Electronic Colloquium on Computational Complexity, Report No. 75 (2014)

A simple proof that AND-compression of NP-complete problems is hard

Holger Dell

LIAFA, Université Paris Diderot

May 16, 2014*

Abstract

Drucker [1] proved the following result: Unless the unlikely complexity-theoretic collapse $\mathsf{coNP} \subseteq \mathsf{NP/poly}$ occurs, there is no *AND-compression* for SAT. The result has implications for the compressibility and kernelizability of a whole range of NP-complete parameterized problems. We present a simple proof of this result.

An AND-compression is a deterministic polynomial-time algorithm that maps a set of SAT-instances $x_1, \dots, x_t$ to a single SAT-instance y of size $\mathsf{poly}(\max_i |x_i|)$ such that y is satisfiable if and only if all x_i are satisfiable. The "AND" in the name stems from the fact that the predicate "y is satisfiable" can be written as the AND of all predicates "x_i is satisfiable". Drucker's result complements the result by Bodlaender, Downey, Fellows, *et al.* [2] and Fortnow and Santhanam [3], who proved the analogous statement for OR-compressions, and Drucker's proof not only subsumes that result but also extends it to *randomized* compression algorithms that are allowed to have a certain probability of failure.

The overall structure of our proof is similar to the arguments of Ko [4] for P-selective sets, which use the fact that tournaments have dominating sets of logarithmic size. We generalize this fact to hypergraph tournaments. For the information-theoretic part of the proof, we consider a natural generalization of the average noise sensitivity of a Boolean function, which is bounded for compressive maps. We prove this with mechanical calculations that involve the Kullback–Leibler divergence.

1 Introduction

The influential "OR-conjecture" by Bodlaender, Downey, Fellows, *et al.* [2] asserts that t instances $x_1, \dots, x_t$ of SAT cannot be mapped in polynomial time to an instance y of size $\mathsf{poly}(\max_i |x_i|)$ so that y is a yes-instance if and only if at least one x_i is a yes-instances. Conditioned on the OR-conjecture, the "composition framework" of Bodlaender, Downey, Fellows, *et al.* [2] has been used to show that many different problems

*This is a draft. Any comments, even minor ones, are welcome: `2014@holgerdell.com`.

ISSN 1433-8092

in parameterized complexity do not have polynomial kernels. Fortnow and Santhanam [3] were able to prove that the OR-conjecture holds unless $\mathsf{coNP} \subseteq \mathsf{NP/poly}$, thereby connecting the OR-conjecture with an existing hypothesis in complexity theory.

The results of [2, 3] can be used not only to rule out deterministic kernelization algorithms, but also to rule out randomized kernelization algorithms with one-sided error, as long as the success probability is bigger than zero; this is the same as allowing the kernelization algorithm to be a coNP-algorithm. Left open was the question whether the complexity-theoretic hypothesis $\mathsf{coNP} \not\subseteq \mathsf{NP/poly}$ (or some other hypothesis believed by complexity theorists) suffices to rule out kernelization algorithms that are randomized and have two-sided error. Drucker [1] resolved this question affirmatively; his results can rule out kernelization algorithms that have a constant gap in their error probabilities. This result indicates that randomness does not help to decrease the size of kernels significantly.

With the same proof, Drucker [1] resolves another, perhaps more important question: the question whether the "AND-conjecture", which has also been formulated by Bodlaender, Downey, Fellows, *et al.* [2] analogous to the OR-conjecture, can be derived from existing complexity-theoretic assumptions. This is an intriguing question by itself, and it is also relevant for parameterized complexity as, for some parameterized problems, we can rule out polynomial kernels under the AND-conjecture, but we do not know how to do so under the OR-conjecture. Drucker [1] proved that the AND-conjecture is true if $\mathsf{coNP} \not\subseteq \mathsf{NP/poly}$ holds. This paper contains a new, comparatively simple proof of Drucker's theorem.

1.1 Main Theorem: Ruling out OR- and AND-compressions

Since "AND-compression of L" is the same as "OR-compression of $\overline{L}$", and since the complexity consequence that we obtain for L is actually closed under complementation, we will only consider OR-compressions. The following theorem states Drucker's result in a robust form.

Theorem 1.1 (Robust version of Drucker's theorem). *Let $L, L' \subseteq \{0,1\}^*$ be languages, let $e_s, e_c \in [0,1]$ be error probabilities with $e_s + e_c < 1$, and let $\epsilon > 0$. Assume that there exists a randomized polynomial-time algorithm A that maps any set $x = \{x_1, \dots, x_t\} \subseteq \{0,1\}^n$ for some n and t to $y = A(x)$ such that:*

- *(Soundness) If all x_i's are no-instances of L, then y is a no-instance of L' with probability $\geqslant 1 - e_s$.*
- *(Completeness) If exactly one x_i is a yes-instance of L, then y is a yes-instance of L' with probability $\geqslant 1 - e_c$.*
- *(Size bound) The size of y is bounded by $t^{1-\epsilon} \cdot \mathsf{poly}(n)$.*

Then $L \in \mathsf{NP/poly} \cap \mathsf{coNP/poly}$.

The procedure A above is not a "full" OR-compression for two reasons: Firstly, it only needs to work, or be analyzed, in the case that all input instances have the same length;

this is useful in hardness of kernelization proofs as it allows to group similar instances together. Secondly, it only needs to work, or be analyzed, in the case that at most one of the input instances is a yes-instance of L. That this milder completeness requirement is sufficient is easy to see in our simplified proof. The theorem of Fortnow and Santhanam [3] corresponds to the special case of the above theorem in which $e_c = 0$. However, they only obtain the consequence $L \in \mathsf{coNP/poly}$, which prevents their result from applying to AND-compressions in a non-trivial way. Moreover, their proof uses the full completeness requirement, and the milder completeness condition above does not seem to suffice.

The proof of Theorem 1.1 appears in §3 as a special case of Theorem 3.1, which allows for finer control over the output size bound; in particular, it allows for the output size to be up to ϵt for any fixed polynomial $t = t(n)$, as long as $\epsilon > 0$ is small enough compared to the error probability. The proof in §3 does not become more complicated by doing this. In §4, we present a *slightly* more complicated proof that allows for the output size to be up to $O(t \log t)$ in the zero-error case $e_s = e_c = 0$; this culminates in Theorem 4.1. All of these results –perhaps with the exception of the milder requirement in the completeness case– already appear in Drucker [1].

1.2 Overview and comparison of the proof

Both Drucker's proof and the proof presented here use the OR-compression A to construct a $\mathsf{P/poly}$-reduction from L to the *statistical distance problem*, which is known to be in the intersection of $\mathsf{NP/poly}$ and $\mathsf{coNP/poly}$ by previous work.

To construct the polynomial advice of the reduction, Drucker uses the minimax theorem and a game-theoretic sparsification argument. We avoid these arguments and construct the advice in a combinatorial way more directly: We generalize the fact that tournaments have dominating sets of logarithmic size to *hypergraph tournaments*; these are complete t-uniform hypergraphs with the additional property that, for each hyperedge, one of its elements gets "selected". In particular, for each set $e \subseteq \overline{L}$ of t no-instances, we will select one element of e based on the fact that A's behavior on e somehow proves that the selected instance is a no-instance of L. The advice of the reduction is going to be a small dominating set of this hypergraph tournament on the set of no-instances of L. The crux of the reduction is that we can efficiently test, with the help of the statistical distance problem oracle, whether an instance is dominated or not. Since any instance is dominated if and only if it is a no-instance of L, this suffices to solve L.

The basic argument above can be seen as a generalization of Ko's argument for P-selective sets [4]. The arguments used by Fortnow and Santhanam [3] and Dell and Van Melkebeek [5] for compression-type procedures and Dell, Kabanets, Melkebeek, *et al.* [6] for isolation procedures have a similar flavor.

In the information-theoretic part of the proof, we consider a natural generalization of the notion of average noise sensitivity of a Boolean function to non-Boolean functions, and we show that compressive maps have small average noise sensitivity. This is similar to Drucker's "distributional stability". Our proof consists of a mechanical calculation that bounds the statistical distance in terms of the Kullback–Leibler divergence using Pinsker's inequality, and then relates it to the mutual information.

1.3 Application: The composition framework for ruling out $O(k^{d-\epsilon})$ kernels

We briefly describe a modern variant of the composition framework that is sufficient to rule out kernels of size $O(k^{d-\epsilon})$ using Theorem 1.1. It is almost identical to Lemma 1 of [5, 7] and Definition 2.2 of [8]. By applying the framework for unbounded d, we can also use it to rule out polynomial kernels.

Definition 1. Let L be a language, and let Π with parameter k be a parameterized problem. A *d-partite composition of L into* Π is a polynomial-time algorithm A that maps any set $x = \{x_1, \ldots, x_t\} \subseteq \{0,1\}^n$ for some n and t to $y = A(x)$ such that:

(1) If all x_i's are no-instances of L, then y is a no-instance of Π.

(2) If exactly one x_i is a yes-instance of L, then y is a yes-instance of Π.

(3) The parameter k of y is bounded by $t^{1/d+o(1)} \cdot \mathsf{poly}(n)$.

This notion of composition has one crucial advantage over previous notions of OR-composition: The algorithm A does not need to work, or be analyzed, in the case that two or more of the x_i's are yes-instances.

Definition 2. Let Π be a parameterized problem. We call Π *d-compositional* if there exists an NP-hard or coNP-hard problem L that has a d-partite composition algorithm into Π.

The above definition encompasses both "AND-compositional" and "OR-compositional" because an OR-composition of L into Π is the same as an AND-composition of $\overline{L}$ into Π. We have the following corollary of Drucker's theorem.

Corollary 1.2. *If* $\mathsf{coNP} \not\subseteq \mathsf{NP/poly}$*, then no d-compositional problem has kernels of size $O(k^{d-\epsilon})$. Moreover, this even holds when the kernelization algorithm is allowed to be a randomized algorithm with at least a constant gap in error probability.*

Proof. Let L be an NP-hard or coNP-hard problem that has a d-partite composition A' into Π. Assume for contradiction that Π has a kernelization algorithm with soundness error at most e_s and completeness error at most e_c so that e_s+e_c is bounded by a constant smaller than one. The concatenation of A' with the assumed $O(k^{d-\epsilon'})$-kernelization gives rise to an algorithm A that satisfies the conditions of Theorem 1.1, for example with $\epsilon = \epsilon'/(2d)$. Therefore, we get $L \in (\mathsf{coNP/poly} \cap \mathsf{NP/poly})$ and thus $\mathsf{coNP} \subseteq \mathsf{NP/poly}$, a contradiction. ∎

Several variants of the framework provided by this corollary are possible:

1. In order to rule out $\mathsf{poly}(k)$-kernels for a parameterized problem Π, we just need to prove that Π is d-compositional for all $d \in \mathbb{N}$; let's call Π *compositional* in this case. One way to show that Π is compositional is to construct a single *composition* from a hard problem L into Π; this is an algorithm as in Definition 1, except that we replace (3) with the bound $k \leqslant t^{o(1)}\mathsf{poly}(n)$.

2. Since all x_i's in Definition 1 are promised to have the same length, we can consider a padded version $\tilde{L}$ of the language L in order to filter the input instances of length n of the original L into a polynomial number of equivalence classes. Each input length of $\tilde{L}$ in some interval $[p_1(n), p_2(n)]$ corresponds to one equivalence class of length-n instances of L. So long as $\tilde{L}$ remains NP-hard or coNP-hard, it is sufficient to consider a composition from $\tilde{L}$ into Π. This has been formalized in Definition 4 of [9].

3. The composition algorithm can also use randomness, as long as the overall probability gap of the concatenation of composition and kernelization is not negligible.

4. In the case that L is NP-hard, the composition algorithm can also be a coNP-algorithm or an oracle communication game in order to get the collapse [3, 5]. This, however, does not seem to follow from Drucker's proof and it seems to require the full completeness condition for the OR-composition. This variant of the framework has been exploited in [10, 11].

2 Preliminaries

For any set $R \subseteq \{0,1\}^*$ and any $\ell \in \mathbb{N}$, we write $R_\ell \doteq R \cap \{0,1\}^\ell$ for the set of all length-ℓ strings inside of R. For any $t \in \mathbb{N}$, we write $[t] \doteq \{1, \ldots, t\}$. For a set V, we write $\binom{V}{\leqslant t}$ for the set of all subsets $x \subseteq V$ that have size at most t. We will work over a finite alphabet, usually $\Sigma = \{0,1\}$. For a vector $a \in \Sigma^t$, a number $j \in [t]$, and a value $y \in \Sigma$, we write $a|_{j \leftarrow y}$ for the string that coincides with a except in position j, where it has value y. For background in complexity theory, we defer to the book by Arora and Barak [12]. We assume some familiarity with the complexity classes NP and coNP as well as their non-uniform versions NP/poly and coNP/poly.

2.1 Distributions and Randomized Mappings

A *distribution* on a finite ground set Ω is a function $\mathcal{D} : \Omega \to [0,1]$ with $\sum_{\omega \in \Omega} \mathcal{D}(\omega) = 1$. The *support* of $\mathcal{D}$ is the set $\operatorname{supp} \mathcal{D} = \{\omega \in \Omega : \mathcal{D}(\omega) > 0\}$. The *uniform distribution* $\mathcal{U}_\Omega$ on Ω is the distribution with $\mathcal{U}_\Omega(\omega) = \frac{1}{|\Omega|}$ for all $\omega \in \Omega$. We often view distributions as *random variables*, that is, we may write $f(\mathcal{D})$ to denote the distribution $\mathcal{D}'$ that first produces a sample $\omega \sim \mathcal{D}$ and then outputs $f(\omega)$, where $f : \Omega \to \Omega'$. We use any of the following notations:

$$\mathcal{D}'(\omega') = \Pr(f(\mathcal{D}) = \omega') = \Pr_{\omega \sim \mathcal{D}}(f(\omega) = \omega') = \sum_{\omega \in \Omega} \mathcal{D}(\omega) \cdot \Pr(f(\omega) = \omega') .$$

The last term $\Pr(f(\omega) = \omega')$ in this equation is either 0 or 1 if f is a deterministic function, but we will also allow f to be a *randomized mapping*, that is, f has access to some "internal" randomness. This is modeled as a function $f : \Omega \times \{0,1\}^r \to \Omega'$ for some $r \in \mathbb{N}$, and we write $f(\mathcal{D})$ as a short-hand for $f(\mathcal{D}, \mathcal{U}_{\{0,1\}^r})$. That is, the internal randomness consists of a sequence of independent and fair coin flips.

2.2 Statistical Distance

The *statistical distance* $d(X,Y)$ between two distributions X and Y on Ω is defined as

$$d(X,Y) = \max_{T \subseteq \Omega} \big| \Pr(X \in T) - \Pr(Y \in T) \big| \,. \tag{1}$$

The statistical distance between X and Y is a number in $[0,1]$, with $d(X,Y)=0$ if and only if $X=Y$ and $d(X,Y)=1$ if and only if the support of X is disjoint from the support of Y. It is an exercise to show the standard equivalence between the statistical distance and the 1-norm:

$$d(X,Y) = \frac{1}{2} \cdot \left\| X - Y \right\|_1 = \frac{1}{2} \sum_{\omega \in \Omega} \big| \Pr(X = \omega) - \Pr(Y = \omega) \big| \,.$$

2.3 The Statistical Distance Problem

For $\mathcal{U} = \mathcal{U}_{\{0,1\}^n}$ and $0 \leqslant \delta < \Delta \leqslant 1$, let $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ be the following promise problem:

yes-instances: Pairs of circuits $C, C' : \{0,1\}^n \to \{0,1\}^*$ so that $d(C(\mathcal{U}), C'(\mathcal{U})) \geqslant \Delta$.

no-instances: Pairs of circuits $C, C' : \{0,1\}^n \to \{0,1\}^*$ so that $d(C(\mathcal{U}), C'(\mathcal{U})) \leqslant \delta$.

The statistical distance problem is not known to be polynomial-time computable, and in fact it is not believed to be. On the other hand, the problem is also not believed to be NP-hard because the problem is computationally easy in the following sense.

Theorem 2.1 ([13] + [14]).
If $\delta < \Delta$ are constants, we have $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta} \in \Big(\mathsf{NP/poly} \,\cap\, \mathsf{coNP/poly} \Big)$.
Moreover, the same holds when $\delta = \delta(n)$ and $\Delta = \Delta(n)$ are functions of the input length that satisfy $\Delta - \delta \geqslant \frac{1}{\mathsf{poly}(n)}$.

This is the only fact about the SD-problem that we will use in this paper.

Slightly stronger versions of this theorem are known: For example, Xiao [13, p. 144ff] proves that $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta} \in \mathsf{AM} \cap \mathsf{coAM}$ holds. In fact, Theorem 2.1 is established by combining his theorem with the standard fact that $\mathsf{AM} \subseteq \mathsf{NP/poly}$, i.e., that Arthur–Merlin games can be derandomized with polynomial advice [14]. Moreover, when we have the stronger guarantee that $\Delta^2 > \delta$ holds, then $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ can be solved using *statistical zero-knowledge proof systems* [15, 16]. Finally, if $\Delta = 1$, the problem can be solved with *perfect* zero-knowledge proof systems [15, Proposition 5.7]. Using these stronger results whenever possible gives slightly stronger complexity collapses in the main theorem.

3 Ruling out OR-compressions

In this section we prove that any language L that has an OR-compression A into a language L' is in $\mathsf{coNP/poly} \cap \mathsf{NP/poly}$, provided some conditions on the error probabilities and the compression ratio are satisfied.

Theorem 3.1 (ϵt-compressive version of Drucker's theorem). *Let $L, L' \subseteq \{0,1\}^*$ be languages and $e_s, e_c \in [0,1]$ be some constants denoting the error probabilities. Let $t = t(n) > 0$ be a polynomial and $\epsilon > 0$. Let*

$$A : \binom{\{0,1\}^n}{\leqslant t} \to \{0,1\}^{\epsilon t} \tag{2}$$

be a randomized $\mathsf{P/poly}$-algorithm such that, for all $x \in \binom{\{0,1\}^n}{\leqslant t}$,

- *if $|x \cap L| = 0$, then $A(x) \in \overline{L'}$ holds with probability $\geqslant 1 - e_s$, and*
- *if $|x \cap L| = 1$, then $A(x) \in L'$ holds with probability $\geqslant 1 - e_c$.*

If $e_s + e_c < 1 - \sqrt{(2\ln 2)\epsilon}$, then $L \in \mathsf{NP/poly} \cap \mathsf{coNP/poly}$.

We optimized our proof for simplicity and not for the optimality of the conditions on the parameters e_s, e_c, and ϵ. Drucker's original bound is $e_s + e_c < 1 - \sqrt{(\ln 2/2)\epsilon}$. Using the slightly more complicated setup of §4, we would be able to achieve this bound. However, to obtain the robust version of Drucker's theorem, Theorem 1.1, as a corollary, these differences do not matter; we now fill in the technicalities for obtaining Theorem 1.1 from Theorem 3.1 before we prove the latter.

Proof (of Theorem 1.1). Let A be the algorithm assumed in Theorem 1.1, and let $C \geqslant 2$ be large enough so that the output size of A is bounded by $t^{1-1/C} \cdot C \cdot n^C$. We transform A into an algorithm as required for Theorem 3.1. Let $\epsilon > 0$ be a small enough constant so that $e_s + e_c < 1 - \sqrt{(2\ln 2)\epsilon}$. Moreover, let $t(n)$ be a large enough polynomial so that $(t(n))^{1-1/C} \cdot C \cdot n^C < \epsilon t(n)$ holds. Then we restrict A to a family of functions $A_n : \binom{\{0,1\}^n}{\leqslant t(n)} \to \{0,1\}^{<\epsilon t(n)}$. Now a minor observations is needed to get an algorithm of the form (2): The set $\{0,1\}^{<\epsilon t}$ can be efficiently encoded in $\{0,1\}^{\epsilon t}$ (which changes the output language from L' to some L''). Thus we constructed a family A_n as required by Theorem 3.1, which proves the claim. ∎

3.1 ORs are sensitive to Yes-instances

The first basic fact about OR-compressions is that they are "L-sensitive": They show a dramatically different behavior for all-no input sets vs. input sets that contain a single yes-instance of L. The following simple fact is the only place where we use the semantic properties of A in the overall proof.

Lemma 3.2. *For all distributions X on $\binom{\overline{L}}{<t}$ and all $v \in L$, we have*

$$d\Big(A(X)\ ,\ A(X \cup \{v\})\Big) \geqslant \Delta \doteq 1 - (e_s + e_c)\,. \tag{3}$$

Proof. The probability that $A(X)$ outputs an element of L' is at most e_s, and similarly, the probability that $A(X \cup \{v\})$ outputs an element of L' is at least $1 - e_c$. By (1) with $T = L'$, the statistical distance between the two distributions is at least Δ. ∎

Despite the fact that OR-compressions are sensitive to the presence or absence of a yes-instance, we argue next that their behavior *within* the set of no-instances is actually quite predictable.

3.2 The average noise sensitivity of compressive maps is small

OR-compressions are in particular compressive maps. The following lemma says that the average noise sensitivity of any compressive map is low. Here, "average noise sensitivity" refers to the difference in the behavior of a function when the input is subject to random noise; in our case, we change the input in a single random location and notice that the behavior of a compressive map does not change much.

Lemma 3.3. *Let $t \in \mathbb{N}$, let X be the uniform distribution on $\{0,1\}^t$, and let $\epsilon > 0$. Then, for all randomized mappings $f : \{0,1\}^t \to \{0,1\}^{\epsilon t}$, we have*

$$\mathop{\mathbf{E}}_{j \sim \mathcal{U}_{[t]}} \quad d\Big(f(X|_{j\leftarrow 0})\ ,\ f(X|_{j\leftarrow 1})\Big) \quad \leqslant \delta \doteq \sqrt{2\ln 2 \cdot \epsilon}\,. \tag{4}$$

We defer the purely information-theoretic and mechanical proof of this lemma to §3.5. We remark that, in the special case where $f : \{0,1\}^t \to \{0,1\}$ is a Boolean function, the left-hand side of (4) coincides with the usual definition of the average noise sensitivity.

We translate Lemma 3.3 to our OR-compression A as follows.

Lemma 3.4. *Let $A : \binom{\{0,1\}^n}{\leqslant t} \to \{0,1\}^{\epsilon t}$. For all $e \in \binom{\{0,1\}^n}{t}$, there exists $v \in e$ so that*

$$d\Big(A(\mathcal{U}_{2^e} \setminus \{v\})\ ,\ A(\mathcal{U}_{2^e} \cup \{v\})\Big) \leqslant \delta\,. \tag{5}$$

Here $\mathcal{U}_{2^e}$ samples a subsets of e uniformly at random. Note that we replaced the expectation over j from (4) with the mere existence of an element v in (5) since this is all we need; the stronger property also holds.

Proof. To prove the claim, let $v_1, \dots, v_t$ be the elements of e in lexicographic order. For $b \in \{0,1\}^t$, let $g(b) \subseteq e$ be such that $v_i \in g$ holds if and only if $b_i = 1$. We define the randomized mapping $f : \{0,1\}^t \to \{0,1\}^{\epsilon t}$ as follows:

$$f(b_1, \dots, b_t) \doteq A\Big(g(b)\Big)\,.$$

Then $f(X|_{j\leftarrow 0}) = A(\mathcal{U}_{2^e} \setminus \{v_j\})$ and $f(X|_{j\leftarrow 1}) = A(\mathcal{U}_{2^e} \cup \{v_j\})$. The claim follows from Lemma 3.3 with $v \doteq v_j$ for some j that minimizes the statistical distance in (4). ∎

The lemma suggest the following tournament idea. We let $V = \overline{L}_n$ be the set of no-instances, and we let them compete in matches consisting of t players each. That is, a match corresponds to a hyperedge $e \in \binom{V}{t}$ of size t and every such hyperedge is present, so we are looking at a complete t-uniform hypergraph. We say that a player $v \in e$ is "selected" in the hyperedge e if the behavior of A on $\mathcal{U}_{2^e} \setminus \{v\}$ is not very different from the behavior of A on $\mathcal{U}_{2^e} \cup \{v\}$, that is, if (5) holds. The point of this construction is that v being selected proves that v must be a no-instance because (3) does not hold. We obtain a "selector" function $S : \binom{V}{t} \to V$ that, given e, selects an element $v = S(e) \in e$. We call S a *hypergraph tournament* on V.

3.3 Hypergraph tournaments have small dominating sets

Tournaments are complete directed graphs, and it is well-known that they have dominating sets of logarithmic size. A straightforward generalization applies to hypergraph tournaments $S : \binom{V}{t} \to V$. We say that a set $g \in \binom{V}{t-1}$ *dominates* a vertex v if $v \in g$ or $S(g \cup \{v\}) = v$ holds. A set $\mathcal{D} \subseteq \binom{V}{t-1}$ is a *dominating set* of S if all vertices $v \in V$ are dominated by at least one element in $\mathcal{D}$.

Lemma 3.5. *Let V be a finite set, and let $S : \binom{V}{t} \to V$ be a hypergraph tournament. Then S has a dominating set $\mathcal{D} \subseteq \binom{V}{t-1}$ of size at most $t \log |V|$.*

Proof. We construct the set $\mathcal{D}$ inductively. Initially, it has $k = 0$ elements. After the k-th step of the construction, we will preserve the invariant that $\mathcal{D}$ is of size exactly k and that $|R| \leqslant (1 - 1/t)^k \cdot |V|$ holds, where R is the set of vertices that are not yet dominated, that is,

$$R = \Big\{ v \in V : \ v \notin g \text{ and } S(g \cup \{v\}) \neq v \text{ holds for all } g \in \mathcal{D} \Big\} .$$

If $0 < |R| < t$, we can add an arbitrary edge $g^* \in \binom{V}{t-1}$ with $R \subseteq g^*$ to $\mathcal{D}$ to finish the construction. Otherwise, the following averaging argument, shows that there is an element $g^* \in \binom{R}{t-1}$ that dominates at least a $1/t$-fraction of elements $v \in R$:

$$\frac{1}{t} = \mathop{\mathbf{E}}_{e \in \binom{R}{t}} \Pr_{v \in e} \Big(S(e) = v \Big) = \mathop{\mathbf{E}}_{g \in \binom{R}{t-1}} \Pr_{v \in R - g} \Big(S(g \cup \{v\}) = v \Big) .$$

Thus, the number of elements of R left undominated by g^* is at most $(1 - 1/t) \cdot |R|$, so the inductive invariant holds. Since $(1-1/t)^k \cdot |V| \leqslant \exp(-k/t) \cdot |V| < 1$ for $k = t \log |V|$, we have $R = \varnothing$ after $k \leqslant t \log |V|$ steps of the construction, and in particular, $\mathcal{D}$ has at most $t \log |V|$ elements. ∎

3.4 Proof of the main theorem: Reduction to statistical distance

Proof (of Theorem 3.1). We describe a deterministic $\mathsf{P/poly}$ reduction from L to the statistical distance problem $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ with $\Delta = 1 - (e_s + e_c)$ and $\delta = \sqrt{(2 \ln 2)\epsilon}$. The reduction outputs the conjunction of polynomially many instances of $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$. Since $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ is contained in the intersection of $\mathsf{NP/poly}$ and $\mathsf{coNP/poly}$ by Theorem 2.1, and since this intersection is closed under taking polynomial conjunctions, we obtain $L \in \mathsf{NP/poly} \cap \mathsf{coNP/poly}$. Thus it remains to find such a reduction. To simplify the discussion, we describe the reduction in terms of an algorithm that solves L and uses $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ as an oracle. However, the algorithm only makes non-adaptive queries at the end of the computation and accepts if and only if all oracle queries accept; this corresponds to a reduction that maps an instance of L to a conjunction of instances of $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ as required.

To construct the advice at input length n, we use Lemma 3.4 with $t = t(n)$ to obtain a hypergraph tournament S on $V = \overline{L}_n$, which in turn gives rise to a small dominating set $\mathcal{D} \subseteq \binom{V}{t-1}$ by Lemma 3.5. We remark the triviality that if $|V| \leqslant t = \mathsf{poly}(n)$, then we can use V, the set of all no-instances of L at this input length, as the advice. Otherwise, we define the hypergraph tournament S for all $e \in \binom{V}{t}$ as follows:

$$S(e) \doteq \min \Big\{ v \in e \;\Big|\; d\Big(A(\mathcal{U}_{2^e} \setminus \{v\}) \;,\; A(\mathcal{U}_{2^e} \cup \{v\})\Big) \leqslant \delta \Big\} \;.$$

By Lemma 3.4, the set over which the minimum is taken is non-empty, and thus S is well-defined. Furthermore, the hypergraph tournament has a dominating set $\mathcal{D}$ of size at most tn by Lemma 3.5. As advice for input length n, we choose this set $\mathcal{D}$. Now we have $v \in \overline{L}$ if and only if v is dominated by $\mathcal{D}$. The idea of the reduction is to efficiently check the latter property.

The algorithm works as follows: Let $v \in \{0,1\}^n$ be an instance of L given as input. If $v \in g$ holds for some $g \in \mathcal{D}$, the algorithm rejects v and halts. Otherwise, it queries the SD-oracle on the instance $(A(\mathcal{U}_{2^g}), A(\mathcal{U}_{2^g} \cup \{v\}))$ for each $g \in \mathcal{D}$. If the oracle claims that all queries are yes-instances, our algorithm accepts, and otherwise, it rejects.

First note that distributions of the form $A(\mathcal{U}_{2^g})$ and $A(\mathcal{U}_{2^g} \cup \{v\})$ can be be sampled by using polynomial-size circuits, and so they form syntactically correct instances of the SD-problem: The information about A, g, and v is hard-wired into these circuits, the input bits of the circuits are used to produce a sample from $\mathcal{U}_{2^g}$, and they serve as internal randomness of A in case A is a randomized algorithm.

It remains to prove the correctness of the reduction. If $v \in L$, we have for all $g \in \mathcal{D} \subseteq \overline{L}$ that $v \notin g$ and that the statistical distance of the query corresponding to g is at least $\Delta = c - s$ by Lemma 3.2. Thus all queries that the reduction makes satisfy the promise of the SD-problem and the oracle answers the queries correctly, leading our reduction to accept. On the other hand, if $v \notin L$, then, since $\mathcal{D}$ is a dominating set of $\overline{L}$ with respect to the hypergraph tournament S, there is at least one $g \in \mathcal{D}$ so that $v \in g$ or $S(g \cup \{v\}) = v$ holds. If $v \in g$, the reduction rejects. The other case implies that the statistical distance between $A(\mathcal{U}_{2^g})$ and $A(\mathcal{U}_{2^g} \cup \{v\})$ is at most δ. The query corresponding to this particular g therefore satisfies the promise of the SD-problem, which means that the oracle answers correctly on this query and our reduction rejects. ■

3.5 Information-theoretic arguments

We now prove Lemma 3.3. The proof uses the *Kullback–Leibler divergence* as an intermediate step. Just like the statistical distance, this notion measures how similar two distributions are, but it does so in an information-theoretic way rather than in a purely statistical way. In fact, it is well-known in the area that the Kullback–Leibler divergence and the mutual information are almost interchangeable in a certain sense. We prove a version of this paradigm formally in Lemma 3.6 below; then we prove Lemma 3.3 by bounding the statistical distance in terms of the Kullback–Leibler divergence using standard inequalities.

We introduce some basic information-theoretic notions. The *Shannon entropy* $H(X)$ of a random variable X is

$$H(X) = \mathop{\mathbf{E}}_{x \sim X} \log \left(\frac{1}{\Pr(X = x)} \right) \,.$$

The conditional Shannon entropy $H(X|Y)$ is

$$\begin{aligned} H(X|Y) &= \mathop{\mathbf{E}}_{y\sim Y} H(X|Y=y) \\ &= \mathop{\mathbf{E}}_{y\sim Y} \sum_x \Pr(X=x|Y=y)\cdot \log\left(\frac{1}{\Pr(X=x|Y=y)}\right). \end{aligned}$$

The *mutual information* between X and Y is $I(X:Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$. Note that $I(X:Y) \leqslant \log|\operatorname{supp} X|$, where $|\operatorname{supp} X|$ is the size of the support of X. The *conditional mutual information* can be defined by the *chain rule of mutual information* $I(X:Y \mid Z) = I(X:YZ) - I(X:Z)$. If Y and Z are independent, then a simple calculation reveals that $I(X:Y) \leqslant I(X:Y\mid Z)$ holds.

We now establish a bound on the Kullback–Leibler divergence. The application of Lemma 3.3 only uses $\Sigma = \{0,1\}$. The proof does not become more complicated for general Σ, and we will need the more general version later in this paper.

Lemma 3.6. *Let $t \in \mathbb{N}$, let $\epsilon > 0$, let $X_1,\ldots,X_t$ be independent distributions on some finite set Σ, and let $X = X_1,\ldots,X_t$. Then, for all randomized mappings $f : \Sigma^t \to \{0,1\}^*$ with $I\big(f(X):X\big) \leqslant \epsilon t$, we have the following upper bound on the expected value of the Kullback–Leibler divergence:*

$$\mathop{\mathbf{E}}_{j\sim\mathcal{U}_{[t]}} \mathop{\mathbf{E}}_{x\sim X_j} D_{\mathrm{KL}}\Big(f(X) \mid\mid f(X|_{j\leftarrow x})\Big) \leqslant \epsilon\,.$$

Proof. The result follows by a basic calculation with entropy notions. The first equality is the definition of the Kullback–Leibler divergence, which we rewrite using the logarithm rule $\log(a/b) = \log(1/b) - \log(1/a)$ and the linearity of expectation:

$$\begin{aligned} &\mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} D_{\mathrm{KL}}\Big(f(X) \mid\mid f(X|_{j\leftarrow x})\Big) \\ &= \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x}\sum_z \log\left(\frac{\Pr(f(X|_{j\leftarrow x})=z)}{\Pr(f(X)=z)}\right)\cdot \Pr\Big(f(X|_{j\leftarrow x})=z\Big) \\ &= \mathop{\mathbf{E}}_{j}\sum_z \log\left(\frac{1}{\Pr(f(X)=z)}\right)\cdot \mathop{\mathbf{E}}_{x}\Pr\Big(f(X|_{j\leftarrow x})=z\Big) \\ &\quad - \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x}\sum_z \log\left(\frac{1}{\Pr(f(X|_{j\leftarrow x})=z)}\right)\cdot \Pr\Big(f(X|_{j\leftarrow x})=z\Big). \end{aligned}$$

As $\mathbf{E}_x \Pr\left(f(X|_{j\leftarrow x})=z\right) = \Pr(f(X)=z)$, both terms of the sum above are entropies, and we can continue the calculation as follows:

$$
\begin{aligned}
\ldots &= H(f(X)) - \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} H(f(X)|X_j = x) && \text{(definition of entropy)}\\
&= H(f(X)) - \mathop{\mathbf{E}}_{j} H(f(X)|X_j) && \text{(definition of conditional entropy)}\\
&= \mathop{\mathbf{E}}_{j} I(f(X) : X_j) && \text{(definition of mutual information)}\\
&\leqslant \frac{1}{t} \cdot \sum_{j\in[t]} I\Big(f(X) : X_j \mid X_1 \ldots X_{j-1}\Big) && \text{(by independence of } X_j\text{'s)}\\
&= \frac{1}{t} \cdot I(f(X) : X) \leqslant \epsilon\,. && \text{(chain rule of mutual information)} \quad \blacksquare
\end{aligned}
$$

We now turn to the proof of Lemma 3.3, where we bound the statistical distance in terms of the Kullback–Leibler divergence.

Proof (of Lemma 3.3). We observe that $I(f(X) : X) \leqslant \log|\operatorname{supp} f(X)| \leqslant \epsilon t$, and so we are in the situation of Lemma 3.6 with $\Sigma = \{0, 1\}$. We first apply the triangle inequality to the left-hand side of (4). Then we use Pinsker's inequality [17, Lemma 11.6.1] to bound the statistical distance in terms of the Kullback–Leibler divergence, which we can in turn bound by ϵ using Lemma 3.6.

$$
\begin{aligned}
&\mathop{\mathbf{E}}_{j\sim\mathcal{U}_{[t]}} d\Big(f(X|_{j\leftarrow 0})\ ,\ f(X|_{j\leftarrow 1})\Big)\\
&\leqslant \mathop{\mathbf{E}}_{j\sim\mathcal{U}_{[t]}} d\Big(f(X)\ ,\ f(X|_{j\leftarrow 0})\Big) && \text{(triangle inequality)}\\
&+ \mathop{\mathbf{E}}_{j\sim\mathcal{U}_{[t]}} d\Big(f(X)\ ,\ f(X|_{j\leftarrow 1})\Big)\\
&= 2 \cdot \mathop{\mathbf{E}}_{j\sim\mathcal{U}_{[t]}} \mathop{\mathbf{E}}_{x\sim X_j} d\Big(f(X)\ ,\ f(X|_{j\leftarrow x})\Big)\\
&\leqslant 2 \cdot \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} \sqrt{\frac{\ln 2}{2} \cdot D_{\mathrm{KL}}\Big(f(X) \mid\mid f(X|_{j\leftarrow x})\Big)} && \text{(Pinsker's inequality)}\\
&\leqslant 2 \cdot \sqrt{\frac{\ln 2}{2} \cdot \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} D_{\mathrm{KL}}\Big(f(X) \mid\mid f(X|_{j\leftarrow x})\Big)} && \text{(Jensen's inequality)}\\
&\leqslant 2 \cdot \sqrt{\frac{\ln 2}{2} \cdot \epsilon} = \delta\,. && \text{(Lemma 3.6)}
\end{aligned}
$$

The equality above uses the fact that X_j is the uniform distribution on $\{0, 1\}$. ■

4 Extension: Ruling out OR-compressions of size $O(t \log t)$

In this section we tweak the proof of Theorem 3.1 so that it works even when the t instances of L are mapped to an instance of L' of size at most $O(t \log t)$. The drawback is that we cannot handle positive constant error probabilities for randomized

OR-compression anymore. For simplicity, we restrict ourselves to *deterministic* OR-compressions of size $O(t \log t)$ throughout this section.

Theorem 4.1 ($O(t \log t)$-compressive version of Drucker's theorem). *Let $L, L' \subseteq \{0,1\}^*$ be languages. Let $t = t(n) > 0$ be a polynomial. Assume there exists a $\mathsf{P/poly}$-algorithm*

$$A : \binom{\{0,1\}^n}{\leqslant t} \to \{0,1\}^{O(t \log t)}$$

such that, for all $x \in \binom{\{0,1\}^n}{\leqslant t}$,

- *if $|x \cap L| = 0$, then $A(x) \in \overline{L'}$, and*
- *if $|x \cap L| = 1$, then $A(x) \in L'$.*

Then $L \in \mathsf{NP/poly} \cap \mathsf{coNP/poly}$.

The main reason why the proof in §3 breaks down for compressions to size ϵt with $\epsilon = O(\log t)$ is that the bound on the statistical distance in Lemma 3.3 becomes trivial. This happens already when $\epsilon \geqslant \frac{1}{2\ln 2} \approx 0.72$. On the other hand, the bound that Lemma 3.6 gives for the Kullback–Leibler divergence remains non-trivial even for $\epsilon = O(\log t)$. To see this, note that the largest possible divergence between $f(X)$ and $f(X|_{j \leftarrow x})$, that is, the divergence without the condition on the mutual information between $f(X)$ and X, is $t \cdot \log |\Sigma|$, and the bound that Lemma 3.6 yields for $\epsilon = O(\log t)$ is logarithmic in that.

Inspecting the proof of Lemma 3.3, we realize that the loss in meaningfulness stems from Pinsker's inequality, which becomes trivial in the parameter range under consideration. Luckily, there is a different inequality between the statistical distance and the Kullback–Leibler divergence, Vajda's inequality, that still gives a non-trivial bound on the statistical distance when the divergence is $\geqslant \frac{1}{2\ln 2}$. The inequality works out such that if the divergence is logarithmic, then the statistical distance is an inverse polynomial away from 1. We obtain the following analogue to Lemma 3.3.

Lemma 4.2. *Let $t \in \mathbb{N}$ let $X_1, \ldots, X_t$ be independent uniform distributions on some finite set Σ, and write $X = X_1, \ldots, X_t$. Then, for all randomized mappings $f : \Sigma^t \to \{0,1\}^*$ with $I(f(X) : X) \leqslant O(t \cdot \log t)$, we have*

$$\mathop{\mathbf{E}}_{j \sim \mathcal{U}_{[t]}} \mathop{\mathbf{E}}_{x \sim X_j} d\Big(f(X|_{X_j \neq x}) \ , \ f(X|_{X_j = x})\Big) \leqslant 1 - \frac{1}{\mathsf{poly}(t)} + \frac{1}{|\Sigma|} \, . \tag{6}$$

The notation $X|_{X_j \neq x}$ refers to the random variable that samples $x_i \sim X_i = \mathcal{U}_\Sigma$ independently for each $i \neq j$ as usual, and that samples x_j from the distribution X_j conditioned on the event that $X_j \neq x$, that is, the distribution $\mathcal{U}_{\Sigma \setminus \{a\}}$. The notation $X|_{X_j = x} = X|_{j \leftarrow x}$ is as before, that is, $x_j = x$ is fixed.

We defer the proof of the lemma to the end of this section and discuss now how to use it to obtain the stronger result for $O(t \log t)$ compressions. First note that we could not have directly used Lemma 4.2 in place of Lemma 3.3 in the proof of the main result, Theorem 3.1. This is because for $\Sigma = \{0,1\}$, the right-hand side of (6) becomes bigger than 1 and thus trivial. In fact, this is the reason why we formulated Lemma 3.6 for general Σ. We need to choose Σ with $|\Sigma| = \mathsf{poly}(t)$ large enough to get anything meaningful out of (6).

4.1 A different hypergraph tournament

To be able to work with larger Σ, we need to define the hypergraph tournament in a different way; not much is changing on a conceptual level, but the required notation becomes a bit less natural. We do this as follows.

Lemma 4.3. *Let $A : \binom{\{0,1\}^n}{\leqslant t} \to \{0,1\}^{\epsilon t}$. There exists a large enough constant $C \in \mathbb{N}$ such that with $\Sigma = [t^C]$ we have: For all $e = e_1 \dot\cup e_2 \dot\cup \ldots \dot\cup e_t \subseteq \{0,1\}^n$ with $|e_i| = |\Sigma|$, there exists an element $v \in e$ so that*

$$d\Big(A(X_e|_{v \notin X_e})\ ,\ A(X_e|_{v \in X_e})\Big) \leqslant 1 - \frac{1}{\mathsf{poly}(t)}\,, \tag{7}$$

where X_e is the distribution that samples the t-element set $\{\mathcal{U}_{e_1}, \ldots, \mathcal{U}_{e_t}\}$, and $X_e|_E$ is the distribution X_e conditioned on the event E.

For instance if $v \in e_1$, then $X_e|_{v \notin X_e}$ samples the t-element set $\{\mathcal{U}_{e_1 \setminus \{v\}}, \mathcal{U}_{e_2}, \ldots, \mathcal{U}_{e_t}\}$ and $X_e|_{v \in X_e}$ samples the t-element set $\{v, \mathcal{U}_{e_2}, \ldots, \mathcal{U}_{e_t}\}$. The proof of this lemma is analogous to the proof of Lemma 3.4.

Proof. We choose C as a constant that is large enough so that the right-hand side of (6) becomes bounded by $1 - 1/\mathsf{poly}(t)$. Let $e_{ia} \in \{0,1\}^n$ for $i \in [t]$ and $a \in \Sigma$ be the lexicographically a-th element of e_i. We define the function $f : \Sigma^t \to \{0,1\}^{O(t \log t)}$ as follows: $f(a_1, \ldots, a_t) \doteq A(e_{1a_1}, \ldots, e_{ta_t})$. Finally, we let the distributions X_i be $X_i = \mathcal{U}_\Sigma$ for all $i \in [t]$. We apply Lemma 3.3 to f and obtain indices $j \in [t]$ and $x \in \Sigma$ minimizing the statistical distance on the left-hand side of (6). Since $f(X_e|_{e_{jx} \notin X_e}) = A(X|_{X_j \neq x})$ and $f(X_e|_{e_{jx} \in X_e}) = A(X|_{X_j = x})$, we obtain the claim with $v \doteq e_{jx}$. ∎

4.2 Proof of Theorem 4.1

Proof (of Theorem 4.1). As in the proof of Theorem 3.1, we construct a deterministic $\mathsf{P/poly}$ reduction from L to a conjunction of polynomially many instances of the statistical distance problem $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$, but this time we let $D = 1$ and $\delta = 1 - \frac{1}{\mathsf{poly}(t)}$ be equal to the right-hand side of (7). Since there is a polynomial gap between d and D, Theorem2.1 implies that $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ is contained in the intersection of $\mathsf{NP/poly}$ and $\mathsf{coNP/poly}$. Since the intersection is closed under polynomial disjunctions, we obtain $L \in \mathsf{NP/poly} \cap \mathsf{coNP/poly}$. Thus it remains to find such a reduction.

To construct the advice at input length n, we use Lemma 4.3 with $t = t(n)$, which guarantees that the following hypergraph tournament $S : \binom{V}{|\Sigma| \cdot t} \to V$ with $V = \overline{L}_n$ is well-defined:

$$S(e) \doteq \min\Big\{ v \in e \ \Big|\ d\Big(A(X_e|_{v \notin X_e})\ ,\ A(X_e|_{v \in X_e})\Big) \leqslant \delta \Big\}\ .$$

We remark that if $|V| \leqslant |\Sigma| t = \mathsf{poly}(n)$, then we can use V as the advice. Otherwise, the advice at input length n is the dominating set $\mathcal{D} \subseteq \binom{V}{\Sigma \cdot t-1}$ guaranteed by Lemma 3.5; in particular, its size is bounded by $t \cdot |\Sigma| \cdot n = \mathsf{poly}(n)$.

The algorithm for L that uses $\mathrm{SD}^{\geqslant \Delta}_{\leqslant \delta}$ as an oracle works as follows: Let $v \in \{0,1\}^n$ be an instance of L given as input. If $v \in g$ holds for some $g \in \mathcal{D}$, the reduction rejects v and halts. Otherwise, for each $g \in \mathcal{D}$, it queries the SD-oracle on the instance $\left(A(X_e|_{v \notin X_e})\ ,\ A(X_e|_{v \in X_e})\right)$ with $e = g \cup \{v\}$. If the oracle claims that all queries are yes-instances, our reduction accepts, and otherwise, it rejects.

The correctness of this reduction is analogous to the proof Theorem 3.1: If $v \in L$, then Lemma 3.2 guarantees that the statistical distance of all queries is one, and so all queries will detect this. If $v \in \overline{L}$, then since $\mathcal{D}$ is a dominating set of S, we have $v \in g$ or $S(g \cup \{v\}) = v$ for some $g \in \mathcal{D}$. The latter will be detected in the query corresponding to g since $\delta < D$. This completes the proof of the theorem. ∎

4.3 Information-theoretic arguments

Proof (of Lemma 4.2). We use Vajda's inequality [18, 19] instead of Pinsker's inequality to bound the statistical distance in terms of the Kullback–Leibler divergence, which we in turn bound by the mutual information using Lemma 3.6 (with $\epsilon = C \cdot \log t$ for a constant C large enough so that $I(f(X) : X) \leqslant \epsilon t$ holds):

$$\begin{aligned}
&\mathop{\mathbf{E}}_{j}\ \mathop{\mathbf{E}}_{x} d\Big(f(X)\ ,\ f(X|_{j\leftarrow x})\Big) \\
&\leqslant \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} \Big(1 - \exp\Big(-1 - D_{\mathrm{KL}}(f(X)\ ||\ f(X|_{j\leftarrow x}))\Big)\Big) && \text{(Vajda's inequality)} \\
&\leqslant 1 - \exp\Big(-1 - \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} D_{\mathrm{KL}}\Big(f(X)\ ||\ f(X|_{j\leftarrow x})\Big)\Big) && \text{(Jensen's inequality)} \\
&\leqslant 1 - e^{-1-C\log t} = 1 - 1/\mathsf{poly}(t) && \text{(Lemma 3.6)}
\end{aligned}$$

Now (6) follows from the triangle inequality as follows.

$$\begin{aligned}
\mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} d\Big(f(X|_{X_j\neq x})\ ,\ f(X|_{X_j=x})\Big) &\leqslant \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} d\Big(f(X|_{X_j\neq x})\ ,\ f(X)\Big) \\
&\quad + \mathop{\mathbf{E}}_{j}\mathop{\mathbf{E}}_{x} d\Big(f(X)\ ,\ f(X|_{X_j=x})\Big) \\
&\leqslant \frac{1}{|\Sigma|} + 1 - \frac{1}{\mathsf{poly}(t)}\,.
\end{aligned}$$

For this, note that a simple calculation from (1) shows that

$$\begin{aligned}
&d(f(X|_{X_j\neq x}), f(X)) \leqslant d(X|_{X_j\neq x}, X) \\
&\quad\leqslant \Pr(X_j \neq x) \cdot d(X|_{X_j\neq x}, X|_{X_j\neq x}) + \Pr(X_j = x) \cdot d(X|_{X_j\neq x}, X|_{X_j=x}) \\
&\quad\leqslant \Pr(X_j \neq x) \cdot 0 + \Pr(X_j = x) \cdot 1 = \Pr(X_j = x)
\end{aligned}$$

always holds, and the latter equals $\frac{1}{|\Sigma|}$ since X_j is uniformly distributed on Σ. ∎

5 Extension: f-compression

We end this paper with a small observation: Instead of OR-compressions or AND-compressions, we could just as well consider f-compressions for a Boolean function $f : \{0,1\}^t \to \{0,1\}$. If the function f is symmetric, that is, if $f(x)$ depends only on the Hamming weight of x, then we can represent f as a function $f : \{0,\dots,t\} \to \{0,1\}$. We make the observation that Drucker's theorem applies to f-compressions whenever f is a non-constant, symmetric function.

Definition 3. Let $f : \{0,\dots,t\} \to \{0,1\}$ be any function. Then an f-compression of L into L' is a mapping

$$A : \binom{\{0,1\}^n}{\leqslant t} \to \{0,1\}^{\epsilon t}\,,$$

such that, for all $x \subseteq \binom{\{0,1\}^n}{\leqslant t}$, we have $A(x) \in L'$ if and only if $f(|x \cap L|) = 1$.

Examples:

- OR-compressions are f-compressions with $f(i) = 1$ if and only if $i > 0$.
- AND-compressions are f-compressions with $f(i) = 1$ if and only if $i = t$.
- Majority-compressions are f-compressions with $f(i) = 1$ if and only if $i > t/2$.
- Parity-compressions are f-compressions with $f(i) = 1$ if and only if i is odd.

We can apply Theorem 3.1 and 4.1 whenever f is not a constant function.

Lemma 5.1. *Let $f : \{0,\dots,t\} \to \{0,1\}$ be non-constant. Then every f-compression for L with size ϵt can be transformed into a compression for L or for $\overline{L}$, in the sense of Theorem 3.1 and with size bound at most $2\epsilon t$.*

Proof. Let A be an f-compression from L into L'. Then A is also a $(1-f)$-compression from L into $\overline{L'}$, an $(f(t-i))$-compression from $\overline{L}$ into L', and a $(1-f(t-i))$-compression from $\overline{L}$ into $\overline{L'}$. Since f is not constant, at least one of these four views corresponds to a function f' for which there is an index $i \leqslant t/2$ so that $f'(i) = 0$ and $f'(i+1) = 1$, holds. Assume without loss of generality that this holds already for f. Then we define $A' : \binom{\{0,1\}^n}{\leqslant t-i} \to \{0,1\}^{\epsilon t}$ as follows:

$$A'(\{x_i, x_{i+1}, \dots, x_t\}) \doteq A(\{\top_1, \dots, \top_{i-1}, x_i, x_{i+1}, \dots, x_t\})\,,$$

where $\top_1, \dots, \top_{i-1}$ are arbitrary distinct yes-instances of L. For the purposes of Theorem 3.1, these instances can be written in the non-uniform advice of A'. If this many yes-instances do not exist, then the language L is trivial to begin with. To ensure that the x_j's are distinct from the $\top_j$'s, we actually store a list of $2t$ yes-instances $\top_j$ and inject only $i-1$ of those that are different from the x_j's

A' is just like A, except that $i-1$ inputs have already been fixed to yes-instances. Then A' is a compressive map that satisfies the following: If $|x \cap L| = 0$ then $A'(x) \notin L'$, and if $|x \cap L| = 1$ then $A'(x) \in L'$. Since the number of inputs has decreased to $t' = t - i \geqslant t/2$, the new size of the compression is $\epsilon t \leqslant 2\epsilon t'$ in terms of t'. ■

Acknowledgments. I would like to thank Andrew Drucker, Martin Grohe, and others for encouraging me to pursue the publication of this manuscript, David Xiao for pointing out Theorem 2.1 to me, Dániel Marx for some comments on an earlier version of this paper, and Dieter van Melkebeek for some helpful discussions.

References

[1] Andrew Drucker, “New limits to classical and quantum instance compression,” in *Proceedings of the 53rd Annual Symposium on Foundations of Computer Science, FOCS 2012*, 2012, pp. 609–618. DOI: `10.1109/FOCS.2012.71`.

[2] Hans L. Bodlaender, Rodney G. Downey, Michael R. Fellows, and Danny Hermelin, “On problems without polynomial kernels,” *Journal of Computer and System Sciences*, vol. 75, no. 8, pp. 423–434, 2009, ISSN: 0022-0000. DOI: `10.1016/j.jcss.2009.04.001`.

[3] Lance Fortnow and Rahul Santhanam, “Infeasibility of instance compression and succinct PCPs for NP,” *Journal of Computer and System Sciences*, vol. 77, no. 1, pp. 91–106, 2011. DOI: `10.1016/j.jcss.2010.06.007`.

[4] Ker-I Ko, “On self-reducibility and weak P-selectivity,” vol. 26, pp. 209–211, 1983. DOI: `10.1016/0022-0000(83)90013-2`.

[5] Holger Dell and Dieter van Melkebeek, “Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses,” *Journal of the ACM*, 2014, to appear. [Online]. Available: `http://eccc.hpi-web.de/report/2010/038/`.

[6] Holger Dell, Valentine Kabanets, Dieter Melkebeek, and Osamu Watanabe, “Is Valiant–Vazirani’s isolation probability improvable?,” *Computational Complexity*, vol. 22, no. 2, pp. 345–383, 2013. DOI: `10.1007/s00037-013-0059-7`.

[7] Holger Dell and Dániel Marx, “Kernelization of packing problems,” in *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012*, 2012, pp. 68–81. DOI: `10.1137/1.9781611973099.6`.

[8] Danny Hermelin and Xi Wu, “Weak compositions and their applications to polynomial lower bounds for kernelization,” in *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012*, SIAM, 2011, pp. 104–113. DOI: `10.1137/1.9781611973099.9`.

[9] Hans L. Bodlaender, Bart M. P. Jansen, and Stefan Kratsch, “Kernelization lower bounds by cross-composition,” *SIAM Journal on Discrete Mathematics*, vol. 28, no. 1, pp. 277–305, 2014. DOI: `10.1137/120880240`.

[10] Stefan Kratsch, “Co-nondeterminism in compositions: a kernelization lower bound for a Ramsey-type problem,” in *Proceedings of the 23rd Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2012*, 2012, pp. 114–122. DOI: `10.1137/1.9781611973099.10`.

[11] Stefan Kratsch, Geevarghese Philip, and Saurabh Ray, “Point line cover: the easy kernel is essentially tight,” in *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014*, 2014, pp. 1596–1606. DOI: `10.1137/1.9781611973402.116`.

[12] Sanjeev Arora and Boaz Barak, *Computational complexity – A modern approach.* Cambridge University Press, 2009, ISBN: 978-0-521-42426-4.

[13] David Xiao, “New perspectives on the complexity of computational learning, and other problems in theoretical computer science,” PhD thesis, Princeton University, 2009. [Online]. Available: `ftp://ftp.cs.princeton.edu/techreports/2009/866.pdf`.

[14] Leonard M. Adleman, “Two theorems on random polynomial time,” in *Proceedings of the 19th Annual Symposium on Foundations of Computer Science, FOCS 1978*, 1978, pp. 75–83. DOI: `10.1109/SFCS.1978.37`.

[15] Amit Sahai and Salil Vadhan, “A complete problem for statistical zero knowledge,” *Journal of the ACM*, vol. 50, no. 2, pp. 196–249, 2003. DOI: `10.1145/636865.636868`.

[16] Oded Goldreich and Salil Vadhan, “On the complexity of computational problems regarding distributions (a survey),” Electronic Colloquium on Computational Complexity (ECCC), Tech report TR11-004, 2011. [Online]. Available: `http://eccc.hpi-web.de/report/2011/004/`.

[17] Thomas M. Cover and Joy A. Thomas, *Elements of information theory.* John Wiley & Sons, 2012.

[18] Alexei A. Fedotov, Peter Harremoës, and Flemming Topsoe, “Refinements of Pinsker’s inequality,” *IEEE Transactions on Information Theory*, vol. 49, no. 6, pp. 1491–1498, 2003. DOI: `10.1109/TIT.2003.811927`.

[19] Mark Reid and Bob Williamson, “Generalised Pinsker inequalities,” in *Conference on Learning Theory*, vol. 2009, 2009, pp. 18–21. [Online]. Available: `http://www.cs.mcgill.ca/~colt2009/papers/013.pdf`.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de