathbf{u}_i|}{(\lambda-\alpha)|\mathbf{u}_i|} \cdot \binom{|\mathbf{u}_i|}{\lambda|\mathbf{u}_i|}^{-1} \geqslant 2^{\left((1-\alpha-\beta)\cdot h\left(\frac{\lambda-\alpha}{1-\alpha-\beta}\right) - h(\lambda)\right)\cdot d/\ell} \cdot d^{-\mathcal{O}(1)}$$

Hence if $\kappa \geqslant h(\lambda) - (1-\alpha-\beta) \cdot h\left(\frac{\lambda-\alpha}{1-\alpha-\beta}\right)$, then $|C_i| \geqslant 1/p$ for

$$p := 2^{\left((1-\alpha-\beta)\cdot h\left(\frac{\lambda-\alpha}{1-\alpha-\beta}\right) - h(\lambda)\right)\cdot d/\ell} \cdot d^{-\mathcal{O}(1)}.$$

Therefore, the probability that there exists $\mathbf{c} \in C_i$ satisfying both properties is at least $1 - (1-p)^{1/p} \geqslant 1 - e^{-1}$.

Finally, the probability that for every $i \in [\ell]$ there exists $\mathbf{c} \in C_i$ such that a and b hold is at least $(1 - 2^{-1})^\ell$ which is larger than success probability of Observation 3.1 which is $d^{-\mathcal{O}(\ell)}$. ◀

3.3 Auxiliary Families

In order to efficiently check if there exists a certificate of orthogonality in $C_1 \times \ldots \times C_\ell$, we introduce auxiliary families. For every $i \in [\ell]$ and every set $\mathbf{x} \subseteq \mathbf{u}_i$, define the following two families of sets of C_i:

$$L^{(i)}(\mathbf{x}) := \{\mathbf{c} \in C_i \mid \mathbf{x} \subseteq \mathbf{c}\} \text{ and } R^{(i)}(\mathbf{x}) := \{\mathbf{c} \in C_i \mid \mathbf{x} \cap \mathbf{c} = \emptyset\}. \tag{3}$$

▷ Claim 3.4. The families $L^{(i)}(\mathbf{x})$ and $R^{(i)}(\mathbf{x})$ for every $i \in [\ell]$ and every $\mathbf{x} \subseteq \mathbf{u}_i$ can be computed in $\mathcal{O}(\ell \cdot 2^{2d/\ell})$ time.

Proof. Since, for every $i \in [\ell]$, the number of subsets $\mathbf{x} \subseteq \mathbf{u}_i$ is $2^{d/\ell}$, and the number of sets $\mathbf{c}$ in $C_i \subseteq 2^{\mathbf{u}_i}$ is at most $2^{d/\ell}$, the families of sets can be constructed by iterating over all $i \in [\ell]$, all $\mathbf{x} \subseteq \mathbf{u}_i$ and all $\mathbf{c} \in C_i$ in time at most $\mathcal{O}(\ell \cdot 2^{d/\ell} \cdot 2^{d/\ell})$. ◁

▶ **Lemma 3.5.** *For a fixed $\mathbf{a} \in A$ with $|\mathbf{a}| = \alpha d$, with probability at least $d^{-\mathcal{O}(\ell)}$ it holds that:*

$$|L^{(1)}(\mathbf{a} \cap \mathbf{u}_1)| \cdot |L^{(2)}(\mathbf{a} \cap \mathbf{u}_2)| \cdots |L^{(\ell)}(\mathbf{a} \cap \mathbf{u}_\ell)| \leqslant 2^{c_A \cdot d} \cdot d^{\mathcal{O}(\ell)},$$

where $c_A \coloneqq (1-\alpha) \cdot h\left(\frac{\lambda - \alpha}{1-\alpha}\right) - h(\lambda) + \kappa$.

Proof. We condition on the event that $|\mathbf{a} \cap \mathbf{u}_i| = \alpha |\mathbf{u}_i|$ for all $i \in [\ell]$, which happens with probability at least $d^{-\mathcal{O}(\ell)}$ by Observation 3.1. We first prove that for any $i \in [\ell]$ and any $\mathbf{x} \subseteq \mathbf{u}_i$ with $|\mathbf{x}| = \alpha \cdot |\mathbf{u}_i|$ it holds that

$$\mathbb{E}\left[|L^{(i)}(\mathbf{x})|\right] \leqslant 2^{c_A \cdot |\mathbf{u}_i|} \cdot d^{\mathcal{O}(1)}. \tag{4}$$

By definition, $|L^{(i)}(\mathbf{x})|$ is the number of sets $\mathbf{c} \in C_i$ such that $\mathbf{x} \subseteq \mathbf{c}$. Since sets $\mathbf{c} \in C_i$ are subsets of $\mathbf{u}_i$ drawn uniformly at random such that $|\mathbf{c}| = \lambda |\mathbf{u}_i|$, by Equation (1), the probability that $\mathbf{x} \subseteq \mathbf{c}$ is

$$\mathbb{P}\left[\mathbf{x} \subseteq \mathbf{c}\right] = \binom{(1-\alpha)|\mathbf{u}_i|}{(\lambda-\alpha)|\mathbf{u}_i|} \cdot \binom{|\mathbf{u}_i|}{\lambda |\mathbf{u}_i|}^{-1} \leqslant 2^{\left((1-\alpha)h\left(\frac{\lambda-\alpha}{1-\alpha}\right) - h(\lambda)\right) \cdot |\mathbf{u}_i|}.$$

By the linearity of expectation, this means that:

$$\mathbb{E}\left[|L^{(i)}(\mathbf{x})|\right] = |C_i| \cdot \mathbb{P}\left[\mathbf{x} \subseteq \mathbf{c}\right] \leqslant 2^{\left((1-\alpha)h\left(\frac{\lambda-\alpha}{1-\alpha}\right) - h(\lambda)\right) \cdot |\mathbf{u}_i|} \cdot 2^{\kappa \cdot |\mathbf{u}_i|} \cdot d^{\mathcal{O}(1)},$$

which establishes (4).

Now notice that for any $i \in [\ell]$, by Markov inequality we have that

$$\mathbb{P}\left[|L^{(i)}(\mathbf{x})| \geqslant 2\mathbb{E}\left[|L^{(i)}(\mathbf{x})|\right]\right] \leqslant 1/2,$$

therefore the probability that we have $|L^{(i)}(\mathbf{a} \cap \mathbf{u}_i)| < 2\mathbb{E}\left[|L^{(i)}(\mathbf{a} \cap \mathbf{u}_i)|\right]$ for all $i \in [\ell]$ is at least $1/2^\ell \geqslant d^{-\mathcal{O}(\ell)}$. This directly implies that with probability at least $d^{-\mathcal{O}(\ell)}$ we have

$$|L^{(1)}(\mathbf{a} \cap \mathbf{u}_1)| \cdot |L^{(2)}(\mathbf{a} \cap \mathbf{u}_2)| \cdots |L^{(\ell)}(\mathbf{a} \cap \mathbf{u}_\ell)| \leqslant 2^{c_A \cdot d} \cdot d^{\mathcal{O}(\ell)}.$$ ◀

Similarly, one can prove:

▶ **Lemma 3.6.** *For a fixed $\mathbf{b} \in B$ with $|\mathbf{b}| = \beta d$, with probability at least $d^{-\mathcal{O}(\ell)}$ it holds that:*

$$|R^{(1)}(\mathbf{b} \cap \mathbf{u}_1)| \cdot |R^{(2)}(\mathbf{b} \cap \mathbf{u}_2)| \cdots |R^{(\ell)}(\mathbf{b} \cap \mathbf{u}_\ell)| \leqslant 2^{c_B \cdot d} \cdot d^{\mathcal{O}(\ell)},$$

where $c_B \coloneqq (1-\beta) \cdot h\left(\frac{1-\lambda-\beta}{1-\beta}\right) - h(\lambda) + \kappa$.

Proof. The proof is similar as for Lemma 3.5 by observing that, for every $i \in [\ell]$

$$\mathbb{P}\left[\mathbf{x} \cap \mathbf{c} = \emptyset\right] = \binom{(1-\beta)|\mathbf{u}_i|}{\lambda |\mathbf{u}_i|} \cdot \binom{|\mathbf{u}_i|}{\lambda |\mathbf{u}_i|}^{-1},$$

for sets $\mathbf{x}, \mathbf{c} \subseteq \mathbf{u}_i$ with $|\mathbf{x}| = \beta |\mathbf{u}_i|$ and $|\mathbf{c}| = \lambda |\mathbf{u}_i|$. ◀

3.4 Algorithm

Now, with $L^{(i)}$ and $R^{(i)}$ in hand, we can present the algorithm given in pseudocode in Algorithm 1. First, we construct a set $\mathsf{Candidates}$ of tuples in $C_1 \times \cdots \times C_\ell$ that satisfy property a. We do this by iterating over every set $\mathbf{a} \in A$ and for such $\mathbf{a}$ we iterate over every $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in L^{(1)}(\mathbf{a} \cap \mathbf{u}_1) \times \cdots \times L^{(\ell)}(\mathbf{a} \cap \mathbf{u}_\ell)$ and add it to the set $\mathsf{Candidates}$.

Clearly, after this step, every set that satisfies a is in $\mathsf{Candidates}$. Now, our goal is to decide if there exists at least one tuple in $\mathsf{Candidates}$ that additionally satisfies b. To do so, we iterate over every $\mathbf{b} \in B$ and then over every $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in R^{(1)}(\mathbf{b} \cap \mathbf{u}_1) \times \cdots \times R^{(\ell)}(\mathbf{b} \cap \mathbf{u}_\ell)$. If $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in \mathsf{Candidates}$, then we know that there exists a certificate of orthogonality and immediately report that there exists an orthogonal pair. If none of the loops reported yes, at the end of the algorithm we report that there is no pair of orthogonal vectors.

Algorithm 1 Pseudocode of the algorithm of Lemma 3.7.

1 **function** $\mathsf{OV}(A, B)$:
2 Guess $\alpha d, \beta d \in [d]$ *// Only $\mathcal{O}(d^2)$ possibilities.*
3 Randomly partition $[d] = \mathbf{u}_1 \sqcup \ldots \sqcup \mathbf{u}_\ell$ into equal size sets *// see Sec 3.1*
4 Draw families $C_1, \ldots, C_\ell$ as defined in (2)
5 Construct $L^{(1)}, \ldots, L^{(\ell)}, R^{(1)}, \ldots, R^{(\ell)}$ as defined in (3)
6 Let $\mathsf{Candidates} = \{\}$
7 **foreach** $\mathbf{a} \in A$ **do**
8   **foreach** $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in L^{(1)}(\mathbf{a} \cap \mathbf{u}_1) \times \cdots \times L^{(\ell)}(\mathbf{a} \cap \mathbf{u}_\ell)$ **do** *// break if more than $2^{c_A \cdot d}$*
9   Add $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell)$ to $\mathsf{Candidates}$.
10 **foreach** $\mathbf{b} \in B$ **do**
11   **foreach** $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in R^{(1)}(\mathbf{b} \cap \mathbf{u}_1) \times \cdots \times R^{(\ell)}(\mathbf{b} \cap \mathbf{u}_\ell)$ **do** *// break if more than $2^{c_B \cdot d}$*
12   **if** $(\mathbf{c}_1, \ldots, \mathbf{c}_\ell) \in \mathsf{Candidates}$ **then**
13   **return** Exist Orthogonal Pair
14 **return** No Orthogonal Pair

Finally, to simplify the analysis of Algorithm 1, we stop iterating over every candidate set in Line 8 and Line 11 if the number of iterations exceeds respectively $2^{c_A \cdot d} \cdot d^{\mathcal{O}(\ell)}$ or $2^{c_B \cdot d} \cdot d^{\mathcal{O}(\ell)}$ steps, where c_A and c_B are the constants defined in Lemmas 3.5 and 3.6. When we break these loops, we do not add any sets to $\mathsf{Candidates}$ and immediately continue the loops.

This concludes the description of Algorithm 1. The following lemma certifies its correctness and bounds its running time in terms of the parameters $\alpha, \beta, \lambda(\alpha, \beta)$ and $\kappa(\alpha, \beta)$. We then provide an analytical analysis of the running time by optimizing α and β.

▶ **Lemma 3.7.** *Given $A, B \subseteq 2^{[d]}$, there exists an*

$$\left(2^{c_A \cdot d}|A| + 2^{c_B \cdot d}|B| + \ell \cdot 2^{2d/\ell}\right) \cdot d^{\mathcal{O}(\ell)}$$

time randomized, one-sided error algorithm with $1 - 2^{-d^{\Omega(1)}}$ probability of success that decides if there exist $\mathbf{a} \in A$ and $\mathbf{b} \in B$ such that $\mathbf{a} \cap \mathbf{b} = \emptyset$, where c_A and c_B are as in Lemmas 3.5 and 3.6.

Proof. We repeat Algorithm 1 $d^{\mathcal{O}(\ell)}$ times and return true iff any execution returns true. Consider Algorithm 1. The preprocessing in Section 3.1 takes $\mathcal{O}(d)$ time. By Claim 3.4, the construction of data structures $L^{(1)}, \dots, L^{(\ell)}, R^{(1)}, \dots, R^{(\ell)}$ takes $\mathcal{O}(\ell \cdot 2^{2d/\ell})$ time. Finally, the for loops in Algorithm 1 take in total $(2^{c_A \cdot d}|A| + 2^{c_B \cdot d}|B|) \cdot d^{\mathcal{O}(\ell)}$ time. This concludes the running time analysis. It remains to prove the correctness.

Note that if there do not exist any disjoint pairs in A and B, then Algorithm 1 never returns true. Hence, to bound the probability of false-negative, assume that $\mathbf{a} \in A$ and $\mathbf{b} \in B$ are orthogonal. By Lemma 3.3, with probability at least $d^{-\mathcal{O}(\ell)}$, there exists a certificate of orthogonality in $C_1 \times \cdots \times C_\ell$ for $\mathbf{a}$ and $\mathbf{b}$. By Lemmas 3.5 and 3.6, with probability at least $d^{-\mathcal{O}(\ell)}$, the loop in Line 8 corresponding to $\mathbf{a}$ and the loop in Line 11 corresponding to $\mathbf{b}$ are fully executed. Observe that by Lemma 3.5, with probability at least $d^{-\mathcal{O}(\ell)}$, the loop in Line 8 corresponding to $\mathbf{a}$ is fully executed. If that happens, the set $\mathsf{Candidates}$ contains every tuple $(\mathbf{c}_1, \dots, \mathbf{c}_\ell) \in L^{(1)}(\mathbf{a} \cap \mathbf{u}_1) \times \cdots \times L^{(\ell)}(\mathbf{a} \cap \mathbf{u}_\ell)$. In particular, it contains the certificate of orthogonality for $\mathbf{a}$ and $\mathbf{b}$. Moreover, by Lemma 3.6, with probability at least $d^{-\mathcal{O}(\ell)}$, the loop in Line 11 corresponding to $\mathbf{b}$ is also fully executed. Hence this certificate will be detected and the algorithm returns true.

To conclude, if $\mathbf{a} \in A$ and $\mathbf{b} \in B$ are orthogonal, Algorithm 1 returns true with probability $d^{-\mathcal{O}(\ell)}$. To guarantee probability of success $1 - 2^{-d^{\Omega(1)}}$, it suffices to repeat Algorithm 1 $d^{\mathcal{O}(\ell)}$ many times. ◀

Proof of Theorem 1.1. Assume that there exist $\mathbf{a} \in A$ and $\mathbf{b} \in B$ such that $\mathbf{a} \cap \mathbf{b} =$ and let $|\mathbf{a}| = \alpha d$ and $|\mathbf{b}| = \beta d$ for some $\alpha, \beta \in [0,1]$. By Lemma 3.7, one can find the pair $(\mathbf{a}, \mathbf{b})$ in time

$$\widetilde{\mathcal{O}}\left(2^{c_A \cdot d}|A| + 2^{c_B \cdot d}|B| + \ell \cdot 2^{2d/\ell}\right),$$

where λ and κ are parameters depending on α and β to be optimized, and c_A and c_B are constants depending on α, β, λ and κ. Note that we have selected $\ell = 100$, so the last term is upper bounded by $\widetilde{\mathcal{O}}(1.25^d \cdot n)$. Therefore, it suffices to prove that for any α, β, by properly selecting λ and κ, we can bound $2^{c_A d} + 2^{c_B d}$ by $\widetilde{\mathcal{O}}(1.25^d)$.

First, observe that the constants c_A and c_B defined in Lemmas 3.5 and 3.6 grow with the parameter κ. On the other hand, by Lemma 3.3, we need $\kappa \geqslant h(\lambda) - (1-\alpha-\beta) \cdot h(\frac{\lambda-\alpha}{1-\alpha-\beta})$ for the correctness analysis in Lemma 3.7 to hold. Hence to minimise $2^{c_A d} + 2^{c_B d}$, we choose κ to be minimal, i.e. we set $\kappa \coloneqq h(\lambda) - (1-\alpha-\beta) \cdot h(\frac{\lambda-\alpha}{1-\alpha-\beta})$. Then we have

$$\begin{aligned} c_A &= (1-\alpha) \cdot h\left(\frac{\lambda-\alpha}{1-\alpha}\right) - (1-\alpha-\beta) \cdot h\left(\frac{\lambda-\alpha}{1-\alpha-\beta}\right), \\ c_B &= (1-\beta) \cdot h\left(\frac{1-\lambda-\beta}{1-\beta}\right) - (1-\alpha-\beta) \cdot h\left(\frac{\lambda-\alpha}{1-\alpha-\beta}\right), \end{aligned} \tag{5}$$

and using Lemma 2.1, we get the following bound

$$2^{c_A d} + 2^{c_B d} \leqslant \widetilde{\mathcal{O}}\left(\left(\binom{(1-\alpha)d}{(\lambda-\alpha)d} + \binom{(1-\beta)d}{(1-\lambda-\beta)d}\right) \cdot \binom{(1-\alpha-\beta)d}{(\lambda-\alpha)d}^{-1}\right).$$

So by setting $\lambda \coloneqq (1+\alpha-\beta)/2$ and noticing that $\binom{(1-\alpha-\beta)d}{(1-\alpha-\beta)d/2}^{-1} \leqslant \widetilde{\mathcal{O}}(2^{-(1-\alpha-\beta)d})$, we can further bound

$$\begin{aligned} 2^{c_A d} + 2^{c_B d} &\leqslant \widetilde{\mathcal{O}}\left(\left(\binom{(1-\alpha)d}{(1-\alpha-\beta)d/2} + \binom{(1-\beta)d}{(1-\alpha-\beta)d/2}\right) \cdot \binom{(1-\alpha-\beta)d}{(1-\alpha-\beta)d/2}^{-1}\right) \\ &\leqslant \widetilde{\mathcal{O}}\left(\left(\binom{(1-\alpha)d}{(1-\alpha-\beta)d/2} + \binom{(1-\beta)d}{(1-\alpha-\beta)d/2}\right) \cdot 2^{-(1-\alpha-\beta)d}\right). \end{aligned}$$

Notice that both binomials select $(1-\alpha-\beta)d/2$ elements and recall that we assume $\alpha \leqslant \beta$. Hence the left term dominates the right term and by using Lemma 2.1 we bound the expression

$$\begin{aligned}2^{c_A d} + 2^{c_B d} &\leqslant \widetilde{\mathcal{O}}\left(\binom{(1-\alpha)d}{(1-\alpha-\beta)d/2} \cdot 2^{-(1-\alpha-\beta)d}\right) \\ &\leqslant \widetilde{\mathcal{O}}\left(2^{h\left(\frac{(1-\alpha-\beta)/2}{1-\alpha}\right)\cdot(1-\alpha)d} \cdot 2^{-(1-\alpha-\beta)d}\right) \\ &= \widetilde{\mathcal{O}}\left(2^{h\left(\frac{1}{2}-\frac{\beta}{2(1-\alpha)}\right)\cdot(1-\alpha)d-(1-\alpha-\beta)d}\right) = \widetilde{\mathcal{O}}\left(2^{\left(h\left(\frac{1}{2}-\frac{\beta}{2(1-\alpha)}\right)-1\right)\cdot(1-\alpha)d+\beta d}\right).\end{aligned}$$

The term involving α, that is $\left(h\left(\frac{1}{2}-\frac{\beta}{2(1-\alpha)}\right)-1\right)\cdot(1-\alpha)d$, is maximized when $\alpha = 0$. This setting allows β to take any value in $[0,1]$. Hence the above term is maximized when $\alpha = 0$ and $\beta \in [0,1]$ is maximizing $h(1/2+\beta/2)-1+\beta$. The derivative is $(\log_2(4-4\beta)-\log_2(\beta+1))/2$, therefore we conclude that this expression has a single local maximum at $\beta = 3/5$. In that case, the running time is bounded by:

$$\widetilde{\mathcal{O}}\left(2^{(h(1/5)-1+3/5)\cdot d} \cdot n\right) = \widetilde{\mathcal{O}}\left(1.25^d \cdot n\right). \qquad ◀$$

▶ Remark 8. The computer evaluation suggests that the maximum is obtained for the choice $\alpha = \beta = 1/3$, for which the best $\lambda = 1/2$. See Figure 1. This choice yields the running time:

$$\widetilde{\mathcal{O}}\left(\left(\frac{4}{3}\right)^{d/2} \cdot n\right) \leqslant \mathcal{O}(1.16^d \cdot n).$$

4 Algorithm for k-Orthogonal Vectors (k-OV)

We now show that for every fixed $k \geqslant 2$, there exists an algorithm for k-OV with $\mathcal{O}(2^{(1-\varepsilon_k)\cdot d} \cdot n)$ running time, where $\varepsilon_k > 0$ is a constant depending only on k. We remark that ε_k tends to 0 as k tends to infinity. This is consistent with the lower bound we present in Theorem 1.3.

▶ **Theorem 1.2.** *For every $k \geqslant 2$, there exists $\varepsilon_k > 0$ such that the k-Orthogonal Vectors problem can be solved deterministically in time $\mathcal{O}(2^{(1-\varepsilon_k)d} \cdot n)$.*

To prove Theorem 1.2, we will need the following combinatorial algorithm that follows from [4, Theorem 1].

▶ **Lemma 4.1.** *Let $k \in \mathbb{N}$ be a fixed integer. Given $A_1, \dots, A_k \subseteq 2^{[d]}$, we can count the number of tuples $(\mathbf{a}_1, \dots, \mathbf{a}_k) \in A_1 \times \dots \times A_k$ such that $\mathbf{a}_1 \cap \mathbf{a}_2 \cap \ldots \cap \mathbf{a}_k = \emptyset$ in time at most $\mathcal{O}(d \cdot (|{\downarrow}A_1| + \ldots + |{\downarrow}A_k|))$.*

Proof. This directly follows from [4, Theorem 1] extended to k families. For completeness, we detail the proof here.

Let #OV be the number of solutions for the given instance, i.e. the number of tuples $(\mathbf{a}_1, \dots, \mathbf{a}_k) \in A_1 \times \dots \times A_k$ such that $\mathbf{a}_1 \cap \dots \cap \mathbf{a}_k = \emptyset$. We can express #OV as follows, where the second equality comes from the fact that any non-empty set has as many subsets of even size as subsets of odd size.

$$
\begin{aligned}
\#\mathrm{OV} &= \sum_{\mathbf{a}_1 \in A_1} \sum_{\mathbf{a}_2 \in A_2} \cdots \sum_{\mathbf{a}_k \in A_k} [\![\mathbf{a}_1 \cap \mathbf{a}_2 \cap \cdots \cap \mathbf{a}_k = \emptyset]\!] \\
&= \sum_{\mathbf{a}_1 \in A_1} \sum_{\mathbf{a}_2 \in A_2} \cdots \sum_{\mathbf{a}_k \in A_k} \sum_{\mathbf{x} \subseteq [d]} (-1)^{|\mathbf{x}|} \cdot [\![\mathbf{x} \subseteq \mathbf{a}_1 \cap \mathbf{a}_2 \cap \cdots \cap \mathbf{a}_k]\!] \\
&= \sum_{\mathbf{x} \subseteq [d]} (-1)^{|\mathbf{x}|} \sum_{\mathbf{a}_1 \in A_1} \sum_{\mathbf{a}_2 \in A_2} \cdots \sum_{\mathbf{a}_k \in A_k} [\![\mathbf{x} \subseteq \mathbf{a}_1]\!] \cdot [\![\mathbf{x} \subseteq \mathbf{a}_2]\!] \ldots [\![\mathbf{x} \subseteq \mathbf{a}_k]\!] \\
&= \sum_{\mathbf{x} \in \bigcup_{i=1}^{k} \downarrow A_i} (-1)^{|\mathbf{x}|} \cdot f_1(\mathbf{x}) \cdot f_2(\mathbf{x}) \cdots f_k(\mathbf{x})
\end{aligned}
$$

where $f_i(\mathbf{x}) \coloneqq \sum_{\mathbf{a} \in A_i} [\![\mathbf{x} \subseteq \mathbf{a}]\!]$ for $i \in [k]$. For each $i \in [k]$, we compute the values of $f_i(\cdot)$ for all $\mathbf{x} \in \downarrow A_i$ in time $\mathcal{O}(d \cdot |\downarrow A_i|)$ as follows. For each element $j \in \{0, 1, \ldots, d\}$ and subset $\mathbf{x} \in \downarrow A_i$ define $g_j(\mathbf{x})$ to be the number of sets $\mathbf{a} \in A_i$ such that $\mathbf{x} \subseteq \mathbf{a}$ and $\mathbf{a} \cap \{1, 2, \ldots, j\} = \mathbf{x} \cap \{1, 2, \ldots, j\}$. In particular, $g_0(\mathbf{x}) = f_i(\mathbf{x})$ and $g_d(\mathbf{x}) = [\![\mathbf{x} \in A_i]\!]$. By induction on j, we can prove that

$$
g_{j-1}(\mathbf{x}) = [\![j \notin \mathbf{x}]\!] \cdot g_j(\mathbf{x}) + [\![\mathbf{x} \cup \{j\} \in \downarrow A_i]\!] \cdot g_j(\mathbf{x} \cup \{j\}).
$$

Using this recurrence relation, we can compute $g_j(\mathbf{x})$ for all $\mathbf{x} \in \downarrow A_i$ and for all $j \in \{0, 1, \ldots, d\}$, and thus in particular $f_i(\mathbf{x})$, in time $\mathcal{O}(d |\downarrow A_i|)$. By repeating this for every $i \in [k]$ and using the above formula for #OV, we can count the number of k-OV solutions in time $\mathcal{O}(d \cdot (|\downarrow A_1| + \cdots + |\downarrow A_k|))$. ◀

Proof of Theorem 1.2. Fix $k \geqslant 3$ and let $0 < \varepsilon_k \leqslant 1/2$ be a parameter depending only on k that we define later. Consider a k-OV instance $A_1, \ldots, A_k \subseteq 2^{[d]}$. Note that if the down-closures of $A_1, \ldots, A_k$ are all small enough, i.e. of size at most $\mathcal{O}(2^{(1-\varepsilon_k)d} \cdot n)$, then Lemma 4.1 provides the desired running time. If this is not the case, then we reduce the k-OV instance to a k'-OV instance for smaller $k' < k$. Since Lemma 2.2 proves the statement for $k = 2$ with value $\varepsilon_2 = 1/2$, we only need to show that ε_k stays positive.

Assume that there exists a solution $(\mathbf{a}_1, \ldots, \mathbf{a}_k) \in A_1 \times \cdots \times A_k$ such that $\mathbf{a}_1 \cap \cdots \cap \mathbf{a}_k = \emptyset$. Let $\alpha_1, \ldots, \alpha_k \in [0, 1]$ be such that $|\mathbf{a}_i| = \alpha_i d$ for every $i \in [k]$. By enumerating all possible sizes, we can guess the values of $\alpha_1, \ldots, \alpha_k$ in $\mathcal{O}(d^k)$ time. This factor will be negligible compared to $2^{(1-\varepsilon_k)d}$ as $k \geqslant 2$ is a fixed constant. After guessing $\alpha_1, \ldots, \alpha_k$, we assume that the family A_i contains only sets of cardinality exactly $\alpha_i d$ for every $i \in [k]$. Next, we distinguish between two cases.

If $\alpha_i \leqslant (1 - \varepsilon_k)$ for every $i \in [k]$, then the size of the down-closure of A_i is $|\downarrow A_i| = \mathcal{O}(2^{\alpha_i d} \cdot |A_i|)$. Using the algorithm of Lemma 4.1, we solve the k-OV instance $A_1, \ldots, A_k$ in time $\mathcal{O}(2^{(1-\varepsilon_k)d} \cdot n \cdot d)$.

If there exists $i \in [k]$ such that $\alpha_i > (1 - \varepsilon_k)$, then we recurse as follows. By reordering the families, we can assume that $\alpha_1 \leqslant \ldots \leqslant \alpha_k$. Let $\ell \in [k]$ be the smallest index such that $\alpha_\ell > (1 - \varepsilon_k)$, i.e. $\alpha_1 \leqslant \ldots \leqslant \alpha_{\ell-1} \leqslant (1 - \varepsilon_k) < \alpha_\ell \leqslant \ldots \leqslant \alpha_k$. Note that $(\mathbf{a}_\ell, \ldots, \mathbf{a}_k) \in A_\ell \times \cdots \times A_k$. Furthermore, for every $i \in \{\ell, \ldots, k\}$, since $\varepsilon_k \leqslant 1/2$, we can bound $|A_i| \leqslant \sum_{j=(1-\varepsilon_k)d}^{d} \binom{d}{j} \leqslant \binom{d}{(1-\varepsilon_k)d} \cdot d = \binom{d}{\varepsilon_k d} \cdot d$. By Equation (1) we thus get $|A_i| \leqslant 2^{h(\varepsilon_k)d} \cdot d$. Hence we can enumerate $A_\ell \times \cdots \times A_k$ to guess the sets $\mathbf{a}_\ell, \ldots, \mathbf{a}_k$ in time $\mathcal{O}\left(2^{h(\varepsilon_k)d(k-\ell+1)}\right)$. To find the remaining sets $\mathbf{a}_1, \ldots, \mathbf{a}_{\ell-1}$, observe that we can restrict the universe to $\mathbf{r} \coloneqq \mathbf{a}_\ell \cap \cdots \cap \mathbf{a}_k$ as we have $\mathbf{a}_1 \cap \cdots \cap \mathbf{a}_k = \mathbf{a}_1 \cap \cdots \cap \mathbf{a}_{\ell-1} \cap \mathbf{r} = (\mathbf{a}_1 \cap \mathbf{r}) \cap \cdots \cap (\mathbf{a}_{\ell-1} \cap \mathbf{r})$. So we recursively solve the $(\ell - 1)$-OV instance $A_1^r, \ldots, A_{\ell-1}^r$ where $A_i^r \coloneqq \{\mathbf{a} \cap \mathbf{r} \mid \mathbf{a} \in A_i\}$ for all $i \in \{1, \ldots, \ell - 1\}$. By induction, this takes time $\widetilde{\mathcal{O}}(2^{(1-\varepsilon_{\ell-1})d} \cdot n)$. In total, we solve

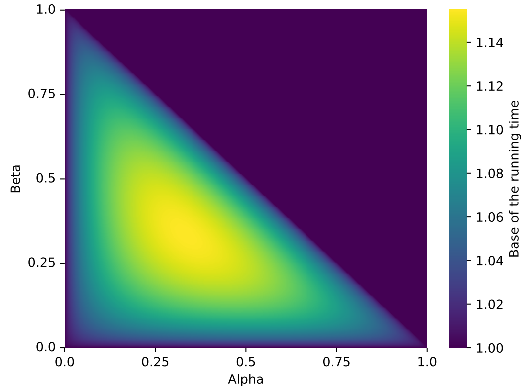

Figure 1 For different values of $\alpha, \beta \in [0,1]$, we optimize the value of $\lambda(\alpha,\beta)$ to minimise the running time bound $\widetilde{\mathcal{O}}(x^d \cdot n)$ of our algorithm, where the base of the running time is $x = \max\{2^{c_A}, 2^{c_B}\}$, and c_A and c_B are as in (5). The running time is maximized for $\alpha = \beta = 1/3$, for which the best found value for λ is $\lambda = 1/2$, resulting in a running time base $x \leqslant 1.16$.

the k-OV instance $A_1, \ldots, A_k$ in time $\widetilde{\mathcal{O}}(2^{h(\varepsilon_k)d(k-\ell+1)} \cdot 2^{(1-\varepsilon_{\ell-1})d} \cdot n) = \widetilde{\mathcal{O}}(2^{(1-\varepsilon_k)d} \cdot n)$ for $\varepsilon_k \leqslant \varepsilon_{\ell-1} - h(\varepsilon_k)\cdot(k-\ell+1)$. Note that since $\varepsilon_k \leqslant 1/2$, the function $\varepsilon_k \mapsto \varepsilon_k + h(\varepsilon_k)\cdot(k-\ell+1)$ is increasing. Furthermore, $\lim_{\varepsilon_k \to 0^+} \varepsilon_k + h(\varepsilon_k)\cdot(k-\ell+1) = 0$. Since $\varepsilon_2 > 0$ by Lemma 2.2, we can set ε_k to a positive value satisfying $\varepsilon_k \leqslant \min\{1/2, \varepsilon_{\ell-1} - h(\varepsilon_k)\cdot(k-\ell+1)\}$. Achieving running time $\mathcal{O}(2^{(1-\varepsilon_k)d} \cdot n)$, instead of $\widetilde{\mathcal{O}}(2^{(1-\varepsilon_k)d} \cdot n)$ follows directly by scaling all ε_k (e.g. by 0.99). ◀

5 Faster k-Orthogonal Vectors under the Asymptotic Rank Conjecture

In the Set Cover problem, we are given a universe $\mathbf{u}$, a set family F and a positive integer t. The goal is to determine if there exist t sets in F whose union is $\mathbf{u}$. We say that a set family F is δ-bounded for some $\delta \in (0,1)$ if no set in F is larger than $\delta|\mathbf{u}|$. Björklund et al. [3] showed that under the Asymptotic Rank Conjecture, Set Cover instances with δ-bounded set families can be solved faster than $2^{|\mathbf{u}|}$ time.

▶ **Theorem 5.1** (Theorem 1.2 [3]). *Let $\delta \in (0, 1/4)$ be a fixed constant. Assuming the Asymptotic Rank Conjecture, there exist $c > 0$ and $\varepsilon > 0$, such that any Set Cover instance $(\mathbf{u}, F, t)$ where F is δ-bounded, can be solved deterministically in $\mathcal{O}((2-\varepsilon)^{|\mathbf{u}|} \cdot |\mathbf{u}|^c)$ time.*

Notice, that the running time of the above theorem does not depend on $|F|$ (which is upper bounded by $\binom{|\mathbf{u}|}{|\mathbf{u}|/4} \leqslant (\frac{e|\mathbf{u}|}{|\mathbf{u}|/4})^{|\mathbf{u}|/4} < 1.82^{|\mathbf{u}|}$). In particular, it works even if $|F| = \binom{|\mathbf{u}|}{\delta|\mathbf{u}|}$. We exploit that fact to prove the following result, which eliminates the necessity for δ-boundedness.

▶ **Corollary 5.2.** *Assuming the Asymptotic Rank Conjecture, there exist $c > 0$ and $\varepsilon > 0$, such that any Set Cover instance $(\mathbf{u}, F, t)$ can be solved in $\mathcal{O}(2^{(1-\varepsilon)\cdot|\mathbf{u}|} \cdot (|\mathbf{u}| + |F|)^c)$ time.*

Proof. Let $\mathbf{s}_1, \ldots, \mathbf{s}_t \in F$ be a solution to the given Set Cover instance, and c', ε' be the constants from Theorem 5.1. Let $\alpha = 1/5$. If $|\mathbf{s}_i| \leqslant \alpha \cdot |\mathbf{u}|$ for every $i \in [t]$, then we can solve the Set Cover instance in $\mathcal{O}((2-\varepsilon')^{|\mathbf{u}|} \cdot |\mathbf{u}|^{c'})$ time using Theorem 5.1, by considering only the sets of size at most $\alpha \cdot |\mathbf{u}|$ in F.

Otherwise, there exists $i \in [t]$ such that $|\mathbf{s}_i| > \alpha \cdot |\mathbf{u}|$. By guessing $\mathbf{s}_i$ among the sets in F (which takes an extra $\mathcal{O}(|F|)$ factor in the running time), we can restrict the universe to $\mathbf{u}' = \mathbf{u} \setminus \mathbf{s}_i$ and solve the remaining $(t-1)$-Set Cover instance $(\mathbf{u}', F', t-1)$, where $F' = \{\mathbf{s} \cap \mathbf{u}' \mid \mathbf{s} \in F\}$. As the size of the universe is $|\mathbf{u}'| \leqslant (1-\alpha) \cdot |\mathbf{u}|$, we can solve the remaining instance in $2^{(1-\alpha)\cdot|\mathbf{u}|} \cdot (|\mathbf{u}| + |F|)^{\mathcal{O}(1)}$ time by applying the standard dynamic programming algorithm for Set Cover (e.g. Theorem 6.1 in [10]), that runs in $2^{|\mathbf{u}'|} \cdot (|\mathbf{u}'| + |F'|)^{\mathcal{O}(1)}$ time.

It follows that for sufficiently large c and sufficiently small $\varepsilon > 0$, one can solve the Set Cover instance $(\mathbf{u}, F, t)$ in $\mathcal{O}(2^{(1-\varepsilon)\cdot|\mathbf{u}|} \cdot (|\mathbf{u}| + |F|)^c)$ time. ◀

We now show an equivalence between Set Cover and k-Orthogonal Vectors in the setting of exact algorithms.

▶ **Theorem 5.3.** *The following statements are equivalent:*

(a) *There exist $c > 0$ and $\varepsilon > 0$, such that k-Orthogonal Vectors on n vectors in dimension d can be solved in $\mathcal{O}(2^{(1-\varepsilon)\cdot d} \cdot n^c)$ time (algorithm works uniformly for a given $k \in \mathbb{N}$).*

(b) *There exists $c > 0$ and $\varepsilon > 0$, such that Set Cover on a universe $\mathbf{u}$ and a set family F can be solved in $\mathcal{O}(2^{(1-\varepsilon)\cdot|\mathbf{u}|} \cdot (|\mathbf{u}| + |F|)^c)$ time.*

Proof.

Implication (a) $\Rightarrow$ (b). Let $(\mathbf{u}, F, t)$ be an instance of Set Cover. Let $c > 0$ and $\varepsilon > 0$ be fixed constants such that t-Orthogonal Vectors is solvable in $\mathcal{O}(2^{(1-\varepsilon)\cdot d} \cdot n^c)$ time. Consider the instance $A_1, \ldots, A_t$ of t-Orthogonal Vectors where $A_1 = \ldots = A_t$ and $\mathbf{a} \subseteq \mathbf{u}$ is in A_1 iff $\mathbf{u} \setminus \mathbf{a} \in F$. Since we constructed $|F|$ vectors of dimension $|\mathbf{u}|$, it suffices to show that there exists a solution for the Set Cover instance iff there exists a solution for the t-Orthogonal Vectors instance.

If there exists a solution $\mathbf{a}_1, \ldots, \mathbf{a}_t$ for the t-Orthogonal Vectors instance, then $\bigcap_{i=1}^t \mathbf{a}_i = \emptyset$, meaning that $\bigcup_{i=1}^t \mathbf{u} \setminus \mathbf{a}_i = \mathbf{u}$. But $\mathbf{u} \setminus \mathbf{a}_i \in F$ by construction, therefore $\mathbf{u} \setminus \mathbf{a}_1, \ldots, \mathbf{u} \setminus \mathbf{a}_t$ is a solution for the Set Cover instance. Similarly, if there exists a solution $\mathbf{s}_1, \ldots, \mathbf{s}_t$ for the Set Cover instance, then $\bigcup_{i=1}^t \mathbf{s}_i = \mathbf{u}$, meaning that $\bigcap_{i=1}^t \mathbf{u} \setminus \mathbf{s}_i = \emptyset$. But $\mathbf{u} \setminus \mathbf{s}_i \in A_i$ by construction, therefore $\mathbf{u} \setminus \mathbf{s}_1, \ldots, \mathbf{u} \setminus \mathbf{s}_t$ is a solution for the t-Orthogonal Vectors problem.

Implication (a) $\Leftarrow$ (b). Let $(A_1, \ldots, A_k)$ be an instance of k-Orthogonal Vectors of dimension d and let $c > 0$ and $\varepsilon > 0$ be fixed constants such that Set Cover is solvable in $\mathcal{O}(2^{(1-\varepsilon)\cdot|\mathbf{u}|} \cdot (|\mathbf{u}| + |F|)^c)$ time. Consider the instance $(\mathbf{u}, F, t)$ of Set Cover where $t := k$, $\mathbf{u} := [d+k]$, and $F := \bigcup_{i\in[k]} F_i$, where $F_i := \{\{d+i\} \cup ([d] \setminus \mathbf{a}) \mid \mathbf{a} \in A_i\}$. Since k is a constant, the size of the universe is $|\mathbf{u}| = d + \mathcal{O}(1)$ and $F = \mathcal{O}(n)$. Hence, the running time of the Set Cover algorithm is $\mathcal{O}(2^{(1-\varepsilon)\cdot d} \cdot n^c)$. It suffices to show that there exists a solution for the k-Orthogonal Vectors instance iff there exists a solution for the constructed Set Cover instance.

Let $\mathbf{a}_1, \ldots, \mathbf{a}_k$ be a solution for the k-Orthogonal Vectors instance. Then $\bigcap_{i=1}^k \mathbf{a}_i = \emptyset$, meaning that $\bigcup_{i=1}^k [d] \setminus \mathbf{a}_i = [d]$. Therefore, the sets $\{\{d+i\} \cup ([d] \setminus \mathbf{a}_i) \mid i \in [k]\}$ cover the universe $\mathbf{u}$, and thus there exists a solution for the Set Cover instance. Similarly, let $\mathbf{s}_1, \ldots, \mathbf{s}_k$ be a solution for the Set Cover instance. As $t = k$ and for every $i \in [k]$ it holds that $\{d+i\} \in \mathbf{s}$ iff $\mathbf{s} \in F_i$, we can assume that $\mathbf{s}_i \in F_i$ for every $i \in [k]$. Because $\mathbf{s}_1, \ldots \mathbf{s}_k$ is a valid cover, we have $\bigcup_{i=1}^k \mathbf{s}_i = \mathbf{u}$, and thus $\bigcup_{i=1}^k (\mathbf{s}_i \setminus \{d+i\}) = [d]$. Therefore $\bigcap_{i=1}^k [d] \setminus \mathbf{s}_i = \emptyset$, meaning that the sets $\{\mathbf{u} \setminus \mathbf{s}_i \mid i \in [k]\}$ are a solution for the k-Orthogonal Vectors instance. ◀

The faster algorithm for k-Orthogonal Vectors follows directly from Corollary 5.2 and Theorem 5.3.

▶ **Theorem 1.4.** *Assuming the Asymptotic Rank Conjecture, there exist $c > 0$ and $\varepsilon > 0$, such that any k-Orthogonal Vectors instance can be solved in $\mathcal{O}(2^{(1-\varepsilon)\cdot d} \cdot n^c)$ time for every $k \in \mathbb{N}$.*

On the other hand, the Set Cover Conjecture implies that there is no $\varepsilon > 0$ for which Set Cover is solvable in $\mathcal{O}^\star(2^{(1-\varepsilon)\cdot|\mathbf{u}|})$ time. In particular, Theorem 5.3 implies the lower bound for k-Orthogonal Vectors stated in Theorem 1.3, based on the Set Cover Conjecture (which is incompatible with the Asymptotic Rank Conjecture [5]).

▶ **Theorem 1.3.** *For every $c > 0$ and $\varepsilon > 0$, there exists $k \in \mathbb{N}$ such that the k-Orthogonal Vectors problem is not solvable in $\mathcal{O}(2^{(1-\varepsilon)d} \cdot n^c)$ time, assuming the Set Cover Conjecture.*

References

1. Amir Abboud, Ryan Williams, and Huacheng Yu. More applications of the polynomial method to algorithm design. In *Proceedings of the Twenty-sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 218–230. Society for Industrial and Applied Mathematics, 2015. URL: `http://dl.acm.org/citation.cfm?id=2722129.2722146`.
2. Josh Alman and Baitian Li. Kronecker Powers, Orthogonal Vectors, and the Asymptotic Spectrum. In *66th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2025*, pages 1411–1441. IEEE, 2025. `doi:10.1109/FOCS63196.2025.00074`.
3. Andreas Björklund, Radu Curticapean, Thore Husfeldt, Petteri Kaski, and Kevin Pratt. Fast deterministic chromatic number under the asymptotic rank conjecture. In *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025*, pages 2804–2818. SIAM, 2025. `doi:10.1137/1.9781611978322.91`.
4. Andreas Björklund, Thore Husfeldt, Petteri Kaski, and Mikko Koivisto. Counting Paths and Packings in Halves. In Amos Fiat and Peter Sanders, editors, *Algorithms - ESA 2009, 17th Annual European Symposium. Proceedings*, 2009.
5. Andreas Björklund and Petteri Kaski. The Asymptotic Rank Conjecture and the Set Cover Conjecture Are Not Both True. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024*, pages 859–870. ACM, 2024. `doi:10.1145/3618260.3649656`.
6. Timothy M. Chan and Ryan Williams. Deterministic APSP, Orthogonal Vectors, and More: Quickly Derandomizing Razborov-Smolensky. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016*, pages 1246–1255. SIAM, 2016. `doi:10.1137/1.9781611974331.CH87`.
7. Lijie Chen and Ryan Williams. An Equivalence Class for Orthogonal Vectors. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019*, 2019.
8. Nikolai Chukhin, Alexander S. Kulikov, Ivan Mihajlin, and Arina Smirnova. Conditional complexity hardness: Monotone circuit size, matrix rigidity, and tensor rank under NSETH and beyond. *Electron. Colloquium Comput. Complex.*, TR25-038, 2025. URL: `https://eccc.weizmann.ac.il/report/2025/038`.
9. Imre Csiszár and Paul C Shields. *Information theory and statistics: A tutorial.* Now Publishers Inc, 2004.
10. Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms.* Springer, 2015. `doi:10.1007/978-3-319-21275-3`.
11. Fedor V. Fomin, Daniel Lokshtanov, Fahad Panolan, and Saket Saurabh. Efficient Computation of Representative Families with Applications in Parameterized and Exact Algorithms. *J. ACM*, 63(4):29:1–29:60, 2016. `doi:10.1145/2886094`.
12. Jiawei Gao, Russell Impagliazzo, Antonina Kolokolova, and Ryan Williams. Completeness for First-order Properties on Sparse Structures with Algorithmic Applications. *ACM Trans. Algorithms*, 15(2):23:1–23:35, 2019. `doi:10.1145/3196275`.
13. Jesper Nederlof and Karol Węgrzycki. Improving Schroeppel and Shamir's algorithm for Subset Sum via Orthogonal Vectors. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1670–1683. ACM, 2021. `doi:10.1145/3406325.3451024`.

14 Kevin Pratt. A Stronger Connection between the Asymptotic Rank Conjecture and the Set Cover Conjecture. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024*, pages 871–874. ACM, 2024. doi:10.1145/3618260.3649620.

15 Timothy W. Randolph and Karol Węgrzycki. Beating Meet-in-the-Middle for Subset Balancing Problems. *CoRR*, abs/2511.10823, 2025. doi:10.48550/arXiv.2511.10823.

16 Herbert Robbins. A remark on Stirling's formula. *The American mathematical monthly*, 62(1):26–29, 1955.

17 Ryan Williams. A new algorithm for optimal 2-constraint satisfaction and its implications. *Theor. Comput. Sci.*, 348(2-3):357–365, 2005. doi:10.1016/J.TCS.2005.09.023.

18 Ryan Williams. The Orthogonal Vectors Conjecture and Non-Uniform Circuit Lower Bounds. In *65th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2024*, pages 1372–1387. IEEE, 2024. doi:10.1109/FOCS61266.2024.00088.

19 Virginia Vassilevska Williams. Hardness of Easy Problems: Basing Hardness on Popular Conjectures such as the Strong Exponential Time Hypothesis (Invited Talk). In *IPEC 2015*, pages 17–29, 2015. doi:10.4230/LIPIcs.IPEC.2015.17.