Electronic Colloquium on Computational Complexity, Report No. 9 (2007)

Nearly-Exponential Size Lower Bounds for Symbolic Quantifier Elimination Algorithms and OBDD-Based Proofs of Unsatisfiability

Nathan Segerlind
Department of Computer Science
Portland State University
Portland, Oregon
nsegerli@cs.pdx.edu

January 8, 2007

Abstract

We demonstrate a family of propositional formulas in conjunctive normal form so that a formula of size N requires size $2^{\Omega(\sqrt[7]{N/logN})}$ to refute using the tree-like OBDD refutation system of Atserias, Kolaitis and Vardi [3] with respect to all variable orderings. All known symbolic quantifier elimination algorithms for satisfiability generate tree-like proofs when run on unsatisfiable CNFs, so this lower bound applies to the run-times of these algorithms. Furthermore, the lower bound generalizes earlier results on OBDD-based proofs of unsatisfiability in that it applies for all variable orderings, it applies when the clauses are processed according to an arbitrary schedule, and it applies when variables are eliminated via quantification.

1 Introduction

Ordered binary decision diagrams (OBDDs) are data structures for representing Boolean functions [6, 7, 31] that are widely used when solving problems in circuit synthesis and model checking (cf. [6, 7, 30, 13]). A large number of OBDD-based algorithms have been implemented for solving the Boolean satisfiability problem [6, 42, 18, 10, 11, 1, 34, 33, 2, 14, 36, 22, 3, 24]. Many of these algorithms are known to efficiently generate proofs of unsatisfiability for CNFs known to require exponential running times for other methods, such as the pigeonhole principle that states $n + 1$ objects cannot be placed into n holes without a collision, and it is not immediately clear what the limitations of OBDD-based methods are. While it would immediately follow from the hypothesis $P \neq NP$ that such methods cannot solve all satisfiability instances in time polynomially-bounded by the input size, that sort of thinking strikes us as begging the question, and here we present *unconditional* limitations for algorithms of this kind: We unconditionally show that a wide class of OBDD-based satisfiability algorithms cannot solve all satisfiability instances in sub-exponential time. Prior analyses of the runtimes of OBDD-based satisfiability methods have been limited in their application because of assumptions on the order of processing the input clauses [20, 19] or an assumption on the variable ordering used when building the OBDDs [3], so this is the first unconditional lower bound that applies even to a system that explicitly constructs the OBDD for a CNF by selecting a variable ordering and then conjoining the clauses according to a heuristically chosen order.

ISSN 1433-8092

More formally, we present superpolynomial size lower bounds for the *tree-like OBDD refutation system* and satisfiability algorithms based on *explicit OBDD construction* and *symbolic quantifier elimination*. We give two motivations for studying minimum refutation sizes for proof systems and satisfiability algorithms. The first is that it is a necessary and tractable step towards understanding larger questions: Whether or not there is a polynomial-time algorithm for satisfiability, and whether or not propositional proof systems manipulating Boolean circuits can prove every tautology in size bounded by a polynomial in the size of the tautology (formalized as whether or the *extended-Frege proof systems* are polynomially bounded, cf. [26]). Both of these problems seem well beyond our current understanding. Rather than try to understand *all* polynomial-time computations or *all* extended-Frege proofs, we study the sizes of proofs of unsatisfiability for a particular class of satisfiability algorithms and extended-Frege proofs: In this case, tree-like OBDD refutations. Under this interpretation, the main result of this paper can be interpreted as saying "As far as symbolic quantifier elimination algorithms are concerned, P is different from NP." The second motivation is to develop taxonomy of satisfiability methods and identify the kinds of reasoning best suited to each method. Under this interpretation, the main result of this paper can be interpreted as saying "While symbolic quantifier elimination methods can perform efficiently on some structured formulas such as the $n+1$ to n pigeonhole principle, such methods inherently face an exponential blow-up when reasoning about the behavior of a system acted upon by a permutation."

1.1 Using OBBDs for Satisfiability and Propositional Proofs

One motivation for developing satisfiability algorithms based on OBDDs is the hope to escape the limitations of the resolution proof system. Most current satisfiability engines, in particular, the DLL with clause learning approach [29, 32, 17, 16], implement the resolution proof system [40] and therefore require exponential running times on the many CNFs known to require exponential size resolution refutations [21, 43, 12, 5, 37, 4]. The hope is that by developing algorithms that implement proof systems other than resolution, new satisfiability algorithms will be able to efficiently solve satisfiability instances not yet efficiently solvable.

An OBDD is a read-once branching program in which the variables appear according to a fixed order along every path (ie. the nodes are arranged in levels, all nodes at a level query the same variable, and each variable corresponds to at most one level). The choice of variable ordering can affect the size of the OBDD by an exponential factor and choosing a suitable variable ordering for a task is of utmost importance. The primary utility of the ordering restriction is that with respect to each fixed ordering, the OBDD computing a Boolean function is unique, up to a linear-time reduction to normal form (cf. [31]). Because of this canonicity property, the equality test for two Boolean functions represented as OBDDs is simply a check that their OBDDs are identical. Many simple but useful functions have small OBDDs with respect to some variable ordering, and many set operations, such as union and intersection, can be computed in polynomial time from two OBDDs. These properties make OBDDs well-suited for reasoning about symbolically encoded sets of states, and their use revolutionized the field of model checking [30, 13]. In light of this success, a number of attempts have been made to utilize OBDDs for more efficient satisfiability algorithms. This results of this paper apply to two such methods, explicit construction and symbolic quantifier elimination, but do not clearly apply to a third, compressed resolution.

Explicit construction. In the literature, this is sometimes called the "OBDD apply" method. In this method, a variable ordering is selected, the OBDD for the CNF with respect to that ordering is constructed, and it is checked whether this OBDD is the constant false [6]. Proofs in this system are straightforward: We begin with the OBDDs representing each clause, and we repeatedly conjoin them together until we obtain an OBDD for the conjunction of all the clauses. There are two opportunities for cleverness - the variable ordering used to construct the OBDDs, and the order in which the clauses are joined together, cf. [42, 1, 22]. Empirical studies [42, 14] and a mathematical analysis of the implementation in which the clauses are conjoined in the same order as the input presentation [20] have suggested that this method is incomparable with resolution-based methods.

Symbolic quantifier elimination. This method extends the explicit construction method by strategically eliminating variables via the application of existential quantifiers [18, 1, 36, 22, 41]. In particular, to determine if a CNF $\bigwedge_{i=1}^{m} C_i(\vec{x})$ is satisfiable, rather than build an OBDD for $\bigwedge_{i=1}^{m} C_i(\vec{x})$, it suffices to build one for $\exists \vec{x} \bigwedge_{i=1}^{m} C_i(\vec{x})$. This is can be more efficient because it is often the case that the OBDD for $\exists \vec{x} F(\vec{x}, \vec{y})$ are significantly smaller than the OBDD for $F(\vec{x}, \vec{y})$. One example of this approach is to first use heuristic methods to partition the variables into sets $X_1, \ldots X_k$ and the clauses into sets $A_1, \ldots A_k$ so that for each $i = 1, \ldots k$, the variables of X_i do not appear in the clauses belonging to sets $A_{i+1}, \ldots A_k$, then construct the OBDD for the quantified Boolean formula:

$$\exists X_k \left(\ldots \left(\exists X_2 \left(\exists X_1 \bigwedge_{C \in A_1} C(X_1, \ldots X_k) \right) \wedge \bigwedge_{C \in A_2} C(X_2, \ldots X_k) \right) \ldots \right) \wedge \bigwedge_{C \in A_k} C(X_k)$$

It has been observed that symbolic quantifier elimination leads to significant speed-ups over explicit OBDD construction on random 3-CNFs [18, 1], and that, on a certain mix of structured benchmarks, symbolic quantifier elimination solves more instances before time-out than solvers based on resolution or compressed resolution [22, 36].

When formalized as proof systems, these algorithms can be viewed as treelike versions of the OBDD propositional proof system described by Atserias, Kolaitis and Vardi [3]. This proof system is highly non-trivial: OBDDs are circuits not formulas, so this proof system is a kind of weak extended-Frege system[1]. Because it is not believed possible to convert OBDDs into formulas without an exponential blow-up, the OBDD proof system is not expected to be p-simulatable by Frege systems (in the sense of Cook and Reckhow [15]). The tree-like OBDD system possesses polynomial-size refutations of the $n+1$ to n pigeonhole principle, and it can p-simulate several interesting proof systems, such as tree-like resolution, Gaussian refutations over a finite field, and tree-like cutting planes refutations with unary coefficients [3].

Compressed resolution and compressed search. The analysis of this paper does not apply to these systems in a clear way, and we take a few paragraphs to to discuss why not. Compressed resolution and search methods use OBDDs (or sometimes, a variant known as *ZDDs* or *zero-suppressed binary decision diagrams, cf. [31]*) to encode exponentially large resolution refutations. A well-known example of this method is *multiresolution*, developed by Chatalic and Simon [10, 11]. In multiresolution, the set of clauses in the refutation is represented symbolically with a ZDD, and the Davis-Putnam variable elimination step is performed using ZDD operations, so that many resolution steps are handled simultaneously. In addition to the DP procedure, clause learning and breadth-first search algorithms have been implemented in the compressed setting [33, 34, 35].

The reason that the lower bound of this paper does not seem to apply to "compressed proof systems" is that in these systems, the OBDDs are not over the same variables as the input CNF. The

[1]For uninitiated, *Frege systems* are basically the standard textbook style systems of propositional logic manipulating Boolean formulas whereas *extended Frege systems* manipulate Boolean circuits. From a computational complexity perspective, Frege systems can be thought of as manipulating concepts definable in NC^1 and extended Frege systems can be thought of as manipulating concepts definable in P.

OBDDs symbolically encode a large resolution proof, so they work over new variables that encode clauses over the original variables. A typical encoding has for each literal l over original input CNF variables, a new variable y_l that corresponds to whether or not the literal l is present in a clause. In this way, compressed methods are akin to the "implicit proofs" described by Krajíček [27].

1.2 The Result and Comparisons with Earlier Work

The main result of this paper is that for infinitely many values of N, there is an unsatisfiable CNF Φ of size N so that every tree-like OBDD refutation of Φ has size at least $2^{\Omega(\sqrt[7]{N/\log N})}$ (Theorem 8). This lower bound generalizes earlier work on proving size lowerbounds for OBDD-based proofs of unsatisfiability in three ways: The proofs can use variable elimination via existential quantifiers, the clauses of the input CNF can be processed in any order (so long as they are recombined according to a tree-structure), and the variable ordering of the OBDDs can be arbitrary. The two previously published results regarding size lower bounds for OBDD-proofs of unsatisfiability either made use of a restriction on the order in which the clauses are processed, or held only for a fixed ordering on the variables.

In [20], Groote and Zantema prove a size lower bound for refutations in the OBDD-apply system that conjoins the clauses of the CNF in the order of the input listing (ie. to process $C_1 \wedge (C_2 \wedge C_3)$, an OBDD for $C_2 \wedge C_3$ is built and then one for $C_1 \wedge (C_2 \wedge C_3)$ is built). In fact, in that paper they give a size lower bound for refutations of a formula of the form $\neg x \wedge (x \wedge \psi)$, which is trivial to refute if the formula is processed as $(\neg x \wedge x) \wedge \psi$. Qualitatively, Theorem 8 generalizes their bound by applying to systems that eliminate variables by quantification, and by applying to systems that allow the clauses to be processed in an arbitrary manner. However, their bound is quantitatively stronger: Where N is the size of the difficult CNF, their bound on refutation size is $2^{\Omega(\sqrt{N})}$ whereas ours is $2^{\Omega(\sqrt[7]{N})}$.

In [3], Atserias, Kolaitis, and Vardi formalized the OBDD-based propositional proof system incorporating symbolic quantifier elimination, and proved that for each fixed variable ordering, there is a CNF of size N that requires size $2^{N^{\Omega(1)}}$ to refute in the OBDD proof system using that particular variable ordering. The two results are incomparable. The bound of [3] applies to the general (DAG-like) system, whereas Theorem 8 only applies to the tree-like system. On the other hand, Theorem 8 shows that there is a CNF for which every refutation with respect to every variable ordering has nearly-exponential size. The result of [3] says that for each variable ordering, there is a CNF for which that ordering is a poor choice, and does not elminate the possibility that for each CNF there is a variable ordering for which the CNF will be refuted in (say) time linear in the size of the CNF. Theorem 8 eliminates this possibility for the tree-like case, which includes all known implementations of these algorithms.

The analysis of Theorem 8 is the first that applies to all symbolic quantifier elimination algorithms so far developed [18, 1, 36, 22, 41]. It is not hard to see upon inspection that these algorithms generate proofs of unsatisfiability in the tree-like OBDD system. Moreover, the results of [20] do not apply to these methods as the methods typically perform a preprocessing analysis that chooses the order in which clauses are combined, and the methods eliminate variables via existential quantification. The results of [3] do not apply to these methods because the variable ordering is typically selected by some static analysis of the input CNF.

1.3 The Technique and its Comparison with Earlier Work

The argument is a reduction: We produce a CNF so that if there is a small refutation of the CNF in the tree-like OBDD proof system, then there is a low-communication randomized two-player protocol for the set-disjointness function. The set-disjointness function is known to require high communication [25, 39], so all refutations of this CNF must be large. The reduction is obtained by the interpolation by a communication game technique that has been well-used in the propositional proof complexity community for some time now [23, 3]. However, there is a wrinkle that complicates our return to this well-trodden path. Accounting for all possible variable orderings for the OBDDs corresponds to proving communication lower bounds that hold under all ways of partitioning the inputs, the so-called *best-case* partition model in communication complexity.

The analysis takes a turn from the beaten path at how the reduction fares under this best-case partitioning of variables. Indeed, the reduction can be thought of a variant of the reduction given by Raz and Wigderson [38] in which an adversarial partitioning of the variables has taken place. The reductions in [38, 23, 3] show that there is a search problem in variables $\vec{U}$ and $\vec{V}$, $Search(\vec{U}, \vec{V})$, and a randomized one-sided-error reduction from set-disjointness (in variables $\vec{X}$ and $\vec{Y}$) to $Search(\vec{U}, \vec{V})$ in which player I creates an assignment to $\vec{U}$ using $\vec{X}$ and player II creates an assignment to $\vec{V}$ using $\vec{Y}$. These reductions make heavy use of the structure inherent in the fixed partition of the variables of the search problem. In the best-case partition scenario that our reduction handles, we provide a search problem $Search(\vec{W})$ and show that no matter how the variables of $\vec{W}$ are partitioned into two equal-sized sets $\vec{U}$ and $\vec{V}$, there is a reduction from set-disjointness to the search problem in which player I to creates an assignment to $\vec{U}$ using $\vec{X}$ and player II to creates an assignment to $\vec{V}$ using $\vec{Y}$.

Over the course of the analyzing the randomized reduction, in particular, its distribution on placing gadgets, we develop a framework for passing local density results that hold for the uniform distribution to hold for distributions that we say are "generated by dependent domains with blocking processes". While these techniques are quite simple, they may be of interest for analyzing other random processes and reductions that exploit structure in dense graphs or set systems.

1.4 Outline of this Article

Sections 2 and 3 are notation and background. The CNF that we prove difficult for OBDD refutations is introduced in Section 4. Because of the central role of handling the partition of the variables, Section 5 is dedicated to the bookkeeping involved with handling partitions and defining the density of a partition, which is the parameter governing the quality of the reduction from set-disjointness.

We present the reduction and its analysis in an order that emphasizes the similarities with the reductions of [23] and [38], while encapsulating the differences in some lemmas that are proved in later sections. Section 6 includes the standard argument that a small treelike refutation yields a low-communication search protocol, although some work is needed to guarantee that the search protocol works for a partition of density $\Omega(1)$. Section 7 details the reduction proves the lower bound, modulo a lemma about the distribution on the gadgets used to build the reduction, Lemma 6. The marquee lower bound is presented in Subsection 7.1, Theorem 8.

In Section 8, we construct the objects claimed in Lemma 6. The distribution is very far from uniform, and this makes the analysis quite different from that of [38]. However, to make the reduction work, we need only two properties to hold. The first is that the probabilities assigned to objects at Hamming distance $\Omega(1)$ differ by at most a constant factor (encapsulated as Lemma 13, the "continuity lemma"), and the second is that events ensuring correctness of the reduction occur with probability not-too-much-less than they would under the uniform distribution (encapsulated as Lemma 12, the "completeness lemma"). Because the reduction is based on randomly flinging gadgets into the dense corners of a graph, the distributions get messy and it seems wise to pass to a cleaner framework as soon as possible. We call this framework *distributions from dependent domains with blocking processes*, or *DDWB distributions*. Section 10 lays out the notation used for the probability calculations and states some simple calculations that are needed, while Section 11 is devoted to DDWB distributions and their properties. In Section 12, we show that the distribution of Lemma 6 is a DDWB distribution and use this to prove the continuity lemma and the completeness lemma, which guarantee the correctness of the reduction.

1.5 Open Questions

The main question left open by this paper is to increase the constants for Theorem 8. The constant hidden in the $\Omega()$ of the $2^{\Omega(\sqrt[7]{N/\log N})}$ lowerbound of Theorem 8 is extremely small. Not logician small, but somewhere above Ramsey theorist small and way below computer scientist small. It is well below 2^{-500}. It is doubtful that this is strongest refutation-size lower bound that holds for the system, even for these particular CNFs.

The next question is whether or not we can go from the tree-like to the DAG-like case, ie. can a superpolynomial size lower bound be proved for DAG-like OBDD refutations of some family of CNFs? This would fully resolve the question posed in [3].

What can be said about the expected size of a (tree-like) OBDD refutation of a random 3-CNF? This is open even for the explicit OBDD construction method. It would be especially interesting if such an analysis could explain some of the threshold behavior observed in [14, 1].

It is common for OBDD packages to include a feature that dynamically recomputes the variable ordering when the OBDDs grow too large. The analysis of Theorem 8 does not cover this as the conversion from refutation to search (Lemma 3) seems to depends on every OBDD in a derivation using the same variable ordering. Current work with symbolic quantifier elimination algorithms for satisfiability has suggested that, given current technology, static variable orderings generally lead to better performance than dynamic variable orderings [1, 22]. This may be because these studies compare a default dynamic reordering heuristic against a static order that is customized for the satisfiability problem. A dynamic variable reordering method that consistently outperforms static methods remains unseen. On the other hand, there is no explanation of why static orderings should perform just as well as dynamic orderings. An interesting extension of this work would be to find a proof system that formalizes OBDD-proofs that include dynamic variable reordering and to use this to formally compare methods that use dynamic reordering with those that use static variable orderings. And of course, proving unconditional proof size lower bounds for algorithms that incorporate dynamic variable reordering would be interesting.

To the best of our knowledge, no non-trivial size lower bounds are known for any of the compressed methods [10, 11, 33, 34, 35]. Because these systems work with OBDDs, there is a similar flavor with the systems studied in this article. However, the fact the systems build OBDDs in different variables than those of the input CNF prevents an immediate application of Theorem 8 to these systems.

1.6 Acknowledgments

This work was inspired by conversations with Albert Atserias and Moshe Vardi about the OBDD refutation system at the Workshop on New Directions in Proof Complexity held at the Isaac Newton Institute for Mathematics, where Moshe discussed the findings of [3]. The author would like to thank Moshe and Albert for their enthusiasm and encouragement, as well as workshop co-organizer Jan Krajíček, who was helpful securing the author's attendance. The author also thanks Jan Friso Groote for answering some questions about [20]. Paul Beame provided useful comments on an early draft of the paper. Special thanks go to Cindy Brown and Barton Massey of Portland State University for their generous hospitality.

2 Notation and Communication Complexity Background

Definition 2.1 *The real numbers are denoted by* $\mathbb{R}$ *and* $[0,1]$ *denotes the closed unit interval. Let* n *be an integer. The set of integers* $\{1,\dots n\}$ *is denoted by* $[n]$*. For a set* S *and a non-negative integer* k*, the set of all* k*-tuples over* S *is denoted by* S^k *and the of all size* k *subsets of* S *is denoted by* $\binom{S}{k}$*. For a set* S *we let* χ_S *denote the indicator function for* S *with* $\chi_S(a) = 1$ *if* $a \in S$*,* $\chi_S(a) = 0$ *is* $a \notin S$*. The domain of* χ_S *will always clear from context. For a product space* $\prod_{i\in I} X_i$ *where* I *is a finite set, we will sometimes say that the product space is "*$|I|$ *dimensional" even though is no algebraic structure defined on* $\prod_{i\in I} X_i$*.*

Note that $\binom{[n]}{k}$ is a set with $\left|\binom{[n]}{k}\right| = \binom{n}{k}$.

Definition 2.2 *We use the word "graph" to mean a simple, loopless undirected graph. We use* $\subseteq$ *to denote the (not necessarily induced) subgraph relation, ie.* $G \subseteq H$ *if* $G = (V,E)$ *and* $H = (W,F)$ *with* $V \subseteq W$ *and* $E \subseteq F$ *(as sets). For any two disjoint nonempty sets* A *and* B*, we write* $K(A,B)$ *to denote the complete bipartite graph with partition* $\{A,B\}$*. Let* $G = (V,E)$ *be a graph. Let* $V_0 \subseteq V$ *and let* $E_0 \subseteq E$*. The set of edges* E_0 restricted to V_0*, written* $E_0[V_0]$*, is defined as* $E_0[V_0] = \{e \in E_0 \mid e \subseteq V_0\}$*.*

We use standard results on the randomized two-party communication complexity of the set-disjointness function. For a more thorough introduction to this subject, consult [28].

Definition 2.3 *Let* $f(\vec{X},\vec{Y})$ *be a function. A* randomized two-player protocol for f *is a two-party communication protocol in which Player I has private access to* $\vec{X}$*, Player II has private access to* $\vec{Y}$*, and the players share access to a source of random bits, so that for all inputs* $\vec{X}$ *and* $\vec{Y}$*, with probability at least* $2/3$*, the players agree upon the correct value of* $f(\vec{X},\vec{Y})$*. A* deterministic protocol *is one in which the answer arrived at by the players is independent of any randomness and is uniquely determined by the input* $\vec{X},\vec{Y}$*. The* cost of a protocol *is the maximum number of bits communicated between the two players taken over settings of the input and the random bits. The* randomized communication complexity of f *is the minimum cost of a randomized two-player protocol that computes* f*. The* set-disjointness function on n bits *is a Boolean function* $setdisj_n : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *with*

$$setdisj(\vec{X},\vec{Y}) = \begin{cases} 1 & \text{if } \exists i \in [n],\ X_i = Y_i = 1 \\ 0 & \text{otherwise} \end{cases}$$

Theorem 1 *([25, 39], cf. [28]) The two-party randomized communication complexity of* $setdisj_n$ *is* $\Omega(n)$*.*

3 The Ordered-Binary Decision Diagrams Refutation System

Definition 3.1 *(cf. [7, 31]) A* binary decision diagram *(also known as a* branching program*) is a rooted, directed acyclic graph in which every nonterminal node u labeled by a variable x_u and has two out-arcs, one two a node t_u and the other to a node f_u. Sinks are labeled by Boolean values. The function represented by a branching program is calculated by starting at the root and following a path to the sink as follows: If the current node u is labeled by the variable x_u, and x_u is assigned the value true, then follow the arc t_u, otherwise follow the arc labeled f_u. The value that the function takes is the value labeled on the sink. The* size *of a binary decision diagram is its number of nodes as a DAG. An* ordered binary decision diagram (OBDD) *is a binary decision diagram in which: Along every path from the source to a sink, every variable is queried at most once, and, there is fixed ordering of the variables $\preceq$ so that along all paths from the source to a sink, the order in which variables are queried is consistent with $\preceq$.*

For the purposes of our argument, we do not care if the OBDDs are reduced to canonical normal form. Indeed, all that is actually used about OBDDs is a simple connection between OBDDs and communication complexity that is the starting point for our reduction. We do not use it explicitly in this article, however, it is an ingredient for the proof of Lemma 4.

Proposition: If there is size S OBDD for a function $f(x_1, \dots x_n)$ with respect to some variable order $x_{i_1}, \dots x_{i_n}$, then for each $k \in [n]$, there is a two-party communication protocol computing f with respect to the variable partition $\{x_{i_1}, \dots x_{i_k}\}, \{x_{i_k+1}, \dots x_{i_n}\}$ that uses $\lceil \log S \rceil$ many bits of communication.

Proof sketch: The first player broadcasts the index of the node that is reached in the OBDD after following the path given by the assignment to $\{x_{i_1}, \dots x_{i_k}\}$. The second player continues computation from this node, using the values $\{x_{i_k+1}, \dots x_{i_n}\}$. No further communication is necessary because of the ordering on queries. ■

It is easy to see that the size of the OBDD representing a clause is no more than the size of the clause, plus the two sink nodes for "true" and "false". For this reason, we do not distinguish between a clause and its OBDD with respect to some order.

Proposition: Let C be a clause containing l literals. For every variable ordering, C can be represented by an OBDD of size at most $l + 2$.

Definition 3.2 *Let $\mathcal{C}$ be a set of clauses in variables from a set V. A* OBDD derivation from $\mathcal{C}$ with respect to a variable ordering $\preceq$ on V *is a sequence of OBDDs $F_1, \dots, F_m$ so that each OBDD is built from the variables of V with respect to the order $\preceq$, and each F_i either is a clause in $\mathcal{C}$, or follows from the preceding $F_1, \dots F_{i-1}$ by an application of one of the following inference rules: (A, A_0, and B are OBDDs in the variables V with ordering $\preceq$, where $A \Rightarrow A_0$ as Boolean functions, and $\vec{x}$, $\vec{y}$, $\vec{z}$ are tuples of variables from V):*

$$\textit{Subsumption: } \frac{A}{A_0} \qquad \textit{Conjunction: } \frac{A(\vec{x},\vec{y}) \quad B(\vec{y},\vec{z})}{A(\vec{x},\vec{y}) \wedge B(\vec{y},\vec{z})} \qquad \textit{Projection: } \frac{A(x,\vec{y})}{\exists x A(x,\vec{y})}$$

For a set of clauses $\mathcal{C}$, an OBDD refutation of $\mathcal{C}$ *is a derivation from $\mathcal{C}$ whose final line is the OBDD "false". The size of an OBDD refutation is the sum of the sizes of its OBDDs. An OBDD derivation $F_1, \dots F_m$ is said to be* treelike *if each F_i is used at most once as an antecedent to an inference.*

It is easily checked that the symbolic quantifier elimination algorithms for satisfiability all generate treelike OBDD refutations in the above system when run on unsatisfiable CNFs [18, 1, 22, 36] (so long as a dynamic variable reordering package is not in use).

The lower bound of Theorem 8 actually pertains to many different formulations of the tree-like OBDD refutation system. In particular, most sensible inference rules and axioms can be added and the lower bound will still apply. This is because the conversion from refutation to search protocols (cf. [23, 3]) requires only that (1) the refutation structure is tree-like (2) the OBDDs are in the same variables as the input CNF (3) the OBDDs are each built according to the same variable ordering, and (4) the inference rules are sound and of fan-in at most two. Lemma 2 of the current work requires that the proof structure is preserved under under simultaneous permutations of the variables (such a substitution does change the variable ordering $\preceq$, however).

4 The Difficult CNF: Indirect Matching Principles

The CNF $IndMatch_m$ is a propositional encoding of the fact that in a graph on $3m$ vertices, it is impossible to simultaneously have a perfect matching on $2m$ vertices and an independent set of size $2m+1$. It is similar to CNF $Match_m$ used by Impagliazzo, Pitassi, and Urquhart to prove size lower bounds for the tree-like cutting planes system [23]. However, in order to prove the CNFs difficult for tree-like OBDD refutations with respect to any variable ordering, we introduce a level of indirection via permutations.

4.1 The CNF $Match_m$

There are two distinct kinds of variable used in the CNF $Match_m$:

1. *The edge variables.* There are are $m \cdot \binom{3m}{2}$ many variables used to specify the matching: One variable x_e^i for each $i = 1, \dots m$ and each $e \in [3m]^2$. The intended semantics is that the variable x_e^i is equal to one if and only if the edge e is the i'th edge of the matching.
2. *The vertex variables.* There are $(2m+1)3m = 6m^2 + 3m$ many variables used to specify the independent set: One variable y_k^j for each $j = 1, \dots 2m+1$ and each $k = 1, \dots 3m$. The intended semantics is that the variable y_k^j is equal to one if and only if the element k is the j'th element of the independent set.

The set of all these variables is $MVars_m$. The following clauses form the CNF $Match_m$:

1. (At least m edges in the matching.) For each $i \in [m]$: $\bigvee_{e \in [3m]^2} x_e^i$
2. (Edges form a matching.) For each $i, j \in [2m]$ with $i \neq j$ and each $e, f \in [3m]^2$ with $e \cap f \neq \emptyset$: $\neg x_e^i \vee \neg x_f^j$
3. (At least $2m+1$ vertices in the independent set.) For each $j \in [2m+1]$: $\bigvee_{u \in [3m]} y_u^j$
4. (Vertices in the independent set are distinct.) For each $i, j \in [2m+1]$ with $i \neq j$ and each $u \in [3m]$: $\neg y_u^i \vee \neg y_u^j$
5. (The vertices are independent.) For each $e \in [3m]^2$ with $e = \{u, v\}$, each $k \in [m]$ and each $i, j \in [2m+1]$: $\neg y_u^i \vee \neg y_v^j \vee \neg x_e^k$

Notice that the CNF $Match_m$ has size $O(m^5)$.

4.2 The CNF $IndMatch_m$

The difference between the CNF $IndMatch_m$ and the CNF $Match_m$ is that we add variables specifying a permutation π, and for an assignment A to $MVars_m$, we interpret the independent set not as $\{u \mid \exists j \in [2m+1],\ A(y_u^j) = 1\}$ but instead as $\{\pi(u) \mid \exists j \in [2m+1],\ A(y_u^j) = 1\}$.

Definition 4.1 *Let N be given. A set Π of permutations of N is said to be* pairwise *independent if for all $a, b, c, d \in [N]$ with $a \neq b$ and $c \neq d$:*

$$Pr_{\pi \in \Pi}\left[\pi(a) = c \ \wedge\ \pi(b) = d\right] = \frac{1}{N(N-1)}$$

It is well-known that for any finite field, the set of mappings $\{x \mapsto ax + b \mid a \in \mathbb{F}^*,\ b \in \mathbb{F}\}$ is a pairwise independent family of permutations of size $|\mathbb{F}|(|\mathbb{F}| - 1)$.

Proposition: Whenever m is a power of 3, there is a pairwise-independent family of permutations of $[3m]$, Π_m, with $|\Pi_m| = 9m^2 - 3m$.

The variables used in the CNF $IndMatch_m$ are the variables used in $Match_m$, along with new variables for encoding a permutation: There are $l = \lceil \log(|\Pi|) \rceil$ many variables that encode a permutation from Π: $z_1, \ldots z_l$. The intended semantics is that the variables $z_1, \ldots z_l$ encode the permutations of Π in some surjective fashion. This set of permutation variables is denoted $PVars_m$. The set of variables $IMVars_m$ is $MVars_m \cup PVars_m$. The CNF $IndMatch_m$ has the same clauses of type 1, type 2, type 3 and type 4 that $Match_m$ has, whereas the clauses enforcing independence are as follows:

(Independence between vertices after application of the permutation.) For each $\alpha_1, \ldots \alpha_l \in \{0, 1\}$, each $e \in [3m]^2$ with $e = \{u, v\}$, each $k \in [m]$ and each $i, j \in [2m+1]$, with π denoting the element of Π encoded by $\vec{\alpha}$: $\bigvee_{i=1}^{L} z_i^{1-\alpha_i} \vee \neg y_{\pi(u)}^i \vee \neg y_{\pi(v)}^j \vee \neg x_e^k$

Notice that the CNF $IndMatch_m$ has $O(m^7)$ many clauses, and size $O(m^7 \log m)$.

Definition 4.2 *Let π be a permutation of $[3m]$. For each variable $v \in MVars_m$ we define*

$$\pi(v) = \begin{cases} y_{\pi(u)}^j & \text{if } v = y_u^j \text{ for some } j \in [2m+1],\ u \in [3m] \\ x_e^i & \text{if } v = x_e^i \text{ for some } i \in [m],\ e \in \binom{[3m]}{2} \end{cases}$$

Lemma 2 *Let $\pi \in \Pi$ be fixed. If Γ is a size S refutation of $IndMatch_m$ with variable ordering $v_1, \ldots v_N$, then there is a size S refutation of $Match_m$ that uses the variable ordering $\pi(v_1), \ldots \pi(v_N)$.*

Proof: Let α be the assignment to $\vec{z}$ that selects the permutation π^{-1}. We apply the restriction α to Γ, and we see that the clauses of $IndMatch_m$ that that are not satisfied are the non-independence clauses that do not use any $\vec{z}$ variables (ie. all clauses of type 1, type 2, type 3, and type 4), and the independence clauses of the form $\neg y_{\pi^{-1}(u)}^i \vee \neg y_{\pi^{-1}(v)}^j \vee \neg x_e^k$, for $i, j \in [2m+1]$, $u, v \in [3m]$, $k \in [m]$, and $e \in \binom{[3m]}{2}$. We now replace every occurrence of the variable y_u^i by $y_{\pi(u)}^i$. For the variable ordering, this means that y_u^i takes the place of $y_{\pi(u)}^i$ in the ordering. In each OBDD, each query to y_u^i is replaced by a query to $y_{\pi(u)}^i$. Every OBDD is now constructed according to the query order $\pi(v_1), \ldots \pi(v_N)$. It is easily checked that the proof structure is preserved under this substitution so that the new derivation is a derivation with respect to the order $\pi(v_1), \dots \pi(v_N)$ in the sense of Definition 3.2. Moreover, each clause $\neg y^i_{\pi^{-1}(u)} \vee \neg y^j_{\pi^{-1}(v)} \vee \neg x^k_e$, becomes $\neg y^i_u \vee \neg y^j_v \vee \neg x^k_e$, so that the new refutation is a refutation of $Match_m$. ■

5 Variable Partitions and Their Densities

The central task in the proof of Theorem 8 is to generate reductions from set-disjointness to the false-clause-search of $IndMatch_m$, given an arbitrary partitioning of the variables $IMVars_m$. In this brief subsection we present the machinery for analyzing these partitions. We view the partition as splitting the players into an *edge player*, with access to variables in $\mathcal{V}_I$, and a *vertex player*, with access to variables in $\mathcal{V}_{II}$. In the reduction, the edge player will place his set disjointness variables X_l on edge variables x^i_e and the vertex player will place his set-disjointness variables Y_l on vertex variables y^j_u.

Definition 5.1 *Let m be a positive integer, and let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$. For each $i = 1, \dots m$, define $E_i(\mathcal{V}_I)$ to be $\{e \in [3m]^2 \mid x^i_e \in \mathcal{V}_I\}$. For each $j = 1, \dots 2m+1$, define $V_j(\mathcal{V}_{II})$ to be $\{u \in [3m] \mid y^j_u \in \mathcal{V}_{II}\}$. Except for in the proof of Lemma 5, we do not discuss more than one variable partition at a time, so we usually write E_i instead of $E_i(\mathcal{V}_I)$ and V_j instead of $V_j(\mathcal{V}_{II})$.*

It is helpful to think of the variables of $MVars_m$ as being organized into m rows of edge variables and $2m+1$ rows of vertex variables, with E_i being the set of edge variables in row i available to Player I, and V_j being the set of vertex variables in row j available to Player II. A very important complication is that for distinct $i_1, i_2 \in [m]$, it is possible that $E_{i_1} \neq E_{i_2}$. This means that not only does the edge used in assignment matter, but the identity of the variable specifying the edge matters as well. The same complication is in play regarding the sets V_{j_1} and V_{j_2}. Because the identity of the variables matters, in contrast with the reduction of [38], we must treat the objects seen by the players as assignments to the variables, not merely sets of vertices and edges.

Definition 5.2 *Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$. The* density *of $(\mathcal{V}_I, \mathcal{V}_{II})$, $\delta(\mathcal{V}_I, \mathcal{V}_{II})$, is defined as follows:*

$$\delta(\mathcal{V}_I, \mathcal{V}_{II}) := \frac{1}{m^2(2m+1)^5} \sum_{\vec{\imath} \in [m]^2} \sum_{\vec{\jmath} \in [2m+1]^5} \frac{\left| \bigcap_{k=1}^5 E_{i_1}[V_{j_k}] \cap E_{i_2}[V_{j_k}] \right|}{\binom{3m}{2}}$$

6 From Refutation to Search

We transform small refutations of the $IndMatch_m$ principles into a low-communication protocol for a search problem in the variables $Mvars_m$.

Definition 6.1 *Let A be an assignment to $MVars_m$. We say that A is* non-degenerate *if it satisfies all of the clauses from $Match_m$ of type 1, type 2, type 3, and type 4. (Informally, this means that the assignment selects m distinct edges and $2m+1$ distinct vertices.) An edge $e \in \binom{[3m]}{2}$ is said to be* bad *for A if $e = \{u, v\}$ and there exist $i, j \in [2m+1], k \in [m]$ with $A(y^i_u) = 1$, $A(y^j_v) = 1$, and $A(x^k_e) = 1$.*

Proposition: If A is a non-degenerate assignment to $MVars_m$, then there exists an edge that is bad for A.

Definition 6.2 *Let m be a positive integer, and let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$. The search problem $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ is defined as follows:*

1. *Player I has private access to the variables of $\mathcal{V}_I$.*
2. *Player II has private access to the variables of $\mathcal{V}_{II}$.*
3. *Given a non-degenerate assignment A to $MVars_m$, the players must find a bad edge of A.*

The partition $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$ will play an important role in the quality of the reduction from set-disjointness. We will see that the larger the density of the partition, the larger the instances of set-disjointness that can be reduced to $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$. In particular, when $\delta(\mathcal{V}_I, \mathcal{V}_{II}) = \Omega(1)$, $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ requires communication $\Omega(m)$.

Lemma 3 *There a exists a constant $c > 0$ so that for all $m \geq 84651$, if there is a size S tree-like OBDD refutation of $IndMatch_m$ then there is a partition $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$ so that $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq 2^{-13}$ and there exists a deterministic two-player protocol for the search problem $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ that uses at most $c \log S$ many bits of communication.*

6.1 The Proof of Lemma 3

The following lemma follows from standard arguments.

Lemma 4 *(cf. [23, 3]) There exists a constant $c > 0$ so that for all m, and every partition $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$, if there is treelike OBDD refutation of $Match_m$ of size S that uses a variable order in which either every variable of $\mathcal{V}_I$ precedes every variable of $\mathcal{V}_{II}$, or vice-versa, then for each $i \in [n]$, then there is a deterministic two-player protocol for $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ that uses at most $c \log S$ many bit of communication.*

Lemma 5 *For $m \geq 84651$, if there exists size S refutation of $IndMatch_m$, then there exists a partition of $MVars_m$, $(\mathcal{V}_I, \mathcal{V}_{II})$, with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq 2^{-13}$, and a size S refutation of $Match_m$ in which every variable of $\mathcal{V}_I$ precedes every variable of $\mathcal{V}_{II}$, or vice-versa.*

Proof: Let $v_1, \ldots v_N$ be the variable ordering of $IMVars_m$ used by the refutation of $IndMatch_m$. Let i_0 be the first position to split either the set of vertex variables or the set of edge variables in half. More formally, for each $i = 1, \ldots N$, let $vvars(i)$ be the number of vertex variables in $\{v_1, \ldots v_i\}$, let $evars(i)$ be the number of edge variables in $\{v_1, \ldots v_i\}$, and let i_0 least integer with either $evars(i_0) \geq \frac{m}{2} \cdot \binom{3m}{2}$ or $vvars(i_0) \geq \frac{2m+1}{2} \cdot 3m$. Notice that there are two possible cases: The first is that $evars(i_0) \geq \frac{m}{2} \cdot \binom{3m}{2}$ so that $\{v_1, \ldots v_{i_0}\}$ contains exactly $\frac{m}{2} \cdot \binom{3m}{2}$ many edge variables and $\{v_{i_0+1}, \ldots v_N\}$ contains at least $\frac{1}{2} \cdot (6m^2 + 3m)$ many vertex variables. The second is that $vvars(i_0) \geq \frac{2m+1}{2} \cdot 3m$ so that $\{v_1, \ldots v_{i_0}\}$ contains exactly $\frac{1}{2} \cdot (6m^2 + 3m)$ many vertex variables and $\{v_{i_0+1}, \ldots v_N\}$ contains at least $\frac{m}{2} \cdot \binom{3m}{2}$ many edge variables. In the first case, we set $\mathcal{V}_I = \{v_1, \ldots v_{i_0}\}$ and $\mathcal{V}_{II} = \{v_{i_0+1}, \ldots v_N\}$. In the second case, we set $\mathcal{V}_{II} = \{v_1, \ldots v_{i_0}\}$ and $\mathcal{V}_I = \{v_{i_0+1}, \ldots v_N\}$. In either case, $\frac{1}{m} \sum_{i=1}^{m} |E_i| \geq \frac{1}{2}\binom{3m}{2}$ and $\frac{1}{2m+1} \sum_{i=1}^{2m+1} |V_j| \geq \frac{3m}{2}$. Therefore, by Lemma 16: $\frac{1}{m^2} \sum_{\vec{i} \in [m]^2} |E_{i_1} \cap E_{i_2}| \geq \frac{1}{4}\binom{3m}{2}$, and $\frac{1}{(2m+1)^5} \sum_{\vec{j} \in [2m+1]^5} |V_{j_1} \cap V_{j_2} \cap V_{j_3} \cap V_{j_4} \cap V_{j_5}| \geq \frac{3m}{32}$.

We now calculate the expected value of $\delta(\pi(\mathcal{V}_I), \pi(\mathcal{V}_{II}))$ over $\pi \in \Pi$. We begin by noting that for all $i \in [m]$, $E_i(\pi(\mathcal{V}_I)) = E_i(\mathcal{V}_I) = E_i$ and for all $j \in [2m+1]$, $V_j(\pi(\mathcal{V}_{II})) = \pi(V_j(\mathcal{V}_{II})) = \pi(V_j)$. For each $\vec{\imath} \in [3m]^2$, let $E_{\vec{\imath}} = E_{i_1} \cap E_{i_2}$ and for each $\vec{\jmath} \in [2m+1]^5$, let $V_{\vec{\jmath}} = V_{j_1} \cap V_{j_2} \cap V_{j_3} \cap V_{j_4} \cap V_{j_5}$. For each $\{u, v\} \in \binom{[3m]}{2}$, by the pairwise independence of the permutations, we have that:

$$\begin{aligned}
\Pr_{\pi\in\Pi}\left[\{\pi(u), \pi(v)\} \in E_{\vec{\imath}}\right] &= \sum_{\{a,b\}\in E_{\vec{\imath}}} \left(\Pr_{\pi\in\Pi}\left[\pi(u) = a,\ \pi(v) = b\right] + \Pr_{\pi\in\Pi}\left[\pi(u) = b,\ \pi(v) = a\}\right]\right) \\
&= \frac{2|E_{\vec{\imath}}|}{3m(3m-1)} = \frac{|E_{\vec{\imath}}|}{\binom{3m}{2}}
\end{aligned}$$

Therefore, by linearity of expectation, we have that:

$$\mathbb{E}_{\pi\in\Pi}\left[|E_{\vec{\imath}}\left[\pi\left(V_{\vec{\jmath}}\right)\right]|\right] = \sum_{\{u,v\}\in\binom{V_{\vec{\jmath}}}{2}} \Pr_{\pi}\left[\{\pi(u), \pi(v)\} \in E_{\vec{\imath}}\right] = \frac{|E_{\vec{\imath}}|}{\binom{3m}{2}} \binom{|V_{\vec{\jmath}}|}{2}$$

And thus we bound $\mathbb{E}_{\pi\in\Pi}\left[\delta(\pi(\mathcal{V}_I, \mathcal{V}_{II}))|\right]$ from below as follows:

$$\begin{aligned}
&\mathbb{E}_{\pi\in\Pi}\left[\frac{1}{m^2(2m+1)^5} \sum_{\vec{\imath}\in[m]^2} \sum_{\vec{\jmath}\in[2m+1]^5} \frac{|\bigcap_{k=1}^5 E_{i_1}(\pi(\mathcal{V}_I))\left[V_{j_k}(\pi(\mathcal{V}_{II}))\right] \cap E_{i_2}(\pi(\mathcal{V}_I))\left[V_{j_k}(\pi(\mathcal{V}_{II}))\right]|}{\binom{3m}{2}}\right] \\
= \ &\mathbb{E}_{\pi\in\Pi}\left[\frac{1}{m^2(2m+1)^5} \sum_{\vec{\imath}\in[m]^2} \sum_{\vec{\jmath}\in[2m+1]^5} \frac{|\bigcap_{k=1}^5 E_{i_1}\left[\pi(V_{j_k})\right] \cap E_{i_2}\left[\pi(V_{j_k})\right]|}{\binom{3m}{2}}\right] \\
= \ &\mathbb{E}_{\pi\in\Pi}\left[\sum_{\vec{\imath}\in[m]^2} \sum_{\vec{\jmath}\in[2m+1]^5} |E_{\vec{\imath}}\left[\pi\left(V_{\vec{\jmath}}\right)\right]|\right] = \sum_{\vec{\imath}\in[m]^2} \sum_{\vec{\jmath}\in[2m+1]^5} \mathbb{E}_{\pi\in\Pi}\left[|E_{\vec{\imath}}\left[\pi\left(V_{\vec{\jmath}}\right)\right]|\right] \\
= \ &\sum_{\vec{\imath}\in[m]^2} \frac{|E_{\vec{\imath}}|}{\binom{3m}{2}} \sum_{\vec{\jmath}\in[2m+1]^5} \binom{|V_{\vec{\jmath}}|}{2} \geq \sum_{\vec{\imath}\in[m]^2} \frac{|E_{\vec{\imath}}|}{\binom{3m}{2}} (2m+1)^5 \binom{3m/32}{2} \\
= \ &(2m+1)^5 \binom{3m/32}{2} \sum_{\vec{\imath}\in[m]^2} \frac{|E_{\vec{\imath}}|}{\binom{3m}{2}} \geq (2m+1)^5 \binom{3m/32}{2} m^2 \left(\frac{1}{4}\right) \\
= \ &\frac{m^2(2m+1)^5}{4} \frac{(3m/32)(3m/32-1)}{2} = \frac{m^2(2m+1)^5}{4\cdot(32)^2} \frac{(3m)(3m-32)}{2} \\
= \ &\frac{m^2(2m+1)^5}{4\cdot(32)^2}\left(\binom{3m}{2} - \frac{(3m)(31)}{2}\right) = \frac{m^2(2m+1)^5}{2^{12}} \binom{3m}{2}\left(1 - \frac{31}{3m-1}\right) \\
= \ &\left(2^{-12} - \frac{31}{3m-1}\right) m^2(2m+1)^5 \binom{3m}{2}
\end{aligned}$$

Choose a permutation π with $\sum_{\vec{\imath}\in[m]^2} \sum_{\vec{\jmath}\in[2m+1]^5} |E_{\vec{\imath}}\left[\pi\left(V_{\vec{\jmath}}\right)\right]| \geq \left(2^{-12} - \frac{31}{3m-1}\right) m^2(2m+1)^5 \binom{3m}{2}$. By Lemma 2, there is a size S refutation of $Match_m$ that uses the variable ordering $\pi(v_1), \ldots \pi(v_N)$. Notice that in this order, either every variable of $\pi(\mathcal{V}_I)$ precedes every variable of $\pi(\mathcal{V}_{II})$, or every variable of $\pi(\mathcal{V}_{II})$ precedes every variable of $\pi(\mathcal{V}_I)$. By the above calculation, $\delta(\pi(\mathcal{V}_I), \pi(\mathcal{V}_{II})) \geq 2^{-12} - \frac{31}{3m-1}$. Because $m \geq 84651$, we have $\frac{31}{3m-1} \leq 2^{-13}$, so $\delta(\pi(\mathcal{V}_I), \pi(\mathcal{V}_{II})) \geq 2^{-12} - 2^{-13} = 2^{-13}$.

■

To prove Lemma 3, simply take the partition of $MVars_m$ and the size S refutation of $Match_m$ guaranteed by Lemma 5 and feed them into Lemma 4.

7 Reduction and Lower Bound

The correctness of the reduction from $setdisj_n$ to $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ depends on the following lemma:

Lemma 6 *(proof in Section 8) For every $\delta > 0$, there exist $c_0, c_1 > 0$ so that for all $m \geq 31(2/\delta)^8$, and all partitions of $MVars_m$, $(\mathcal{V}_I, \mathcal{V}_{II})$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$, for all n with $n \leq c_0 m$, there exists a set $\mathcal{L}$, a distribution $\mathcal{D}$ on $\mathcal{L}$ with measure function μ, a function $A : \mathcal{L} \times \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^{MVars_m}$, and a function $pe : \mathcal{L} \to \binom{[3m]}{2}$ so that:*

1. *For all $L \in \mathcal{L}$, $(\vec{X}, \vec{Y}) \in \{0,1\}^n \times \{0,1\}^n$, all $v \in \mathcal{V}_I$, $A_{L,\vec{X},\vec{Y}}(v)$ is determined by L and $\vec{X}$, and for all $v \in \mathcal{V}_{II}$, $A_{L,\vec{X},\vec{Y}}(v)$ is determined by L and $\vec{Y}$.*

2. *For all $L \in \mathcal{L}$, all $(\vec{X}, \vec{Y}) \in \{0,1\}^n \times \{0,1\}^n$, the assignment $A_{L,\vec{X},\vec{Y}}$ is non-degenerate.*

3. *For all $(\vec{X}, \vec{Y}) \in \{0,1\}^n \times \{0,1\}^n$, and all $e \in \binom{[3m]}{2}$, if e is bad for $A_{L,\vec{X},\vec{Y}}$, then $e = pe(L)$ or $setdisj_n(\vec{X}, \vec{Y}) = 1$.*

4. *For all $(\vec{X}, \vec{Y}) \in \{0,1\}^n \times \{0,1\}^n$ with $setdisj_n(\vec{X}, \vec{Y}) = 1$, there exists $\mathcal{S} \subseteq \mathcal{L}$ with $\mu(\mathcal{S}) \geq \delta^8/2^9$ so that for all $A \in \{A_{L,\vec{X},\vec{Y}} \mid L \in \mathcal{S}\}$:*

$$\max_{e \in \binom{[3m]}{2}} \mu(pe(L) = e \mid A_{L,\vec{X},\vec{Y}} = A,\ L \in \mathcal{S}) \leq 1 - c_1$$

It is helpful to think of $L \in \mathcal{L}$ as a "layout" guiding the construction of an $MVars_m$ assignment from $\vec{X}$, $\vec{Y}$. $A_{L,\vec{X},\vec{Y}}$ is simply the assignment constructed using layout L with set-disjointness instance $(\vec{X}, \vec{Y})$. Condition 1 is the requirement that the Player I can compute the value of $A_{L,\vec{X},\vec{Y}}(v)$ for $v \in \mathcal{V}_I$ without communicating with Player II, and that player II can compute $A_{L,\vec{X},\vec{Y}}(v)$ for $v \in \mathcal{V}_{II}$ without communication. Condition 2 guarantees that the assignment created is a valid instance of the $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ problem. The function pe can be thought of as a "planted bad edge": The reduction is based on the idea of having positions with $X_k = Y_k = 1$ create bad edges. However, because the assignment is nondegenerate, there must always be some bad edge, even when $setdisj_n(\vec{X}, \vec{Y}) = 0$. The players knowingly create one such edge and we call this edge the planted edge for the layout, $pe(L)$. Condition 3 states that when $setdisj_n(\vec{X}, \vec{Y}) = 0$, the only bad edge is the planted edge. Condition 4 states that when $setdisj_n(\vec{X}, \vec{Y}) = 1$, conditioned on the layout coming from the set $\mathcal{S}$, no assignment is overly-correlated with a particular planted edge.

Lemma 7 *For all $\delta > 0$, there exist $C_0, C_1 > 0$ so that for all $m \geq 31(2/\delta)^8$, for all partitions of $MVars_m$, $(\mathcal{V}_I, \mathcal{V}_{II})$, with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$, for all $n \leq C_0 m$, if there is a two-player deterministic protocol $SEARCH$ that solves $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ using r bits of communication, then the randomized communication complexity of $setdisj_n$ is $\leq C_1 r$.*

Proof: Let C_0 be the c_0 as in the statement of Lemma 6. We give a one-sided reduction that never gives a wrong answer when $setdisj_n(\vec{X}, \vec{Y}) = 0$, and when $setdisj_n(\vec{X}, \vec{Y}) = 1$, it gives the correct answer with probability $\geq c_1 \delta^8/2^9$, where c_1 is the second constant guaranteed by Lemma 6. Repeating the protocol a constant number of times and returning a 0 only if all runs produce a 0 gives a protocol with correctness $\geq 2/3$.

1. Using public randomness, the players select a reduction layout L according to the distribution $\mathcal{D}$ guaranteed by Lemma 6.
2. The players run the protocol $SEARCH$ using the assignment $A_{L,\vec{X},\vec{Y}}$ and let e be the edge returned by the protocol $SEARCH$.
   (a) If $pe(L) = e$ then return 0.
   (b) If $pe(L) \neq e$ then return 1.

By Lemma 6, Condition 1, the players can compute the needed values of $A_{L,\vec{X},\vec{Y}}$ with no communication. By Lemma 6, Condition 2, the assignment $A_{L,\vec{X},\vec{Y}}$ is non-degenerate, and is therefore a legal input for the problem $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$. Consider the case when $\vec{X}$ and $\vec{Y}$ are disjoint. By Lemma 6, Condition 3, the only bad edge in $A_{L,\vec{X},\vec{Y}}$ is $pe(L)$, so the protocol returns 0. Consider the case when $\vec{X}$ and $\vec{Y}$ are intersecting. Apply Lemma 6, Condition 4, and let $\mathcal{S}$ be the set guaranteed for the pair $\vec{X}, \vec{Y}$. Define the event $\mathcal{B}$ as $\mathcal{B} = \{L \in \mathcal{S} \mid SEARCH(A_{L,\vec{X},\vec{Y}}) = pe(L)\}$. This is the event that the layout belongs to $\mathcal{S}$ and the protocol gives an erroneous answer. Let $A_{\mathcal{S}} = \{A_{L,\vec{X},\vec{Y}} \mid L \in \mathcal{S}\}$. For each $A \in A_{\mathcal{S}}$, let $\mathcal{S}_A = \{L \in \mathcal{S} \mid A_{L,\vec{X},\vec{Y}} = A\}$ and let $\mathcal{B}_A = \{L \in \mathcal{B} \mid A_{L,\vec{X},\vec{Y}} = A\}$. Because the protocol $SEARCH$ is deterministic, for each A on the set $\mathcal{B}_A$, the function $L \mapsto pe(L)$ is the constant function taking the value returned by $SEARCH(A)$. Therefore, by Lemma 6, Condition 4, for each $A \in A_{\mathcal{S}}$, $\mu(\mathcal{B}_A) \leq (1 - c_1)\mu(\mathcal{S}_A)$, and so:

$$\mu(\mathcal{B}) = \sum_{A \in A_{\mathcal{S}}} \mu(\mathcal{B}_A) \leq \sum_{A \in A_{\mathcal{S}}} \mu(\mathcal{S}_A)(1 - c_1) = (1 - c_1)\mu(\mathcal{S})$$

Therefore $\mu(\mathcal{S} \setminus \mathcal{B}) \geq c_1\mu(\mathcal{S}) \geq c_1\delta^8/2^9$. Of course, $\mathcal{S} \setminus \mathcal{B}$ is the event that $L \in \mathcal{S}$ and the protocol gives the answer 1.

■

7.1 The Lower Bound

Theorem 8 *There exists a constant $C > 0$ so that for sufficiently large m, every tree-like OBDD refutation of $IndMatch_m$ has size at least 2^{Cm}.*

Proof: Apply Theorem 1 and choose $N \geq 0$ and $c^* > 0$ so that for every $n \geq N$, randomized two-player protocols for solving $setdisj_n$ require $\geq c^*n$ bits of communication. Let C_0 and C_1 be the constants of Lemma 7, and let m be so large that $m \geq 31(2/(2^{-13}))^8 = 31 \cdot 2^{112}$ (so that we can apply Lemma 7 with $\delta \geq 2^{-13}$), and $N \leq \lfloor C_0 m \rfloor$ (so that we can apply Theorem 1). Set $n = \lfloor C_0 m \rfloor$. Let $c > 0$ be the constant from Lemma 3. Let Γ be a tree-like OBDD refutation of $IndMatch_m$ of size S. Because $m > 84651$, we may apply Lemma 3 and choose a partition $(\mathcal{V}_I, \mathcal{V}_{II})$ so that $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq 2^{-13}$ and a two-player deterministic communication protocol $FindBadEdge_m(\mathcal{V}_I, \mathcal{V}_{II})$ that uses at most $c \log S$ bits of communication. By Lemma 7, there is a two-party randomized communication protocol for $setdisj_n$ on inputs from $\mathcal{P}_n$ that exchanges at most $C_1 \log S$ bits of communication. Therefore, applying the communication bound for set-disjointness, $C_1 \log S \leq c^*n = c^*\lfloor C_0 m \rfloor$, and thus $S \leq 2^{\frac{c^*\lfloor C_0 m \rfloor}{C_1}}$ ■

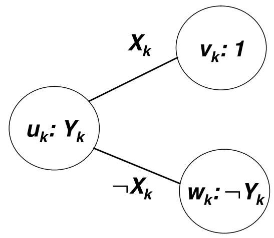

Figure 1: The basic set-disjointness gadget. A bad edge corresponds to the situation when an edge and both of its endpoints receive the label 1. The assignment uses: $x^{i_k}_{\{u_k,v_k\}} = X_k$, $x^{i_k}_{\{u_k,w_k\}} = \neg X_k$, $y^{j_{k,1}}_v = 1$, $y^{j_{k,2}}_{u_k} = Y_k$, and $y^{j_{k,2}}_{w_k} = \neg Y_k$. Notice that $\{u_k, w_k\}$ is never a bad edge, and that $\{u_k, v_k\}$ is a bad edge if and only if $X_k = Y_k = 1$.

8 Reduction Layouts

The reduction from set-disjointness by randomly generates "reduction layouts". A reduction layout is a framework for generating instances of the search problem from instances of set-disjointness, a collection of gadgets. We now take a moment to discuss the gadgets underlying the reduction from set-disjointness to the problem of finding a bad edge.

The basic idea is to create a bad edge for each k with $X_k = Y_k = 1$. To do this without communicating, the players use the public randomness to choose $u_k, v_k, w_k \in [3m]$ with the intent to place $\{u_k, v_k\}$ in the matching if $X_k = 1$ and $\{u_k, w_k\}$ in the matching if $X_k = 0$, and to place v_k in the independent no matter what, but to include u_k if $Y_k = 1$ and to include w_k if $Y_k = 0$. Of course, we must specify which variables are used to place the gadget, and those variables must be available to the players under the partition. The players use the public randomness to choose $i_k \in [m]$ with $x^{i_k}_{\{u_k,v_k\}}$, $x^{i_k}_{\{u_k,w_k\}} \in \mathcal{V}_I$ (equivalently, $\{u_k, v_k\}, \{u_k, w_k\} \in E_{i_k}$) and $j_{k,1}, j_{k,2} \in [m]$ with $y^{j_{k,1}}_{v_k}, y^{j_{k,2}}_{u_k}, y^{j_{k,2}}_{w_k} \in \mathcal{V}_{II}$, (equivalently, $v_k \in V_{j_{k,1}}$ and $u_k, w_k \in V_{j_{k,2}}$). The situation resembles that in Figure 1, with a bad edge occurring only if $X_k = Y_k = 1$ and only then only at $\{u_k, v_k\}$. The reduction plants one of these gadgets for each $k = 1, \dots n$.

Because there are m edges in the matching and $2m + 1$ vertices in the set, one more vertex must be placed in addition to the two associated with each set-disjointness gadget. A final gadget (thought of as being at position $n + 1$) will contain the "planted bad edge", in which three vertices u_{n+1}, v_{n+1}, and w_{n+1} are all placed in the set, and the edge $\{u_{n+1}, w_{n+1}\}$ is included. Because all three vertices are placed in the set, three variables $y^{j_{n+1,1}}_{u_{n+1}}$, $y^{j_{n+1,2}}_{v_{n+1}}$ and $y^{j_{n+1,3}}_{w_{n+1}}$ are needed with $u_{n+1} \in V_{j_{n+1,1}}$, $v_{n+1} \in V_{j_{n+1,2}}$, and $w_{n+1} \in V_{j_{n+1,3}}$.

The basic idea of the reduction is to randomly plant these $n + 1$ gadgets on disjoint variables. However, to ensure that the probabilities work out as claimed in Lemma 6, we make use of the density of the partition.

Definition 8.1 *Fix a partition of* $MVars_m$, $(\mathcal{V}_I, \mathcal{V}_{II})$. *Set* $\delta = \delta(\mathcal{V}_I, \mathcal{V}_{II})$. *For each* $i \in [m]$ *let* $E_i = E_i(\mathcal{V}_I)$ *and for each* $j \in [2m+1]$ *let* $V_j = V_j(\mathcal{V}_{II})$. *For each* $i \in [m]$, *let* $N_3(i) = \{(j_1, j_2, j_3) \in [2m+1]^3 \mid j_1 \neq j_2,\ j_2 \neq j_3,\ j_3 \neq j_1,\ |E_i[V_{j_1} \cap V_{j_2} \cap V_{j_3}]| \geq (\delta/3)\binom{3m}{2}\}$, *and let* $N_2(i) = \{(j_1, j_2) \in [2m+1]^2 \mid \exists j_3 \in [2m+1],\ (j_1, j_2, j_3) \in N_3(i)\}$. *Set* $G = \{i \in [m] \mid |N_3(i)| \geq (\delta/12)(2m+1)^3\}$. *Of course, each of* G, $N_3(\cdot)$, *and* $N_2(\cdot)$ *depend upon the partition* $(\mathcal{V}_I, \mathcal{V}_{II})$, *but we drop that from notation as we will never discuss more than one partition at a time.*

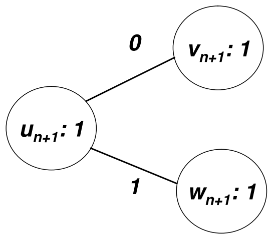

Figure 2: The set-disjointness gadget at the position with a planted bad edge. All three vertices $u_{n+1}, v_{n+1}, w_{n+1}$ are placed in the set of vertices and the edge $\{u_{n+1}, w_{n+1}\}$ is placed in the set of edges. The edge $\{u_{n+1}, w_{n+1}\}$ is a bad edge. The assignment uses: $x^{i_{n+1}}_{\{u_{n+1},v_{n+1}\}} = 0$, $x^{i_{n+1}}_{\{u_{n+1},w_{n+1}\}} = 1$, $y^{j_{n+1,1}}_{u_{n+1}} = 1$, $y^{j_{n+1,2}}_{v_{n+1}} = 1$, $y^{j_{n+1,3}}_{w_{n+1}} = 1$.

Lemma 9 *(Proof in Appendix, Section B) Let $\delta \in [0, 1]$ and let m be an integer $\geq 3/\delta$. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$. $|G| \geq (\delta/12)\, m$*

Definition 8.2 *Fix an integer m, a partition $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$.. A* reduction layout (with respect to $(\mathcal{V}_I, \mathcal{V}_{II})$, of length n) *is a tuple $(i_1, \ldots i_{n+1}, (j_{1,1}, j_{1,2}), \ldots (j_{n,1}, j_{n,2}), (j_{n+1,1}, j_{n+1,2}, j_{n+1,3}), (u_1, v_1, w_1), \ldots (u_{n+1}, v_{n+1}, w_{n+1}))$ from the set $[m]^{n+1} \times ([2m+1]^2)^n \times ([2m+1]^3) \times \left([3m]^3\right)^n$ with the following properties:*

1. *The indices $i_1, \ldots i_{n+1}$ are distinct.*
2. *The indices $j_{1,1}, j_{1,2}, \ldots j_{n,1}, j_{n,2}, j_{n+1,1}, j_{n+1,2}, j_{n+1,3}$ are distinct.*
3. *The integers $u_1, \ldots u_{n+1}, v_1, \ldots v_{n+1}, w_1, \ldots w_{n+1}$ are distinct.*
4. *For each $k = 1, \ldots n+1$, $\{u_k, v_k\} \in E_{i_k}$ and $\{u_k, w_k\} \in E_{i_k}$.*
5. *For each $k = 1, \ldots n+1$, $u_k, v_k, w_k \in V_{j_{k,1}} \cap V_{j_{k,2}}$.*
6. *$u_{n+1}, v_{n+1}, w_{n+1} \in V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}$.*
7. *For all $k \in [n+1]$, $i_k \in G$.*
8. *$(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_{n+1})$*
9. *For $k \in [n]$, each $(j_{k,1}, j_{k,2}) \in N_2(i_k)$.*

The set of all reduction layouts of length n with respect to $(\mathcal{V}_I, \mathcal{V}_{II})$ is denoted $\mathcal{L}_{m,n}(\mathcal{V}_I, \mathcal{V}_{II})$. When m, n, and $(\mathcal{V}_I, \mathcal{V}_{II})$ are clear from context, we simply write $\mathcal{L}$ and call $L \in \mathcal{L}$ a reduction layout.

When listing the elements of a reduction layout, we will abuse notataion write $(\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ despite the fact that a reduction layout is emphatically *not* a member of the set $[m]^{n+1} \times [2m+1]^{2n+3} \times [3m]^{n+1} \times [3m]^{n+1} \times [3m]^{n+1}$. This matters for the purpose of computing Hamming distances. The Hamming distance between two reduction layouts in $\mathcal{L}$ is their Hamming distance as elements of the $3n+3$ "dimensional" product set $[m]^{n+1} \times ([2m+1]^2)^n \times ([2m+1]^3) \times \left([3m]^3\right)^{n+1}$. In particular, if two reduction layouts $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ and $L^* = (\vec{\imath}^*, \vec{\jmath}^*, \vec{u}^*, \vec{v}^*, \vec{w}^*)$ differ in only that $(u_k, v_k, w_k) \neq (u^*_k, v^*_k, w^*_k)$ then they are at Hamming distance 1.

Definition 8.3 *Fix m, n, a partition $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$. Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ be a reduction layout from $\mathcal{L}$, and let $X_1, \dots X_n, Y_1, \dots Y_n$ be a set-disjointness instance. We define an assignment $A_{L,\vec{X},\vec{Y}}$ to the variables of $MVars_m$ as follows: Set $I = \{i_1, \dots i_{n+1}\}$. Set $J = \{j_{1,1}, j_{1,2}, \dots j_{n,1}, j_{n,2}, j_{n+1,1}, j_{n+1,2}, j_{n+1,3}\}$. Set $V = \{u_1, \dots u_{n+1}, v_1, \dots v_{n+1}, w_1, \dots w_{n+1}\}$. Let β, $\beta(L)$, be the lexicographically first assignment to the variables $\{x^i_e \mid i \in [m] - I,\ e \in [3m - V]^2\}$ $\cup\{y^j_u \mid j \in [2m+1] - J,\ u \in [3m] - V\}$ so that β defines a matching of size $m - n - 1$ and an independent set of size $2(m - n - 1)$. Define $A_{L,\vec{X},\vec{Y}}$ as follows:*

$$A_{L,\vec{X},\vec{Y}}(x^i_e) = \begin{cases} \beta(x^i_e) & \text{if } i \in [m] - I \text{ and } e \in ([3m] - V)^2 \\ X_k & \text{if } i = i_k \text{ and } e = \{u_k, v_k\} \text{ for some } k \in [n] \\ \neg X_k & \text{if } i = i_k \text{ and } e = \{u_k, w_k\} \text{ for some } k \in [n] \\ 1 & \text{if } i = i_{n+1} \text{ and } e = \{u_{n+1}, w_{n+1}\} \\ 0 & \text{otherwise} \end{cases}$$

$$A_{L,\vec{X},\vec{Y}}(y^j_x) = \begin{cases} \beta(y^j_x) & \text{if } j \in [2m+1] - j \text{ and } u \in [3m] - V \\ 1 & \text{if } j = j_{k,1} \text{ and } x = v_k \text{ for some } k \in [n] \\ Y_k & \text{if } j = j_{k,2} \text{ and } x = u_k \text{ for some } k \in [n] \\ \neg Y_k & \text{if } j = j_{k,2} \text{ and } x = w_k \text{ for some } k \in [n] \\ 1 & \text{if } j = j_{n+1,1} \text{ and } x = u_{n+1} \\ 1 & \text{if } j = j_{n+1,2} \text{ and } x = v_{n+1} \\ 1 & \text{if } j = j_{n+1,3} \text{ and } x = w_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

Notice that when both players have access to the layout L, condition 4 of Definition 8.2 ensures that Player I can compute the assignment to all variables in $\mathcal{V}_I$ by only consulting his private set-disjointness variables, and conditions 5 and 6 similarly guarantee that Player can compute the assignment to all variables in $\mathcal{V}_{II}$ by only consulting his private set-disjointness variables. This accounts for Condition 1 of Lemma 6. The conditions 1, 2 and 3 of Definition 8.2 ensure that $A_{L,\vec{X},\vec{Y}}$ is well-defined and non-degenerate. This accounts for Condition 2 of Lemma 6.

Definition 8.4 *Let m and n be given. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a variable partition for $MVars_m$. Let $\vec{X}$, $\vec{Y}$ be a set-disjointness instance, and let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ be a reduction layout from $\mathcal{L}_{m,n}$. The* planted edge for $\vec{X}, \vec{Y}, L$, *$pe(L)$, is defined to be $\{u_{n+1}, w_{n+1}\}$.*

Condition 3 of Lemma 6 is the content of the following lemma.

Lemma 10 *(Proof in Appendix Section B) Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ be a reduction layout. If e is a bad edge of $A_{L,\vec{X},\vec{Y}}$ then $e = pe(L)$, or, $e = \{u_l, v_l\}$ with $X_l = Y_l = 1$.*

9 The Distribution on Reduction Layouts

There is a technical point that we defer until after we describe the distribution: Why the experiment cannot "get stuck" and find itself in a position of attempting to choose an item from an empty set. For n a sufficiently small constant fraction of m, this is ruled out by some calculations that follow the description of the experiment. In the process that generates the distribution, we use the following auxiliary definitions:

Definition 9.1 *Let E be a set of edges over $[3m]$, and define $\mathcal{K}_{1,2}(E) := \{(u,v,w) \in [3m]^3 \mid v \neq w,\ \{u,v\} \in E,\ \{u,w\} \in E\}$. Let X be a set. For $U \subseteq X$ define $pm_X(U) : \{(u,v) \in X^2 \mid \{u,v\} \cap U \neq \emptyset\}$ and $tm_X(U) := \{(u,v,w) \in X^3 \mid \{u,v,w\} \cap U \neq \emptyset\}$. (The mnemonic for this notation is "pairs over X that meet U" and "triples over X that meet U".)*

Definition 9.2 *Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a variable partition for $MVars_m$. Let G, $N_3(\cdot)$, and $N_2(\cdot)$ be as in Definition 8.1. The distribution $\mathcal{D}$ on $\mathcal{L}$ is given by the following experiment:*

1. *For each $k = 1, \ldots n+1$: Choose i_k from $G \setminus \{i_1, \ldots i_{k-1}\}$.*
2. *Set $J = \emptyset$.*
3. *For each $k = 1, \ldots n$:*
   (a) *Uniformly choose $(j_{k,1}, j_{k,2})$ from $N_2(i_k) \setminus pm_{[2m+1]}(J)$*
   (b) *Set $J := J \cup \{j_{k,1}, j_{k,2}\}$*
4. *Uniformly choose $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3})$ from $N_3(i_{n+1}) \setminus tm_{[2m+1]}(J)$*
5. *Set $J := J \cup \{j_{n+1,1}, j_{n+1,2}, j_{n+1,3}\}$*
6. *Set $V^* = \emptyset$.*
7. *For each $k = 1, \ldots n$:*
   (a) *Uniformly choose (u_k, v_k, w_k) from $\mathcal{K}_{1,2}(E_{i_k}\left[(V_{j_{k,1}} \cap V_{j_{k,2}})\right]) \setminus tm_{[3m]}(V^*)$.*
   (b) *Set $V^* = V^* \cup \{u_k, v_k, w_k\}$.*
8. *Uniformly choose $(u_{n+1}, v_{n+1}, w_{n+1})$ from $\mathcal{K}_{1,2}(E_{i_{n+1}}\left[(V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}})\right]) \setminus tm_{[3m]}(V^*)$.*
9. *Return the layout $(\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$.*

Proposition: For all $L \in \mathcal{L}$, $\mu(L) > 0$.

The above proposition can be checked by iteratively noting that when we condition on the experiment producing a prefix of L, the probability that it selects the next coordinate of L is non-zero.

The results of the following lemma guarantee that when γ is sufficiently small with respect to δ, the experiment does not "get stuck". The proof is in the Appendix.

Lemma 11 *Let $\delta \in [0,1]$ and let m be an integer $\geq 450/\delta^2$. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$. Let n given with $\gamma = \frac{n+1}{m}$. For all runs of the experiment in Definition 9.2, and for each $k = 1, \ldots n$:*

1. $|G \setminus \{i_1, \ldots i_{k-1}\}| > ((\delta/12) - \gamma)m$.
2. $|N_2(i_k) \setminus pm_{[2m+1]}(J) \geq ((\delta/3) - 2\gamma)(2m+1)^2$
3. $|N_3(i_{n+1}) \setminus tm_{[2m+1]}(J)| \geq ((\delta/3) - 3\gamma)(2m+1)^3$
4. $|\mathcal{K}_{1,2}\left(E_{i_k}\left[V_{j_{k,1}} \cap V_{j_{k,2}}\right]\right) \setminus tm_{[3m]}(V^*)| \geq (\delta^2/10 - 3\gamma)(3m)^3$

5. $|\mathcal{K}_{1,2}\left(E_{i_{n+1}}\left[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}\right]\right) \setminus tm_{[3m]}(V^*)| \geq (\delta^2/10 - 3\gamma)(3m)^3$

The following two statements are used to prove Lemma 6. Their proofs depend upon calculations regarding the distribution $\mathcal{D}$, and seem to be best put in the framework of "distributions from dependent domains processes with blocking".

Definition 9.3 *A reduction layout $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ is said to be l-*switchable *if $(j_{n+1,2}, j_{l,1}, j_{l,2}) \in N_3(i_l)$ and $K(\{u_{n+1}, u_l\}, \{v_{n+1}, v_l, w_{n+1}, w_l\}) \subseteq E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}]$. Let $\mathcal{S}^l$ denote the set of l-switchable reduction layouts from $\mathcal{L}$.*

Lemma 12 *("Completeness lemma", proof in Section 12) For all $\delta > 0$, for all $m \geq 31(2/\delta)^8$, all partitions $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$, for all $n \leq \frac{\delta^{10}}{2^{10}\cdot 3\cdot 5^2}m$, for all $l \in [n]$, $\mu(\mathcal{S}^l) \geq \delta^8/2^9$.*

Lemma 13 *("Continuity lemma", proof in Section 12) For every $\delta > 0$ for every integer $d \geq 1$ for all $m \geq 450/\delta^2$, for all partitions $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$, for all $n \leq (\delta^2/60)m$, for all reduction layouts $L, L^* \in \mathcal{L}$ with $HD(L, L^*) \leq k$, $\mu(L^*) \geq (\delta^2/20)^{2d} e^{-3d} \cdot \mu(L)$.*

9.1 The Proof of Lemma 6

To prove Lemma 6 we use the following helper lemma.

Lemma 14 *(Proof immediately follows that of Lemma 6.) For all $\delta > 0$, all $m \geq 450/\delta^2$, all partitions $(\mathcal{V}_I, \mathcal{V}_{II})$ of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$, all $n \leq (\delta^2/20)m$, and all set-disjointness instances $(\vec{X}, \vec{Y})$, there exists an involution $f : \mathcal{S}^l \to \mathcal{S}^l$ so that for all $L \in \mathcal{S}^l$, $A_{L,\vec{X},\vec{Y}} = A_{f(L),\vec{X},\vec{Y}}$, $pe(f(L)) \neq pe(L)$, and $\mu(f(L)) \geq \mu(L)(\delta^2/20)^{12} e^{-18}$.*

Proof:(of Lemma 6 from Lemma 14) Let $\delta > 0$ be given. Set $c_0 = \frac{\delta^{10}}{2^{10}\cdot 3\cdot 5^2}m$ Let $m \geq 31(2/\delta)^8$ and $n \leq c_0$ be given. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$ with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$. We take $\mathcal{L} = \mathcal{L}_{m,n}(\mathcal{V}_I, \mathcal{V}_{II})$ per Definition 8.2, $\mathcal{D} = \mathcal{D}_{m,n}(\mathcal{V}_I, \mathcal{V}_{II})$ per Definition 9.2, $A : (L, \vec{X}, \vec{Y}) \to A_{L,\vec{X},\vec{Y}}$ per Definition 8.3, and pe per Definition 8.4.

Condition 1 and Condition 2 follow immediately from Definition 8.2, and Condition 3 follows from Lemma 10. What remains to be shown is that Condition 4 holds. Let $(\vec{X}, \vec{Y}) \in \{0,1\}^n \times \{0,1\}^n$ with $setdisj_n(\vec{X}, \vec{Y}) = 1$ be given. Choose $l \in [n]$ with $X_l = Y_l = 1$ and set $\mathcal{S} = \mathcal{S}^l$. By Lemma 12, $\mu(\mathcal{S}^l) \geq \delta^8/2^9$. Set $c = (\delta^2/20)^{12} e^{-18}$ (The constant of Lemma 14.) We now show that for all assignments A to $MVars_m$:

$$\max_e \mu(pe(L) = e \mid A_{L,\vec{X},\vec{Y}} = A,\ L \in \mathcal{S}^l) \leq 1/(1+c)$$

Let A be an assignment to $MVars_m$ and let $e \in \binom{[3m]}{2}$ be given. Let $\mathcal{B}^e_A = \{L \in \mathcal{S}^l \mid A_{L,\vec{X},\vec{Y}} = A,\ pe(L) = e\}$, let $\mathcal{S}^l_A = \{L \in \mathcal{S}^l \mid A_{L,\vec{X},\vec{Y}} = A\}$. Take take as f guaranteed by Lemma 14. Because f maps $\mathcal{S}^l$ to $\mathcal{S}^l$, we have that $f(\mathcal{B}^e_A) \subseteq \mathcal{S}^l$, because $A_{f(L),\vec{X},\vec{Y}} = A_{L,\vec{X},\vec{Y}} = A$, we have that $f(\mathcal{B}^e_A) \subseteq \mathcal{S}^l_A$, and because $pe(f(L)) \neq pe(L) = e$, we have that $f(\mathcal{B}^e_A) \subseteq \mathcal{S}^l_A \setminus \mathcal{B}^e_A$. Because f is an involution of $\mathcal{S}^l$, it is injective, and because $\mu(f(L)) \geq c\mu(L)$ for all L, we have that $\mu(\mathcal{S}^l_A \setminus \mathcal{B}^e_A) \geq \mu(f(\mathcal{B}^e_A)) \geq c_1\mu(\mathcal{B}^e_A)$ and therefore $\mu(\mathcal{S}^l_A) = \mu(\mathcal{S}^l_A \setminus \mathcal{B}^e_A) + \mu(\mathcal{B}^e_A) \geq (1+c)\mu(\mathcal{B}^e_A)$. Therefore: $\mu(pe(L) = e \mid A_{L,\vec{X},\vec{Y}} = A,\ L \in \mathcal{S}^l) = \mu(\mathcal{B}^e_A \mid \mathcal{S}^l_A) = \frac{\mu(\mathcal{B}^e_A)}{\mu(\mathcal{S}^l_A)} \leq \frac{1}{1+c}$. Noting that $1/(1+c) = 1 - c/(1+c)$, we set $c_1 = c/(1+c)$ and we conclude the proof of Lemma 6. ■

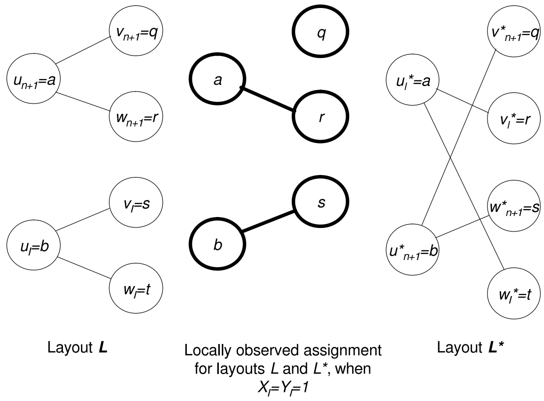

Figure 3: With layouts L and L^*, when $X_l = X_l = 1$, set of vertices and edges specified by the assignments $A_{L,\vec{X},\vec{Y}}$ and $A_{L^*,\vec{X},\vec{Y}}$ are equal. Notice however, that the planted edge under L is $\{a, r\}$ whereas the planted edge under L^* is $\{b, s\}$.

Proof:(of Lemma 14) Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$. We define $f(L) = (\vec{\imath}, \vec{\jmath}^*, \vec{u}^*, \vec{v}^*, \vec{w}^*)$ below. The basic the idea is to modify the reduction layout L by swapping some vertices between the gadgets at positions $n+1$ and l so that the planted edge changes but the assignment remains the same. This is graphically illustrated in Figure 3. Because of the partitioning of the variables, it is not immediately the case that L^* will be a reduction layout. Among other things, we need to ensure that $\{u_l^*, w_l^*\} \in E_{i_l^*}$ and $\{j_{n+1,1}^*, j_{n+1,2}^*, j_{n+1,3}^*\} \in N_3(i_{n+1}^*)$, which is where we make use of the hypothesis that L is l-switchable[2]. We give the full definition of L^* below, along with the case analysis ensuring that the conclusions of the lemma hold.

$$i_k^* = \begin{cases} i_{n+1} & \text{if } k = l \\ i_l & \text{if } k = n+1 \\ i_k & \text{otherwise} \end{cases} \qquad u_i^* = \begin{cases} u_l & \text{if } i = n+1 \\ u_{n+1} & \text{if } i = l \\ u_i & \text{otherwise} \end{cases}$$

$$j_{k,1}^* = \begin{cases} j_{n+1,3} & \text{if } k = l \\ j_{l,2} & \text{if } k = n+1 \\ j_{k,1} & \text{otherwise} \end{cases} \qquad v_k^* = \begin{cases} w_{n+1} & \text{if } k = l \\ v_k & \text{otherwise} \end{cases}$$

$$j_{k,2}^* = \begin{cases} j_{n+1,1} & \text{if } k = l \\ j_{k,2} & \text{otherwise} \end{cases} \qquad w_k^* = \begin{cases} v_l & \text{if } k = n+1 \\ w_k & \text{otherwise} \end{cases}$$

$$j_{n+1,3}^* = j_{l,1}$$

We now check each of the properties required by Lemma 14. This is just case analysis and rewriting. However, in order to show that $f(L) \in \mathcal{S}^l$ we make use of the hypothesis that L is l-switchable.

The mapping f is an involution. This is verified by iterating the definition of f. The details are carried out in the Appendix, Section C.

$A_{L,\vec{X},\vec{Y}} = A_{f(L),\vec{X},\vec{Y}}$. This is follows from expanding the definitions and doing a little bookkeeping, we put the argument in the Appendix, Section C.

[2]A reader carefully checking the case analysis below will note that the definition of l-switchable is a bit stronger than we need. See the discussion in Section 13.

$pe(L) \neq pe(f(L))$**.** Because $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ is a reduction layout, $\{u_{n+1}, w_{n+1}\} \cap \{u_l, v_l\} = \emptyset$. Applying Definition 8.4, we see that $pe(L) = \{u_{n+1}, w_{n+1}\} \neq \{u_l, v_l\} = \{u^*_{n+1}, w^*_{n+1}\} = pe(f(L))$.

$\mu(f(L)) \geq \mu(L) \cdot (\delta^2/20)^{12} e^{-18}$**.** In order to show this, we need that $\mu(L) > 0$ (which holds because $L \in \mathcal{L}$) and $\mu(f(L)) > 0$ (which depends on the fact that $f(L) \in \mathcal{L}$, which we show below). For now we take the non-zero mass of $f(L)$ as a given. The differences between L and $f(L)$ occur only with: $i_{n+1} \neq i^*_{n+1}$, $i_l \neq i^*_l$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \neq (j^*_{n+1,1}, j^*_{n+1,2}, j^*_{n+1,3})$, $(j_{l,1}, j_{l,2}) \neq (j^*_{l,1}, j^*_{l,2})$, $(u_l, v_l, w_l) \neq (u^*_l, v^*_l, w^*_l)$, and $(u_{n+1}, v_{n+1}, w_{n+1}) \neq (u^*_{n+1}, v^*_{n+1}, w^*_{n+1})$. Therefore $HD(L, f(L^*)) \leq 6$. We apply Lemma 13 to deduce that $\mu(f(L)) \geq \mu(L) \cdot (\delta^2/20)^{12} e^{-18}$.

For each $L \in \mathcal{S}^l$, $f(L) \in \mathcal{S}^l$. First we check that $f(L) = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ is indeed a reduction layout. We check each property from Definition 8.2:

1. The indices $i^*_1, \ldots i^*_{n+1}$ are distinct: This holds because $\vec{\imath}^*$ is a permutation of $\vec{\imath}$.
2. The indices $j^*_{1,1}, j^*_{1,2}, \ldots j^*_{n,1}, j^*_{n,2}, j^*_{n+1,1}, j^*_{n+1,2}, j^*_{n+1,3}$ are distinct: This holds because $\vec{\jmath}^*$ is a permutation of $\vec{\jmath}$.
3. The integers $u^*_1, \ldots u^*_{n+1}, v^*_1, \ldots v^*_{n+1}, w^*_1, \ldots w^*_{n+1}$ are distinct: This is true because $u^*_1, \ldots u^*_{n+1}$, $v^*_1, \ldots v^*_{n+1}$, $w^*_1, \ldots w^*_{n+1}$ is a permutation of $u_1, \ldots u_{n+1}, v_1, \ldots v_{n+1}, w_1, \ldots w_{n+1}$.
4. For each $k = 1, \ldots n+1$, $\{u^*_k, v^*_k\} \in E_{i^*_k}$ and $\{u^*_k, w^*_k\} \in E_{i^*_k}$: Because
$$K(\{u_l, u_{n+1}\}, \{v_l, v_{n+1}, w_l, w_{n+1}\}) \subseteq E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}]$$
we have that $\{u^*_l, v^*_l\} = \{u_{n+1}, w_{n+1}\} \in E_{i_{n+1}} = E_{i^*_l}$, $\{u^*_l, w^*_l\} = \{u_{n+1}, w_l\} \in E_{i_{n+1}} = E_{i^*_l}$, $\{u^*_{n+1}, v^*_{n+1}\} = \{u_l, v_{n+1}\} \in E_{i_l} = E_{i^*_{n+1}}$, and $\{u^*_{n+1}, w^*_l\} = \{u_l, w_l\} \in E_{i_l} = E_{i^*_{n+1}}$. For $k \in [n] \backslash \{l\}$, we have that $\{u^*_k, v^*_k\} = \{u_k, v_k\} \in E_{i_k} = E_{i^*_k}$ and $\{u^*_k, w^*_k\} = \{u_k, w_k\} = E_{i_k} \in E_{i^*_k}$.
5. For each $k = 1, \ldots n+1$, $\{u^*_k, v^*_k, w^*_k\} \subseteq V_{j^*_{k,1}} \cap V_{j^*_{k,2}}$: Because
$$K(\{u_l, u_{n+1}\}, \{v_l, v_{n+1}, w_l, w_{n+1}\}) \subseteq E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}]$$
we have that $\{u^*_l, v^*_l, w^*_l\} = \{u_{n+1}, w_{n+1}, w_l\} \subseteq V_{j_{n+1,3}} \cap V_{j_{n+1,1}} = V_{j^*_{l,1}} \cap V_{j^*_{l,2}}$. For the same reason, $\{u^*_{n+1}, v^*_{n+1}, w^*_{n+1}\} = \{u_l, v_{n+1}, v_l\} \subseteq V_{j_{l,2}} \cap V_{j_{n+1,2}} = V_{j^*_{n+1,1}} \cap V_{j^*_{n+1,2}}$. For $k \in [n] \setminus \{l\}$, we have that $\{u^*_k, v^*_k, w^*_k\} = \{u_k, v_k, w_k\} \subseteq V_{j_{k,1}} \cap V_{j_{k,2}} = V_{j^*_{k,1}} \cap V_{j^*_{k,2}}$.
6. We have that $\{u^*_{n+1}, v^*_{n+1}, w^*_{n+1}\} = \{u_l, v_{n+1}, v_l\} \subseteq V_{j_{l,1}} = V_{j^*_{n+1,3}}$, because
$$K(\{u_l, u_{n+1}\}, \{v_l, v_{n+1}, w_l, w_{n+1}\}) \subseteq E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}]$$
7. For each $k \in [n+1]$, $i^*_k \in G$: This holds because $\vec{\imath}^*$ is a permutation of $\vec{\imath}$ and for each $k \in [n+1]$, $i_k \in G$.
8. $(j^*_{n+1,1}, j^*_{n+1,2}, j^*_{n+1,3}) \in N_3(i^*_{n+1})$: Because L is l-switchable, $(j_{n+1,1}, j_{l,1}, j_{l,2}) \in N_3(i_l)$, therefore, $(j^*_{n+1,1}, j^*_{n+1,2}, j^*_{n+1,3}) = (j_{l,2}, j_{n+1,2}, j_{l,1}) \in N_3(i_l) = N_3(i^*_{n+1})$.
9. For each $k = 1, \ldots n$: $(j^*_{k,1}, j^*_{k,2}) \in N_2(i^*_k)$. For $k \in [n] \setminus \{l\}$, we have that $(j^*_{k,1}, j^*_{k,2}) = (j_{k,1}, j_{k,2}) \in N_2(i_k) = N_2(i^*_k)$. When $k = l$, because L is a reduction layout, we have that $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_{n+1})$, and therefore $(j_{n+1,3}, j_{n+1,1}) \in N_2(i_{n+1})$. Thus: $(j^*_{l,1}, j^*_{l,2}) = (j_{n+1,3}, j_{n+1,1}) \in N_2(i_{n+1}) = N_2(i^*_l)$.

This establishes that $f(L) \in \mathcal{L}$. That $f(L) \in \mathcal{S}^l$ follows immediately from the hypothesis that $L \in \mathcal{S}^l$ and the definitions: $(j^*_{n+1,2}, j^*_{l,1}, j^*_{l,2}) = (j_{n+1,2}, j_{n+1,3}, j_{n+1,1}) \in N_3(i_{n+1}) = N_3(i^*_l)$ and

$$\begin{aligned} K(\{u^*_l, u^*_{n+1}\}, \{v^*_l, v^*_{n+1}, w^*_l, w^*_{n+1}\}) &= K(\{u_l, u_{n+1}\}, \{v_l, v_{n+1}, w_l, w_{n+1}\}) \\ &\subseteq E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}] \\ &= E_{i_l}[V_{j_{l,2}} \cap V_{j_{n+1,2}} \cap V_{j_{l,1}}] \cap E_{i_{n+1}}[V_{j_{n+1,3}} \cap V_{j_{n+1,1}}] \\ &= E_{i^*_{n+1}}[V_{j^*_{n+1,1}} \cap V_{j^*_{n+1,2}} \cap V_{j^*_{n+1,3}}] \cap E_{i^*_l}[V_{j^*_{l,1}} \cap V_{j^*_{l,2}}] \end{aligned}$$

■

10 Probability Notation and Background

Definition 10.1 *Let X_i, $i \in I$, be a family of sets indexed by a set I; we write X_I as an abbreviation for the product $\prod_{i\in I} X_i$. Let $\prod_{i\in I} X_i$ and $\prod_{j\in J} X_j$ be product spaces with $I \cap J = \emptyset$. For $\vec{x} \in \prod_{i\in I} X_i$ and $\vec{y} \in \prod_{i\in J} X_i$ we write $\vec{x}\vec{y}$ to denote the concatenation of $\vec{x}$ and $\vec{y}$ (an element of $\prod_{i\in I\cup J} X_i$). We use the same indices for elements in tuples as we do for the factors of the product, ie. for $\vec{u} \in \prod_{i=j}^{t} X_i$, we write $\vec{u} = (u_j, \dots u_t)$, we* do not *write $\vec{u} = (u_1, \dots u_{t-j+1})$. Let f be a function whose domain is a product space $\prod_{i=1}^{t} X_i$. For each $j \in [t]$, for each $\vec{x} \in \prod_{i=1}^{j} X_i$, we write $f^{\vec{x}}$ to denote the curried function with domain $\prod_{i=j+1}^{t} X_i$, that is, $f^{\vec{x}}(\vec{y}) = f(\vec{x}\vec{y})$.*

Definition 10.2 *Let η be a probability distribution over a set X and let $f : X \to \mathbb{R}$. We write $\mathbb{E}_\eta[f]$ to denote the expectation of f with respect to η. At times, the uniform distribution over a set will be written as U. Other times, we will write with $E \subseteq S$, we will write $Pr_{x\in S}[E]$ to denote the probability that $x \in E$ holds when x is selected uniformly from S.*

Definition 10.3 *Let η be a probability distribution on a product space $\prod_{i=1}^{t} X_i$. For each $I \subseteq [t]$, let η_I be the marginal distribution of η on $\prod_{i\in I} X_i$. For each $j \in [t]$ and each $\vec{x} \in \prod_{i=1}^{j} X_i$, let $\eta^{\vec{x}}$ be the probability distribution on $\prod_{i=j}^{t} X_i$ given by the formula $\eta^{\vec{x}}(\vec{y}) = \frac{\eta(\vec{x}\vec{y})}{\eta_{[j]}(\vec{x})}$ if $\eta_{[j]}(\vec{x}) \neq 0$ and 0 otherwise.*

Notice that $\eta^{\vec{x}}$ is the marginal distribution of η to the coordinates $[t] \setminus [j]$ conditioned on the event that the first j coordinates take the value $\vec{y}$. An immediate consequence of the definitions:

Lemma 15 *Let $f : \prod_{i=1}^{t} X_i \to \mathbb{R}$, let $I = \{1, \dots i_0\}$: $\mathbb{E}_\eta[f] = \sum_{\vec{u}\in X_I} \eta_I(\vec{u}) \mathbb{E}_{\eta^{\vec{u}}}[f^{\vec{u}}]$*

Unsurprisingly for a technique based on finding structure in a dense family of sets, we beat the stuffing out Jensen's Inequality, its relatives, and any averaging arguments that we find in the neighborhood.

Proposition:(Jensen's Inequality) Let $f : D \to \mathbb{R}$, let $g : \mathbb{R} \to \mathbb{R}$ be a convex function, and let η be a probability distribution on D. $\mathbb{E}_\eta[g \circ f] \geq g(\mathbb{E}_\eta[f])$.

Lemma 16 *(Proof in the Appendix, section A.) Let X be a finite set, and let $Y_1, \dots Y_n$ be a family of subsets of X. Set $\alpha = \frac{1}{n}\sum_{i=1}^{n} |Y_i|/|X|$, and let k be a non-negative integer: $\frac{1}{n^k}\sum_{\vec{i}\in[n]^k} |\bigcap_{l=1}^{k} Y_{i_l}| \geq \alpha^k |X|$.*

Lemma 17 *(Proof in the Appendix, section A.) There exists a constant $c > 0$ so that for every undirected graph $G = (V, E)$ with $|V| = N$ and $|E| \geq \alpha\binom{N}{2}$. We have that:*

$$\begin{array}{lcl} Pr_{\vec{u}\in V^3}[K(\{u_1\},\{u_2,u_3\}) \subseteq G] & \geq & \alpha^2 - (5/N) \\ Pr_{\vec{u}\in V^6}[K(\{u_1,u_2\},\{u_3,u_4,u_5,u_6\}) \subseteq G] & \geq & \alpha^8 - (23/N) \end{array}$$

Proposition: Let η be a probability measure on a space X, and let $f : X \to [0,1]$ be measurable. For all $\epsilon \in [0,1]$ and all $c > 0$ $\eta(\{x \mid f(x) \geq \frac{1}{c}\mathbb{E}_\eta[f]\}) \geq (1 - 1/c)\mathbb{E}_\eta[f]$.

11 Distributions from DDWB Processes

To prove the completeness lemma (Lemma 12) and the continuity lemma (Lemma 13), we make some detailed calculations about the distribution $\mathcal{D}$. It seems that by moving to slightly more general framework, some of the calculations and case analyses are simplified. In Lemma 20 in Section 12 we show that the distribution $\mathcal{D}$ falls into this framework and use the machinery of DDWB processes developed in this section to finish the proofs of Lemma 12 and Lemma 13.

Definition 11.1 *Let t be an integer, $X_1, \dots X_t$ be sets, and let $S_i : \prod_{k=1}^{i-1} X_k \to \mathfrak{P}(X_i)$, and $F_i : \prod_{k=1}^{i-1} X_k \to \mathfrak{P}(X_i)$ be families of maps with $i \in [t]$. Assume that for all $i = 1, \dots t$, and all $(u_1, \dots u_{i-1}) \in \prod_{k=1}^{i-1} X_i$, $S_i(u_1, \dots u_{i-1}) \setminus F(u_1, \dots u_{i-1}) \neq \emptyset$ and $S_i(u_1, \dots u_{i-1}) \setminus F_i(u_1, \dots u_{i-1}) \neq \emptyset$.*

The distribution given by the dependent domains with blocking process of $\vec{S}$ and $\vec{F}$ *is the distribution $\pi(= \pi_{\vec{S},\vec{F}})$ on $\prod_{i=1}^{t} X_i$ given by the random process that generates a sequence $(u_1, \dots u_t)$ as follows: For $i = 1, \dots t$, choose u_i uniformly from $S_i(u_1, \dots u_{i-1}) \setminus F_i(u_1, \dots u_{i-1})$. The* blockage bound *of a DDWB process $\vec{S}$, $\vec{F}$ is the smallest $\beta \geq 0$ so that for all $i = 1, \dots t$ and all $\vec{u} \in \prod_{k=1}^{i-1} X_k$, $|F_i(\vec{u})| \leq \beta|S_i(\vec{u})|$. The* covering bound *for $\vec{S}$, $\vec{F}$ is the largest $\kappa \in [0,1]$ so that for all $i = 1, \dots t$ and all $\vec{u} \in \prod_{k=1}^{i-1} X_k$, $|S_i(\vec{u}) \setminus F_i(\vec{u})| \geq \kappa|X_i|$.*

The following easy fact is the crux of an induction argument.

Proposition: Let π be the distribution on $\prod_{i=1}^{t} X_i$ given by the DDWB process $\vec{S}$, $\vec{F}$. For each $a \in X_1$, The distribution π_a is generated by the DDWB process on $\prod_{i=2}^{t} X_i$ given by $S_2^a, \dots S_t^a$, $F_2^a, \dots F_t^a$. If the process $\vec{S}$, $\vec{F}$ has a blockage bound $\leq \beta$, then the process $\vec{S}^a$, $\vec{F}^a$ has a blockage bound $\leq \beta$.

11.1 Loss of Expectation Lemma for DDWB Distributions

The following lemma is used to pass density results for the uniform distribution, such as Lemma 17, to certain DDWB distributions. This is how Lemma 12 will be proved. It is a simple but careful combination of two observations: If the domains S_i contain the support of a $[0,1]$ valued function, then uniformly selecting over the S_i's (instead of all of X_i) will only increase the expectation. Of course the blocking of the F_i's could reduce the expectation, but for a DDWB with blockage bound β, each coordinate that the event depends upon can reduce the expectation by at most β.

Lemma 18 *Let $\prod_{i=1}^{t} X_i$ be a product space, and let $f : \prod_{i=1}^{t} X^i \to [0,1]$ be a function that depends upon at most k coordinates, $i_1, \dots i_k$. Let U be the uniform distribution on $\prod_{i=1}^{t} X_i$, and let π be a DDWB distribution on $\prod_{i=1}^{t} X_i$ given by some $\vec{S}$ and $\vec{F}$. If the following two conditions are satisfied:*

1. The DDWB process $\vec{S}$, $\vec{F}$ has blockage bound $\leq \beta$.

2. For all $\vec{a} \in \prod_{i=1}^{t} X_i$, if $f(\vec{a}) > 0$ then for all $j = 1, \ldots k$, $a_{i_j} \in S_{i_j}(a_1, \ldots a_{i_j-1})$.

Then $\mathbb{E}_\pi[f] \geq \mathbb{E}_U[f] - k\beta$.

Proof: We prove the claim by induction on k. The lemma clearly holds for $k = 0$, as in that case f is constant over $\prod_{i=1}^{t} X_i$, and therefore $\mathbb{E}_\pi[f] = \mathbb{E}_U[f]$. We now assume that the lemma holds for functions that depend on only k coordinates, and demonstrate that it holds for functions that depend on only $k+1$ coordinates.

Let t, $\prod_{i=1}^{t} X_i$, π, $\vec{S}$, $\vec{F}$, and be given as in the statement of the lemma- with f dependent only upon $k+1$ coordinates, $i_1, \ldots i_{k+1}$. Let $i = i_1$ be the first coordinate upon which the function f depends. Set $I = [i-1]$ and $J = [t] \setminus [i]$. Let $X_I = \prod_{k \in I} X_k$ and $X_J = \prod_{k \in J} X_k$.

We reduce to the induction hypothesis by showing that for each $\vec{u} \in X_I$, $a \in X_i$, the conditions of the induction hypothesis are met for the function $f^{\vec{u}a}$, with process $\vec{S}^{\vec{u}a}$, $\vec{F}^{\vec{u}a}$, and distribution $\pi^{\vec{u}a}$. Observe that the distribution $\pi^{\vec{u}a}$ is given by the DDWB process $S_{i+1}^{\vec{u}a}, \ldots S_t^{\vec{u}a}$ and $F_{i+1}^{\vec{u}a}, \ldots F_t^{\vec{u}a}$, a process with blockage bound $\leq \beta$ because $\vec{S}$, $\vec{F}$ has blockage bound $\leq \beta$. Moreover, the function $f^{\vec{u}a} : \prod_{j=i+1}^{t} X_i \to [0,1]$ depends on at most k coordinates. By specializing the hypothesis "for all $\vec{a}$, if $f(\vec{a}) > 0$ then for all $j = 1, \ldots k$, $a_{i_j} \in S_{i_j}(a_1, \ldots a_{i_j-1})$" to inputs with prefix $\vec{u}a$ and weakening its conclusion to cover only $j = 2, \ldots k$, we have that "for all $\vec{b} \in X_J$ so that $f(\vec{u}a\vec{b}) > 0$, for all $j = 2, \ldots k$, $b_{i_j} \in S_{i_j}(\vec{u}, a, b_{i+1}, \ldots b_{i_j-1})$". This is equivalent to "for all $\vec{b} \in X_J$ so that $f^{\vec{u}a}(\vec{b}) > 0$, for all $j = 2, \ldots k$, $b_{i_j} \in S_{i_j}^{\vec{u}a}(b_{i+1}, \ldots b_{i_j-1})$". Therefore by the induction hypothesis we have that $\mathbb{E}_{\pi^{\vec{u}a}}[f^{\vec{u}a}] \geq \mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] - k\beta$.

Furthermore, from the hypothesis "for all $\vec{u} \in \prod_{i=1}^{t} X_i$, if $f(\vec{u}) > 0$ then $\forall j \in [k+1]$, $u_{i_j} \in S_{i_j}(u_1, \ldots u_{i_j-1})$" we conclude that for all $\vec{v} \in \prod_{j=1}^{i} X_j$ with $\mathbb{E}_{U^{\vec{v}}}[f^{\vec{v}}] > 0$, $v_i \in S_i(v_1, \ldots v_{i-1})$. Therefore, for all $\vec{u} = (u_1, \ldots u_{i-1}) \in X_I$

$$\mathbb{E}_{U^{\vec{u}}}[f^{\vec{u}}] = \sum_{a \in X_i} \frac{1}{|X_i|} \mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] = \sum_{a \in S_i(\vec{u})} \frac{1}{|X_i|} \mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] \leq \sum_{a \in S_i(\vec{u})} \frac{1}{|S_i(\vec{u})|} \mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}]$$

We now bound the expectation of f with respect to π from below:

$$\begin{aligned}
\mathbb{E}_\pi[f] &= \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in X_i} \sum_{\vec{b} \in X_J} \pi^{\vec{u}}(a\vec{b}) f(\vec{u}a\vec{b}) \\
&= \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in X_i} \sum_{\vec{b} \in X_J} \frac{\chi_{S_i(\vec{u}) \setminus F_i(\vec{u})}(a)}{|S_i(\vec{u}) \setminus F_i(\vec{u})|} \pi^{\vec{u}a}(\vec{b}) f(\vec{u}a\vec{b}) \\
&= \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in S_i(\vec{u})} \sum_{\vec{b} \in X_J} \frac{1 - \chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u}) \setminus F_i(\vec{u})|} \pi^{\vec{u}a}(\vec{b}) f(\vec{u}a\vec{b}) \\
&= \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in S_i(\vec{u})} \frac{1 - \chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u}) \setminus F_i(\vec{u})|} \sum_{\vec{b} \in X_J} \pi^{\vec{u}a}(\vec{b}) f(\vec{u}a\vec{b}) \\
&= \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in S_i(\vec{u})} \frac{1 - \chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u}) \setminus F_i(\vec{u})|} \mathbb{E}_{\pi^{\vec{u}a}}[f^{\vec{u}a}] \\
&\geq \sum_{\vec{u} \in X_I} \pi_I(\vec{u}) \sum_{a \in S_i(\vec{u})} \frac{1 - \chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u}) \setminus F_i(\vec{u})|} \left(\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] - k\beta \right)
\end{aligned}$$

$$
\begin{aligned}
&= -k\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \sum_{a\in S_i(\vec{u})} \frac{1-\chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u})\setminus F_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] \\
&\geq -k\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \sum_{a\in S_i(\vec{u})} \frac{1-\chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] \\
&\geq -k\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \left(\sum_{a\in S_i(\vec{u})} \frac{1}{|S_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] - \sum_{a\in S_i(\vec{u})} \frac{\chi_{F_i(\vec{u})}(a)}{|S_i(\vec{u})|} \right) \\
&\geq -k\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \left(\sum_{a\in S_i(\vec{u})} \frac{1}{|S_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] - \frac{|F_i(\vec{u})|}{|S_i(\vec{u})|} \right) \\
&\geq -k\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \left(\sum_{a\in S_i(\vec{u})} \frac{1}{|S_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] - \beta \right) \\
&= -(k+1)\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u}) \sum_{a\in S_i(\vec{u})} \frac{1}{|S_i(\vec{u})|}\mathbb{E}_{U^{\vec{u}a}}[f^{\vec{u}a}] \\
&\geq -(k+1)\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u})\mathbb{E}_{U^{\vec{u}}}[f^{\vec{u}}] \\
&= -(k+1)\beta + \sum_{\vec{u}\in X_I} \pi_I(\vec{u})\mathbb{E}_U[f] = -(k+1)\beta + \mathbb{E}_U[f]
\end{aligned}
$$

The penultimate equality holds because the function f is independent of the coordinates of I, and therefore, for all $\vec{u} \in X_I$, $\mathbb{E}_{U^{\vec{u}}}[f^{\vec{u}}] = \mathbb{E}_U[f]$.

■

11.2 "Continuity" for DDWB Processes

Lemma 19 *Let π be a distribution on the product space $\prod_{i=1}^t X_i$ given by a DDWB process $\vec{S}$, $\vec{F}$ with covering bound κ. Let c and d be arbitrary. Let $I_0 \subseteq [t]$ so that $|I_0| = d$. Let $\vec{u}, \vec{v} \in \prod_{i=1}^t X_i$ be arbitrary. If for all $i = 1, \dots t$,*

1. *$\pi(\vec{u}) > 0$ and $\pi(\vec{v}) > 0$*
2. *For all $i \in [t] \setminus I_0$, $S_i(u_1, \dots u_{i-1}) = S_i(v_1, \dots v_{i-1})$*
3. *For all $i \in [t] \setminus I_0$, $|F_i(u_1, \dots u_{i-1}) \oplus F_i(v_1, \dots v_{i-1})| \leq (c/t)|X_i|$*

then $\pi(\vec{v}) < \kappa^{-d} e^{c/\kappa} \pi(\vec{u})$.

Proof: Explicit calculation reveals that:

$$
\begin{aligned}
\frac{\pi(\vec{u})}{\pi(\vec{v})} &= \frac{\prod_{i=1}^t \left(\frac{1}{|S_i(u_1,\dots u_{i-1})\setminus F_i(u_1,\dots u_{i-1})|}\right)}{\prod_{i=1}^t \left(\frac{1}{|S_i(v_1,\dots v_{i-1})\setminus F_i(v_1,\dots v_{i-1})|}\right)} = \prod_{i=1}^t \frac{|S_i(v_1,\dots v_{i-1})\setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1})\setminus F_i(u_1,\dots u_{i-1})|} \\
&= \prod_{i\in I_0} \frac{|S_i(v_1,\dots v_{i-1})\setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1})\setminus F_i(u_1,\dots u_{i-1})|} \prod_{i\in[t]\setminus I_0} \frac{|S_i(v_1,\dots v_{i-1})\setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1})\setminus F_i(u_1,\dots u_{i-1})|}
\end{aligned}
$$

$$
\begin{aligned}
&\leq \prod_{i\in I_0} \frac{|X_i|}{\kappa|X_i|} \prod_{i\in[t]\setminus I_0} \frac{|S_i(v_1,\dots v_{i-1}) \setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1}) \setminus F_i(u_1,\dots u_{i-1})|} \\
&= \kappa^{-d} \prod_{i\in[t]\setminus I_0} \frac{|S_i(v_1,\dots v_{i-1}) \setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1}) \setminus F_i(u_1,\dots u_{i-1})|} = \kappa^{-d} \prod_{i\in[t]\setminus I_0} \frac{|S_i(u_1,\dots u_{i-1}) \setminus F_i(v_1,\dots v_{i-1})|}{|S_i(u_1,\dots,u_{i-1}) \setminus F_i(u_1,\dots u_{i-1})|} \\
&\leq \kappa^{-d} \prod_{i\in[t]\setminus I_0} \frac{|S_i(u_1,\dots u_{i-1}) \setminus F_i(u_1,\dots u_{i-1})| + |F_i(v_1,\dots v_{i-1}) \oplus F_i(u_1,\dots u_{i-1})|}{|S_i(u_1,\dots,u_{i-1}) \setminus F_i(u_1,\dots u_{i-1})|} \\
&\leq \kappa^{-d} \prod_{i\in[t]\setminus I_0} \left(1 + \frac{(c/t)|X_i|}{\kappa|X_i|}\right) \leq \kappa^{-d} e^{(t-d)\frac{c}{t\kappa}} \leq \kappa^{-d} e^{\frac{c}{\kappa}}
\end{aligned}
$$

■

12 The Distribution $\mathcal{D}$ is a DDWB Distribution

We give a DDWB process $\vec{S}$, $\vec{F}$ and show that it produces the distribution $\mathcal{D}$ used to generate reduction layouts used in the reduction from set-disjointness to the *FindBadEdge* search lemma. This enables us to use the machinery of DDWB distributions to prove Lemma 12 and Lemma 13.

Definition 12.1 *Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$. Let G, $N_3(\cdot)$, $N_2(\cdot)$ be as in Definition 8.1. We define a DDWB process $\vec{S}$, $\vec{F}$ over the product space $[m]^{n+1} \times \left([2m+1]^2\right)^n \times \left([2m+1]^3\right) \times \left([3m]^3\right)^{n+1}$ as follows:*

1. *When choosing i_k given $i_1, \dots i_{k-1}$: $X_k = [m]$, $S_k = G$ and $F_k(i_1, \dots i_{k-1}) = \{i_1, \dots i_{k-1}\}$.*
2. *When choosing $(j_{k,1}, j_{k,2})$ given $\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{k-1,1}, j_{k-1,2})$ (with $k \leq n$), we have $X_{n+1+k} = [2m+1]^2$, $S_{n+1+k}(\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{k-1,1}, j_{k-1,2})) = N_2(i_k)$, and:*
$$F_{n+1+k}(\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{k-1,1}, j_{k-1,2})) \quad = \quad pm_{[2m+1]}\left(\{j_{1,1}, j_{1,2}, \dots j_{k-1,1}, j_{k-1,2}\}\right)$$
3. *When choosing $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3})$ given $\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{n,1}, j_{n,2})$, we have $X_{2n+2} = [2m+1]^3$, $S_{2n+2}\left(\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{n,1}, j_{n,2})\right) = N_3(i_{n+1})$, and:*
$$F_{2n+2}(\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{n,1}, j_{n,2})) \quad = \quad \{j_{1,1}, j_{1,2}, \dots j_{n,1}, j_{n,2}\}$$
4. *For $k \leq n$, when choosing (u_k, v_k, w_k) given $\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_{k-1}, v_{k-1}, w_{k-1})$, $X_{2n+2+k} = [3m]^3$, $S_{2n+2+k}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_{k-1}, v_{k-1}, w_{k-1})) = \mathcal{K}_{1,2}\left(E_{i_k}[V_{j_{k,1}} \cap V_{j_{k,2}}]\right)$, and*
$$F_{2n+2+k}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_{k-1}, v_{k-1}, w_{k-1})) \quad = \quad tm_{[3m]}(\{u_1, v_1, w_1, \dots u_{k-1}, v_{k-1}, w_{k-1}\})$$
5. *When choosing $(u_{n+1}, v_{n+1}, w_{n+1})$ given $\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_n, v_n, w_n)$, $X_{3n+3} = [3m]^3$, $S_{3n+3}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_{k-1}, v_{k-1}, w_{k-1})) = \mathcal{K}_{1,2}\left(E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]\right)$, and*
$$F_{3n+3}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \dots (u_n, v_n, w_n)) \quad = \quad tm_{[3m]}(\{u_1, v_1, w_1, \dots u_n, v_n, w_n\})$$

Lemma 20 *Let $m \geq 450/\delta^2$. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$. Let $\delta = \delta(\mathcal{V}_I, \mathcal{V}_{II})$ and let $\gamma = \frac{n+1}{m}$. The distribution $\mathcal{D}(\mathcal{V}_I, \mathcal{V}_{II})$ is generated by the DDWB process $\vec{S}$, $\vec{F}$ over the product space $[m]^{n+1} \times \left([2m+1]^2\right)^n \times ([2m+1]^3) \times \left([3m]^3\right)^{n+1}$. Moreover, this process has blockage bound $\leq 30\gamma/\delta^2$ and it has covering bound $\geq \min\{\delta^2/10 - 3\gamma, \delta/3 - 3\gamma, \delta/12 - \gamma\}$.*

Proof: That the DDWB process $\vec{S}$, $\vec{F}$ generates the distribution $\mathcal{D}$ follows immediately by comparing the above functions with the experiment of Definition 8.2. The covering bounds follow immediately from Lemma 11, and the blockage bounds are implicit in those calculations. ■

Corollary 21 *If $\gamma = \delta^2/60$, then the covering bound of the process is $\geq \delta^2/20$, ie. $\kappa \geq \delta^2/20$.*

Now we use Lemma 19 to prove the continuity lemma:

Proof:(of the continuity lemma, Lemma 13) Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ and $L^* = (\vec{\imath}^*, \vec{\jmath}^*, \vec{u}^*, \vec{v}^*, \vec{w}^*)$ be two reduction layouts from $\mathcal{L}^p$ with $HD(L, L^*) \leq d$. Let $\vec{S}$ and $\vec{F}$ be the DDWB process for generating the distribution $\mathcal{D}^p$ as described in Definition 12.1. For the sake of brevity, in the scope of this proof we will write $S_i(L)$ and $S_i(L^*)$ instead of with their proper arguments, eg. $S_{2n+2+k}(L)$ instead of $S_{2n+2+k}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \ldots (u_{k-1}, v_{k-1}, w_{k-1}))$. We do the same with the F_i's. We set I_0 to be the set of indices i so that $S_i(L) \neq S_i(L^*)$. Checking against the definitions of $\vec{S}$, $\vec{F}$, it is easily checked by a case-analysis that $|I_0| \leq 2d$. We place this argument in the Appendix, Section D, as Lemma 22.

We now check that the hypotheses of Lemma 19 are met with the process $\vec{S}$, $\vec{F}$ over $[m]^{n+1} \times \left([2m+1]^2\right)^n \times ([2m+1]^3) \times \left([3m]^3\right)^{n+1}$, with $t = 3n+3$, with $\pi = \mu$, with I_0 as above, and with $\vec{u} = L^*$, $\vec{v} = L$ By Lemma 20 and Corollary 21, the DDWB process generating μ has $\kappa \geq \delta^2/20$ where $\delta = \delta(\mathcal{V}_I, \mathcal{V}_{II})$ and $\gamma = \frac{n+1}{m} \leq \delta^2/60$.

Property 1: $\mu(L) > 0$ and $\mu(L^*) > 0$. This is satisfied because $L \in \mathcal{L}$, and $L^* \in \mathcal{L}$.

Property 2: The set I_0 is defined to be the set of i with $S_i(L) \neq S_i(L^*)$.

Property 3: In the Appendix, Section D, we show that for all $i \in [t]$, $|F_i(L) \oplus F_i(L^*)| \leq (9d\gamma/(3n+3))|X_i|$.

By Lemma 19:

$$\mu(L) \leq \kappa^{-2d} e^{9d\gamma/\kappa} \mu(L^*) \leq (\delta^2/20)^{-2d} e^{9d(\delta^2/60)/(\delta^2/20)} \mu(L) = (20/\delta^2)^{2d} e^{3d} \mu(L) = (20/\delta^2)^{2d} e^{3d} \mu(L)$$

■

Now we use Lemma 18 to prove the completeness lemma:

Proof:(of the completeness lemma, Lemma 12) Fix m, and let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$ so with $\delta = \delta(\mathcal{V}_I, \mathcal{V}_{II})$. Let n be given so that $n \leq \delta^{10}/(2^{10} \cdot 3 \cdot 5^2)m$. Let $l \in [n]$ be given. Let U be uniform distribution on $[m]^{n+1} \times \left([2m+1]^2\right)^n \times ([2m+1]^3) \times \left([3m]^3\right)^{n+1}$. Let μ be the mass function for the distribution $\mathcal{D}$. Set β to be the blockage bound for the DDWB process generating $\mathcal{D}$. Let $\mathcal{A} \subseteq [m]^{n+1} \times \left([2m+1]^2\right)^n \times ([2m+1]^3) \times \left([3m]^3\right)^{n+1}$ be the event that $(j_{n+2,1}, j_{l,1}, j_{l,2}) \in N_3(i_l)$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_{n+1})$, and $K(\{u_{n+1}, u_l\}, \{v_{n+1}, v_l, w_{n+1}, w_l\}) \subseteq E_{i_l}\left[V_{j_{l,1}} \cap V_{j_{l,2}}\right] \cap E_{i_{n+1}}\left[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}\right]$. Notice that $\mathcal{S}^l = \mathcal{L} \cap \mathcal{A}$, and that because $\mu(\mathcal{L}) = 1$, $\mu(\mathcal{S}^l) = \mu(\mathcal{L} \cap \mathcal{A}) = \mu(\mathcal{A})$.

Let I denote the indices $1, \ldots 2n+2$ (so that, using our abused notation, the coordinates of I correspond to $\vec{\imath}, \vec{\jmath}$). Let $A \subseteq [m]^{n+1} \times \left([2m+1]^2\right)^n \times ([2m+1]^3)$ be the event that $i_l, i_{n+1} \in G$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_{n+1})$, and $(j_{n+1,2}, j_{l,1}, j_{l,2}) \in N_3(i_l)$. Notice that $A \supseteq \mathcal{A}_I$ and therefore $\mu(\mathcal{A}) = \sum_{\vec{\imath}, \vec{\jmath} \in A} \mu_I(\vec{\imath}, \vec{\jmath}) \mu^{\vec{\imath}, \vec{\jmath}}(\mathcal{A}(\vec{\imath}, \vec{\jmath}))$.

For each setting of $\vec{\imath}$ and $\vec{\jmath}$, the event $\mathcal{A}(\vec{\imath}, \vec{\jmath})$ depends only on the values of $(u_{n+1}, v_{n+1}, w_{n+1})$, and (u_l, v_l, w_l). Moreover, in the event that $\mathcal{A}$ holds, we have that $(u_l, v_l, w_l) \in \mathcal{K}_{1,2}(E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}])$

and $(u_{n+1}, v_{n+1}, w_{n+1}) \in \mathcal{K}_{1,2}(E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}])$. Therefore we can apply Lemma 18 and conclude for all $\vec{\imath}$, $\vec{\jmath}$: $\mu^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) \geq U^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) - 2\beta$.

For each $\vec{\imath}$ and $\vec{\jmath}$ set $D(\vec{\imath},\vec{\jmath}) = |E_{i_l}\left[V_{j_{l,1}} \cap V_{j_{l,2}}\right] \cap E_{i_{n+1}}\left[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}\right]|/\binom{3m}{2}$. Notice that $\delta(\mathcal{V}_I, \mathcal{V}_{II})$ is the expectation of D over the uniform distribution on $[m]^2 \times [2m+1]^5$. Because the marginal distribution of $U^{\vec{\imath}\vec{\jmath}}$ on (u_l, v_l, w_l) and $(u_{n+1}, v_{n+1}, w_{n+1})$ is just the uniform distribution $[3m]^3 \times [3m]^3$, we can apply Lemma 17: For each choice of $\vec{\imath},\vec{\jmath}$ we have that $U^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) \geq D(\vec{\imath},\vec{\jmath})^8 - (23/3m)$. Therefore:

$$\begin{aligned}
\mu(\mathcal{A}) &= \sum_{\vec{\imath}\vec{\jmath} \in A} \mu_I(\vec{\imath},\vec{\jmath})\mu^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) \geq \sum_{\vec{\imath},\vec{\jmath} \in A} \mu_I(\vec{\imath},\vec{\jmath})\left(U^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) - 2\beta\right) \geq -2\beta + \sum_{\vec{\imath},\vec{\jmath} \in A} \mu_I(\vec{\imath},\vec{\jmath})U^{\vec{\imath},\vec{\jmath}}(\mathcal{A}(\vec{\imath},\vec{\jmath})) \\
&\geq -2\beta + \sum_{\vec{\imath},\vec{\jmath} \in A} \mu_I(\vec{\imath},\vec{\jmath})(D(\vec{\imath},\vec{\jmath})^8 - (23/3m)) \geq -2\beta - (23/3m) + \sum_{\vec{\imath},\vec{\jmath}} \mu_I(\vec{\imath},\vec{\jmath})(D(\vec{\imath},\vec{\jmath}) \cdot \chi_A(\vec{\imath},\vec{\jmath}))^8 \\
&\geq -2\beta - (23/3m) + \left(\sum_{\vec{\imath},\vec{\jmath}} \mu_I(\vec{\imath},\vec{\jmath})D(\vec{\imath},\vec{\jmath})\chi_A(\vec{\imath},\vec{\jmath})\right)^8 = -2\beta - (23/3m) + (\mathbb{E}_{\mu_I}[D \cdot \chi_A])^8
\end{aligned}$$

The final task is to get a lower bound for $\mathbb{E}_{\mu_I}[D \cdot \chi_A]$. This will follow from an application of Lemma 18. Let U denote the uniform distribution over $\vec{\imath},\vec{\jmath}$, In the Appendix, Section D, Lemma 23, it is shown that: $\mathbb{E}_U[D \cdot \chi_A] \geq \delta(\mathcal{V}_I, \mathcal{V}_{II})/2$. Notice that the function $D \cdot \chi_A$ depends only upon 4 coordinates: i_l, i_{n+1}, the triple $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3})$ and the pair $(j_{l,1}, j_{l,2})$. Moreover, whenever $D \cdot \chi_A > 0$, we have that $i_l \in G$, $i_{n+1} \in G$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_{n+1})$, and $(j_{l,1}, j_{l,2}) \in N_2(i_l)$, so we may apply Lemma 18 to conclude that $\mathbb{E}_{\mu_I}[\delta \cdot \chi_A] \geq \delta/2 - 4\beta$. Therefore:

$$\mu(\mathcal{A}) \geq -2\beta - (23/3m) + (\mathbb{E}_{\mu_I}[D \cdot \chi_A])^8 \geq -2\beta - (23/3m) + (\delta/2 - 4\beta)^8$$

Because $m \geq 31(2/\delta)^8$, $\frac{23}{3m} \leq 0.25(\delta/2)^8$. By Lemma 20, $\beta \leq 30\gamma/\delta^2 \leq 30(\delta^{10}/(2^{10} \cdot 3 \cdot 5^2))/\delta^2 = \delta^8/(2^9 \cdot 5)$, therefore:

$$\begin{aligned}
\mu(\mathcal{A}) &\geq -\frac{2\delta^8}{2^9 \cdot 5} - 0.25\left(\frac{\delta}{2}\right)^8 + \left(\frac{\delta}{2} - \frac{4\delta^8}{2^9 \cdot 5}\right)^8 \\
&> -0.2\left(\frac{\delta}{2}\right)^8 - 0.25\left(\frac{\delta}{2}\right)^8 + \left(\frac{\delta}{2}\left(1 - \frac{1}{2^6 \cdot 5}\right)\right)^8 \\
&> -0.45\left(\frac{\delta}{2}\right)^8 + 0.97\left(\frac{\delta}{2}\right)^8 > \frac{\delta^8}{2^9}
\end{aligned}$$

■

13 Debriefing

After digesting the proof of Theorem 8, the reader might notice that there was some overkill in a few of the arguments, and wonder if a tighter argument could improve the constants of Theorem 8. This seems likely, however, it was decided that optimizing between different values of "astronomical" was not worth the added length.

There are two points in the argument particularly worthy of mention. The first is that Definition 9.3 is bit stronger than is needed to prove Lemma 14, and it may be possible with a more careful definition to reduce the exponent of 8 (which comes from trying to randomly find a $K_{2,4}$ in a graph of edge density α) to something smaller, like 4 or 6. This would clearly improve the bound in Lemma 12. Furthermore, it might also allow a slackening of the definition of partition density, Definition 5.2, so that a larger value is guaranteed by an analog to Lemma 5. Furthermore, the DDWB machinery introduces a fair a amount of slop because the blockage bounds (coverage bounds) are taken as a maximum (minimum) over all coordinates, whereas a more careful coordinate-wise analysis of the particular transformation of Lemma 14 would improve the constants seen in Lemma 12 and Lemma 13. Of course, this would likely be a more lengthy analysis.

References

[1] A. Aguirre and M. Vardi. Random 3-SAT and BDDs: The plot thickens further. In *Principles and Practice of Constraint Programming*, pages 121–136, 2001.

[2] F. Aloul, M. Mneimneh, and K. Sakallah. ZBDD-based backtrack search SAT solver. In *Eleventh IEEE/ACM Workshop on Logic & Synthesis*, pages 131–136, 2002.

[3] A. Atserias, P. Kolaitis, and M. Vardi. Constraint propagation as a proof system. In *Tenth International Conference on Principles and Practice of Constraint Programming*, pages 77–91, 2004.

[4] P. Beame, R. Karp, T. Pitassi, and M. Saks. The efficiency of resolution and Davis–Putnam procedures. *SIAM Journal on Computing*, 31(4):1048–1075, August 2002. Preliminary versions in FOCS 1996 and STOC 1998.

[5] E. Ben-Sasson and A. Wigderson. Short proofs are narrow — resolution made simple. *Journal of the ACM*, 48(2):149–169, 2001.

[6] R. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Transactions on Computers*, C-35(8):677–691, 1986.

[7] R. Bryant. Symbolic boolean manipulation with ordered binary decision diagrams. *ACM Computing Surveys*, 24(3):293–318, 1992.

[8] J. L. Burch, E. M. Clarke, and D. E. Long. Representing circuits more efficiently in symbolic model checking. In *Proceedings of the Twenty-eight ACM/IEEE Design Automation Conference*, pages 403–407, 1991.

[9] J. L. Burch, E. M. Clarke, and D. E. Long. Symbolic model checking with partitioned transition relations. In *Proceedings fo the 1991 International Conference on VLSI*, pages 49–58, 1991.

[10] P. Chatalic and L. Simon. Multi-resolution on compressed sets of clauses. In *Proceedings of the Twelfth International Conference on Tools with Artificial Intelligence*, pages 2–10, 2000.

[11] P. Chatalic and L. Simon. Zres: The old Davis-Putnam procedures meets ZBDDs. In *Proceedings of the Seventeenth International Conference on Automated Deduction*, pages 449–454, 2000.

[12] V. Chvátal and E. Szemerédi. Many hard examples for resolution. *Journal of the ACM*, 35(4):759–768, October 1988.

[13] E. Clarke, O. Grumberg, and D. Peled. *Model Checking.* MIT Press, 1999.

[14] C. Coarfa, D. Demopoulos, A. S. M. Aguirre, D. Subramanian, and M. Vardi. Random 3-SAT: The plot thickens. *Constraints*, 8(3):243–261, 2003.

[15] S. Cook and A. Reckhow. The relative efficiency of propositional proof systems. *Journal of Symbolic Logic*, 44(1):36–50, 1979.

[16] N. Eén and N. Sörensson. An extensible SAT-solver. In *Proceedings of SAT 2003*, 2003.

[17] E. Goldberg and Y. Novikov. Berkmin: a fast and robust SAT solver. In *Proceedings of DATE 2002*, 2002.

[18] J. F. Groote. Hiding propositional constants in BDDs. *Formal Methods in System Design: an International Journal*, 8(1):91–96, 1996.

[19] J. F. Groote. Personal communication, 2006.

[20] J. F. Groote and H. Zantema. Resolution and binary decision diagrams cannot simulate each other polynomially. *Discrete Applied Mathematics*, 130(2), 2003.

[21] A. Haken. The intractability of resolution. *Theoretical Computer Science*, 39(2-3):297–308, August 1985.

[22] J. Huang and A. Darwiche. Toward good elimination ordering for symbolic SAT solving. In *Proceedings of the Sixteenth IEEE Conference on Tools with Artificial Intelligence*, pages 566–573, 2004.

[23] R. Impagliazzo, T. Pitassi, and A. Urquhart. Upper and lower bounds for tree-like cutting planes proofs. In *Ninth Annual Symposium on Logic in Computer Science*, pages 220–228, 1994.

[24] T. Jussila, C. Sinz, and A. Biere. Extended resolution proofs for symbolic SAT solving with quantification. In *Proceedings of the Ninth International Conference on Theory and Applications of Satisfiability Testing*, pages 54–60, 2006.

[25] B. Kalyanasundaram and G. Schnitger. The probabilistic communication complexity of set intersection. *SIAM Journal of Discrete Mathematics*, 5(4):545–557, 1992.

[26] J. Krajíček. *Bounded Arithmetic, Propositional Logic, and Complexity Theory.* Cambridge University Press, 1995.

[27] J. Krajíček. Implicit proofs. *Journal of Symbolic Logic*, 69(2), 2004.

[28] E. Kushilevitz and N. Nisan. *Communication Complexity.* Cambridge University Press, 1997.

[29] J. Marques-Silva and K. Sakallah. GRASP a new search algorithm for satisfiability. In *Proceedings of IEEE/ACM Internation Conference on Computer-Aided Design*, 1996.

[30] K. McMillan. *Symbolic Model Checking.* PhD thesis, Carnegie Mellon, 1992.

[31] C. Meinel and T. Theobald. *Algorithms and Data Structures in VLSI Design.* Springer-Verlag, 1998.

[32] M. Moskewicz, C. Madigan, Y. Zhao, L. Zhang, and S. Malik. Chaff: Engineering an efficient SAT solver. In *Proceedings of 2001 Design Automation Conference*, 2001.

[33] D. Motter and I. Markov. A compressed breadth-first search for satisfiability. In *Fourth International Workshop on Algorithms Engineering and Experiments (ALENEX)*, pages 29–42, 2002.

[34] D. Motter and I. Markov. Overcoming resolution-based lower bounds for SAT solvers. In *Eleventh IEEE/ACM Workshop on Logic and Synthesis*, pages 373–378, 2002.

[35] D. Motter, J. Roy, and I. Markov. Resolution cannot polynomially simulate compressed-BFS. *Annals of Mathematics and Artificial Intelligence*, 44(1–2):121–156, 2005.

[36] G. Pan and M. Vardi. Search vs. symbolic techniques in satisfiability solving. In *The Seventh International Conference on Theory and Applications of Satisfiability Testing*, 2004.

[37] R. Raz. Resolution lower bounds for the weak pigeonhole principle. In *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, pages 553–562, 2002.

[38] R. Raz and A. Wigderson. Monotone circuits for matching require linear depth. *Journal of the ACM*, 39(3):736–744, 1992.

[39] A. A. Razborov. On the distributional complexity of disjointness. *Theoretical Computer Science*, 106(2):385–390, 1992.

[40] J. Alan Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, January 1965.

[41] C. Sinz and A. Biere. Extended resolution proofs for conjoining BDDs. In *First International Computer Science Symposium in Russia*, pages 600–611, 2006.

[42] T. Uribe and M. Stickel. Ordered binary decision diagrams and the Davis-Putnam procedure. In *Proceedings of the First International Conference on Constraints in Computational Logics*, pages 34–49, 1994.

[43] A. Urquhart. Hard examples for resolution. *Journal of the ACM*, 34(1):209–219, 1987.

A Proofs and Calculations for Section 2

Proof:(of Lemma 16) A standard application of the convexity of the function $x \mapsto x^k$. For each $x \in X$, let $d_x = |\{i \in [n] \mid x \in Y_i\}|$. Set $\bar{d_x} = \frac{1}{|X|}\sum_{x\in X} d_x$. We have that $\bar{d_x} = \frac{1}{|X|}\sum_{x\in X} d_x = \frac{1}{|X|}\sum_{i=1}^{n} |Y_i| = \alpha n$, and therefore by Jensen's Inequality:

$$\frac{1}{n^k}\sum_{\vec{i}\in[n]^k} |\bigcap_{l=1}^{k} Y_{i_l}| = \frac{1}{n^k}\sum_{x\in X} d_x^k \geq \frac{1}{n^k}|X|\left(\bar{d_x}\right)^k \geq \frac{1}{n^k}|X|\left(\alpha n\right)^k = \alpha^k |X|$$

∎

Proof:(of Lemma 17)

1. Conditioned on the choice of u_1, the probability that $\{u_1, u_2\} \in E$ and $\{u_1, u_3\} \in E$ is $\left(\frac{d_{u_1}}{N}\right)^2$. Because $\frac{1}{N}\sum_u d_u = \frac{1}{N}2\alpha\binom{N}{2} = \alpha(N-1)$, convexity shows that the probability that $\{u_1, u_2\} \in E$ and $\{u_1, u_3\} \in E$ is at least $N^{-3} \cdot N(\alpha(N-1))^2 = \alpha^2(1 - 2/N + 1/N^3)$. We now subtract out the probability that u_1, u_2, u_3 are not all distinct, which is clearly no more than $3/N$, and we obtain the stated bound.

2. For each u_1 and u_2, let $D(u_1, u_2)$ be the number of common neighbors of u_1 and u_2. Because the average degree of $u \in V$ is $\alpha(N-1)$, Lemma 16 shows that $\frac{1}{N^2}\sum_{\vec{u}\in V^2} D(u_1, u_2) \geq \alpha^2((N-1)/N)^2(N-1) \geq \alpha^2(1-2/N)$. Conditioned on the choice of u_1, u_2, the probability that all edges are present is clearly $(D(u_1, u_2)/N)^4$. Apply Jensen's Inequality and we have that the probability that all edges are present is at least $\left(\alpha^2(1-2/N)\right)^4 = \alpha^8(1-2/N)^4 \geq \alpha^8(1-8/N)$. We now subtract out the probability that $u_1, u_2, u_3, u_4, u_5, u_6$ are not all distinct, which is clearly no more than $\binom{6}{2}/N = 15/N$, and we obtain the stated bound.

■

B Proofs and Calculations for Section 8

Proof:(of Lemma 9) Let $\delta = \delta(\mathcal{V}_I, \mathcal{V}_{II})$. Notice that when $m \geq 3\delta \geq ((6/\delta) - 1)/2$, we have that $3/(2m+1) \leq \delta/2$. By Definition 5.2 , we have that

$$\frac{1}{m^2(2m+1)^5}\sum_{\vec{\imath}\in[m]^2}\sum_{\vec{\jmath}\in[2m+1]^5} |\bigcap_{k=1}^{5}(E_{i_1}[V_{j_k}] \cap E_{i_2}[V_{j_k}])| = \delta\binom{3m}{2}$$

And therefore $\frac{1}{m(2m+1)^3}\sum_{i\in[m]}\sum_{\vec{\jmath}\in[2m+1]^3} |E_i[V_{j_1} \cap V_{j_2} \cap V_{j_3}]| \geq \delta\binom{3m}{2}$. Because the number of terms with $j_1 = j_2$, $j_2 = j_3$ or $j_1 = j_3$ is at most $3m(2m+1)^2$, such terms can contribute at most $\frac{1}{m(2m+1)^3}3m(2m+1)^2\binom{3m}{2} = \frac{3}{2m+1}\binom{3m}{2}$ to this sum, so we have:

$$\frac{1}{m(2m+1)^3}\sum_{i\in[m]}\sum_{\substack{\vec{\jmath}\in[2m+1]^3\\ \vec{\jmath}\text{ distinct}}} |E_i[V_{j_1} \cap V_{j_2} \cap V_{j_3}]| \geq (\delta - 3/(2m+1))\binom{3m}{2} \geq (\delta/2)\binom{3m}{2}$$

Combining this with the fact that for each $i \in [m]$, $|E_i| \leq \binom{3m}{2}$, by averaging, we have that with probability at least $\delta/6$ over the choice of i,j_1,j_2,j_3, with j_1, j_2, j_3 all distinct, that $|E_i[V_{j_1} \cap V_{j_2} \cap V_{j_3}]| \geq (\delta/3)\binom{3m}{2}$. Therefore, with probability at least $\delta/12$ over choices of i, there are at least $(\delta/12)[2m+1]^3$ many triples j_1, j_2, j_3 that are distinct and have $|E_i[V_{j_1} \cap V_{j_2} \cap V_{j_3}]| \geq (\delta/3)\binom{3m}{2}$. Therefore, $|G| \geq (\delta/12)m$.

■

Proof:(of Lemma 10) Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ be a reduction layout, and let $X_1, \ldots X_n, Y_1, \ldots Y_n$ be a set intersection instance. Let e be a bad edge of $A_{L,\vec{X},\vec{Y}}$. We recall two useful definitions for the proof of this lemma: From Definition 8.2, the planted edge under $\vec{X}, \vec{Y}, L$ is defined as $pe\left(\vec{X}, \vec{Y}, L\right) = \{u_{n+1}, w_{n+1}\}$. From Definition 8.3, the assignment $A_{L,\vec{X},\vec{Y}}$ is defined as follows: We set $I = \{i_1, \ldots i_{n+1}\}$, set $J = \{j_{1,1}, j_{1,2}, \ldots j_{n,1}, j_{n,2}, j_{n+1,1}, j_{n+1,2}, j_{n+1,3}\}$, and set

$V = \{u_1, \ldots u_{n+1}, v_1, \ldots v_{n+1}, w_1, \ldots w_{n+1}\}$. We set β, $\beta(L)$ to be the lexicographically first assignment to the variables $\{x_e^i \mid i \in [m] - I,\ e \in [3m - V]^2\} \cup \{y_u^j \mid j \in [2m+1] - J,\ u \in [3m] - V\}$, so that β defines a matching of size $m - n - 1$ and an independent set of size $2(m - n - 1)$.

$$A_{L,\vec{X},\vec{Y}}(x_e^i) = \begin{cases} \beta(x_e^i) & \text{if } i \in [m] - I \text{ and } e \in ([3m] - V)^2 \\ X_k & \text{if } i = i_k \text{ and } e = \{u_k, v_k\} \text{ for some } k \in [n] \\ \neg X_k & \text{if } i = i_k \text{ and } e = \{u_k, w_k\} \text{ for some } k \in [n] \\ 1 & \text{if } i = i_{n+1} \text{ and } e = \{u_{n+1}, w_{n+1}\} \\ 0 & \text{otherwise} \end{cases}$$

$$A_{L,\vec{X},\vec{Y}}(y_x^j) = \begin{cases} \beta(y_x^j) & \text{if } j \in [2m+1] - j \text{ and } u \in [3m] - V \\ 1 & \text{if } j = j_{k,1} \text{ and } x = v_k \text{ for some } k \in [n] \\ Y_k & \text{if } j = j_{k,2} \text{ and } x = u_k \text{ for some } k \in [n] \\ \neg Y_k & \text{if } j = j_{k,2} \text{ and } x = w_k \text{ for some } k \in [n] \\ 1 & \text{if } j = j_{n+1,1} \text{ and } x = u_{n+1} \\ 1 & \text{if } j = j_{n+1,2} \text{ and } x = v_{n+1} \\ 1 & \text{if } j = j_{n+1,3} \text{ and } x = w_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

Let e be a bad edge for the assignment $A_{L,\vec{X},\vec{Y}}$. First of all, because β sets no bad edges, $e \cap V \neq \emptyset$. Furthermore, for all e with $|e \cap V| = 1$ have $A_{L,\vec{X},\vec{Y}}(x_e^i) = 0$ for all i, so $e \subseteq V$. Finally, for $e \subseteq V$, with some $A_{L,\vec{X},\vec{Y}}(x_e^i) = 1$, we have that for some $k \in [n]$, $e = \{u_k, v_k\}$ or $e = \{u_k, w_k\}$. Choose k so that $e = \{u_k, v_k\}$ or $e = \{u_k, w_k\}$. If $k = n+1$ then we must have that $e = \{u_{n+1}, w_{n+1}\}$, and e the bad edge, so consider the case when $k \leq n$.

Notice that for all $i' \neq i_k$, $A_{L,\vec{X},\vec{Y}}(x_e^{i'}) = 0$. On the other hand, e is a bad edge, so there is some x_e^i that gets set to 1, therefore $A_{L,\vec{X},\vec{Y}}(x_e^{i_k}) = 1$.

We now rule out the case that $e = \{u_k, w_k\}$. Because $A_{L,\vec{X},\vec{Y}}(x_e^{i_k}) = 1$, we have by construction that $X_k = 0$. Because e is bad, for some j, j', $A_{L,\vec{X},\vec{Y}}(y_{u_k}^j) = 1$ and $A_{L,\vec{X},\vec{Y}}(y_{w_k}^{j'}) = 1$. However, $y_{u_k}^j$ and $y_{w_k}^{j'}$ cannot both be set to 1.

Suppose that $e = \{u_k, v_k\}$. Because $A_{L,\vec{X},\vec{Y}}(x_e^{i_l}) = 1$, we have by construction that $X_l = 1$. If $(X_l, Y_l) = (1, 1)$, then the lemma holds. Otherwise, $Y_l = 0$. But in this case, we have that for all j, $A_{L,\vec{X},\vec{Y}}(y_{u_l}^j) = 0$, contradiction to e being a bad edge.

∎

C Proofs and Calculations for Section 9

Proof:(of Lemma 11) For each $k = 1, \ldots n$, as we choose $(j_{k,1}, j_{k,2})$ (and $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3})$), $|J| \leq 2n < 2(n+1) = 2\gamma m$ and as we choose each (u_k, v_k, w_k), $|V^*| \leq 3n < 3(n+1) = 3\gamma m$.

1. By Lemma 9, $|G| \geq (\delta/12)m$. On the other hand, $|\{i_1, \ldots i_{k-1}\}| \leq n < \gamma m$. Therefore, $|G \setminus \{i_1, \ldots i_{k-1}\}| > ((\delta/12) - \gamma)m$.

2. Because $|J| \leq 2n$, we have that $pm_{[2m+1]}(J) \leq 2n(2m+1) + (2m+1)2n < 2(2\gamma m)(2m+1) = 2(2\gamma m)(2m+1) = 2\gamma(2m)(2m+1) < 2\gamma(2m+1)^2$. Combining this with the fact that $i_k \in G$

and therefore $|N_2(i_k)| \geq |N_3(i_k)| \geq (\delta/3)(2m+1)^2$ we have that $|N_2(i_k) \setminus pm_{[2m+1]}(J)| \geq ((\delta/3) - 2\gamma)(2m+1)^2$.

3. Because $|J| \leq 2n$ we have that $tm_{[2m+1]}(J) \leq 3(2n)(2m+1)^2 < 3(2\gamma m)(2m+1)^2 = 3\gamma(2m)(2m+1)^2 < 3\gamma(2m+1)^3$. Because $i_p \in G$, $|N_3(i_p)| \geq (\delta/3)(2m+1)^3$. Therefore: $|N_3(i_p) \setminus tm_{[2m+1]}(J)| \geq ((\delta/3) - 3\gamma)(2m+1)^3$.

4. Because $|V^*| \leq 3n$, $|tm(V^*)| \leq 3(3n)(3m)^2 < 3(3\gamma m)(3m)^2 = 3\gamma(3m)^3$. We now get a lower bound on the size of $\mathcal{K}_{1,2}\left(E_{i_k}[V_{j_{k,1}} \cap V_{j_{k,2}}]\right)$: First, because $(j_{k,1}, j_{k,2}) \in N_2(i_k)$, there exists some j' with $|E_i[V_{j'} \cap V_{j_{k,1}} \cap V_{j_{k,2}}]| \geq (\delta/3)\binom{3m}{2}$, so we have that $|E_{i_k}\left[V_{j_{k,1}} \cap V_{j_{k,2}}\right] \geq (\delta/3)\binom{3m}{2}$. Feeding this lowerbound on the edge density into Lemma 17, we have that:

$$|\mathcal{K}_{1,2}(E_{i_k}\left[V_{j_{k,1}} \cap V_{j_{k,2}}\right])| \geq \left(\delta^2/9 - (5/m)\right) \cdot (3m)^3$$

Combining the upper bound on $|tm(V^*)|$ and with the preceding lower bound:

$$|\mathcal{K}_{1,2}(E_{i_k}\left[V_{j_{k,1}} \cap V_{j_{k,2}}\right]) \setminus tm(V^*)| \leq \left((\delta^2/9) - (5/m) - 3\gamma\right)(3m)^3$$

Because $m \geq 450/\delta^2$, we have that $5/m \leq \delta^2/90$ and therefore the above quantity is $\geq (\delta^2/9 - \delta^2/90 - 3\gamma)(3m)^3 = (\delta^2/10) - 3\gamma)(3m)^3$.

5. This derivation is identical to the previous, except that it uses the lower bound of $|E_i[V_{j_{p,3}} \cap V_{j_{p,1}} \cap V_{j_{p,2}}]| \geq (\delta/3)\binom{3m}{2}$ that holds because $(j_{p,1}, j_{p,2}, j_{p,3}) \in N_3(i_p)$.

■

Proof:(details for Lemma 14)

The proof that f is an involution. Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ be a reduction layout, and let $(\vec{\imath}^*, \vec{\jmath}^*, \vec{u}^*, \vec{v}^*, \vec{w}^*) = f(L)$, and let $(\vec{\imath}^{**}, \vec{\jmath}^{**}, \vec{u}^{**}, \vec{v}^{**}, \vec{w}^{**}) = f(f(L))$. Applying the definitions shows that:

$$i_k^{**} = \begin{cases} i_{n+1}^* = i_l & \text{if } k = l \\ i_l^* = i_{n+1} & \text{if } k = n+1 \\ i_k^* = i_k & \text{otherwise} \end{cases} \qquad u_k^{**} = \begin{cases} u_l^* = u_{n+1} & \text{if } k = n+1 \\ u_{n+1}^* = u_l & \text{if } k = l \\ u_k^* = u_k & \text{otherwise} \end{cases}$$

$$n \quad j_{k,1}^{**} = \begin{cases} j_{n+1,3}^* = j_{l,1} & \text{if } k = l \\ j_{l,2}^* = j_{n+1} & \text{if } k = n+1 \\ j_{k,1}^* = j_{k,1} & \text{otherwise} \end{cases} \qquad v_k^{**} = \begin{cases} w_{n+1}^* = v_l & \text{if } k = l \\ v_k^* = v_k & \text{otherwise} \end{cases}$$

$$j_{k,2}^{**} = \begin{cases} j_{n+1,1}^* = j_{l,2} & \text{if } k = l \\ j_{k,2}^* = j_{k,2} & \text{otherwise} \end{cases} \qquad w_k^{**} = \begin{cases} v_l^* = w_{n+1} & \text{if } i = n+1 \\ w_k^* = w_k & \text{otherwise} \end{cases}$$

$$j_{n+1,3}^{**} = j_{l,1}^* = j_{n+1,3}$$

The proof that $A_{f(L),\vec{X},\vec{Y}} = A_{L,\vec{X},\vec{Y}}$ We expand the definitions of $A_{L,\vec{X},\vec{Y}}$ and $A_{f(L),\vec{X},\vec{Y}}$, per definition 8.3 Notice that $\{i_1, \ldots i_{n+1}\} = \{i_1^*, \ldots i_{n+1}^*\}$, $\{j_{1,1}, j_{1,2}, \ldots j_{n,1}, j_{n,2}, j_{n+1,1}, j_{n+1,2}, j_{n+1,3},\} = \{j_{1,1}^*, j_{1,2}^*, \ldots j_{n,1}^*, j_{n,2}^*, j_{n+1,1}^*, j_{n+1,2}^*, j_{n+1,3}^*,\}$, and $\{u_1, \ldots u_{n+1}, v_1, \ldots v_{n+1}, w_1, \ldots w_{n+1}\} = \{u_1^*, \ldots u_{n+1}^*, v_1^*, \ldots v_{n+1}^*, w_1^*, \ldots w_{n+1}^*\}$. Let I, J, and V respectively denote these three sets. Because $\beta(L)$ and $\beta(L^*)$ are both the lexicographically first assignment to the variables

$$\{x_e^i \mid i \in [m] - I,\ e \in [3m - V]^2\} \cup \{y_u^j \mid j \in [2m+1] - J,\ u \in [3m] - V\}$$

so that β defines a matching of size $m-n-1$ and an independent set of size $2(m-n-1)$, we have that $\beta(L)=\beta(L^*)$. Write β for this assignment. We compare $A_{L,\vec{X},\vec{Y}}$ and $A_{f(L),\vec{X},\vec{Y}}$ directly:

$$A_{L,\vec{X},\vec{Y}}(x_e^i) = \begin{cases} \beta(x_e^i) & \text{if } i \in [m]-I \text{ and } e \in ([3m]-V)^2 \\ X_k & \text{if } i = i_k \text{ and } e = \{u_k, v_k\} \text{ for some } k \in [n] \setminus \{l\} \\ \neg X_k & \text{if } i = i_k \text{ and } e = \{u_k, w_k\} \text{ for some } k \in [n] \setminus \{l\} \\ 1(=X_l) & \text{if } i = i_l \text{ and } e = \{u_l, v_l\} \\ 0(=\neg X_l) & \text{if } i = i_l \text{ and } e = \{u_l, w_l\} \\ 1 & \text{if } i = i_{n+1} \text{ and } e = \{u_{n+1}, w_{n+1}\} \\ 0 & \text{otherwise} \end{cases}$$

$$A_{f(L),\vec{X},\vec{Y}}(x_e^i) = \begin{cases} \beta(x_e^i) & \text{if } i \in [m]-I \text{ and } e \in ([3m]-V)^2 \\ X_k & \text{if } i = i_k \text{ and } e = \{u_k, v_k\} \text{ for some } k \in [n] \setminus \{l\} \\ \neg X_k & \text{if } i = i_k \text{ and } e = \{u_k, w_k\} \text{ for some } k \in [n] \setminus \{l\} \\ 1 & \text{if } i = i_l (= i^*_{n+1}) \text{ and } e = \{u_l, v_l\} (= \{u^*_{n+1}, w^*_{n+1}\}) \\ 0 & \text{if } i = i_l (= i^*_{n+1}) \text{ and } e = \{u_l, w_l\} (= \{u^*_{n+1}, w^*_l\}) \\ 1(=X_l) & \text{if } i = i_{n+1} (= i^*_l) \text{ and } e = \{u_{n+1}, w_{n+1}\} (= \{u^*_l, v^*_l\}) \\ 0 & \text{otherwise} \end{cases}$$

$$A_{L,\vec{X},\vec{Y}}(y_x^j) = \begin{cases} \beta(y_x^j) & \text{if } j \in [2m+1]-J \text{ and } u \in [3m]-V \\ 1 & \text{if } j = j_{k,1} \text{ and } x = v_k \text{ for some } k \in [n] \\ Y_k & \text{if } j = j_{k,2} \text{ and } x = u_k \text{ for some } k \in [n] \setminus \{l\} \\ \neg Y_k & \text{if } j = j_{k,2} \text{ and } x = w_k \text{ for some } k \in [n] \setminus \{l\} \\ 1(=Y_l) & \text{if } j = j_{l,2} \text{ and } x = u_l \\ 0(=\neg Y_l) & \text{if } j = j_{l,2} \text{ and } x = w_l \\ 1 & \text{if } j = j_{n+1,1} \text{ and } x = u_{n+1} \\ 1 & \text{if } j = j_{n+1,2} \text{ and } x = v_{n+1} \\ 1 & \text{if } j = j_{n+1,3} \text{ and } x = w_{n+1} \\ 0 & \text{otherwise} \end{cases}$$

$$A_{f(L),\vec{X},\vec{Y}}(y_x^j) = \begin{cases} \beta(y_x^j) & \text{if } j \in [2m+1]-J \text{ and } u \in [3m]-V \\ 1 & \text{if } j = j_{k,1} \text{ and } x = v_k \text{ for some } k \in [n] \\ Y_k & \text{if } j = j_{k,2} \text{ and } x = u_k \text{ for some } k \in [n] \setminus \{l\} \\ \neg Y_k & \text{if } j = j_{k,2} \text{ and } x = w_k \text{ for some } k \in [n] \setminus \{l\} \\ 1 & \text{if } j = j_{l,2} (= j^*_{n+1,1}) \text{ and } x = u_l (= u^*_{n+1}) \\ 0 & \text{if } j = j_{l,2} (= j^*_{n+1,1} \text{ and } x = w_l (= w^*_l) \\ 1(=X_l) & \text{if } j = j_{n+1,1} (= j^*_{l,2}) \text{ and } x = u_{n+1} = u^*_l \\ 1 & \text{if } j = j_{n+1,2} (= j^*_{n+1,2}) \text{ and } x = v_{n+1} = v^*_{n+1} \\ 1 & \text{if } j = j_{n+1,3} (= j^*_{l,1}) \text{ and } x = w_{n+1} = v^*_l \\ 0 & \text{otherwise} \end{cases}$$

■

D Proofs and Calculations for Section 12

Lemma 22 *If L and L^* are reduction layouts with $HD(L, L^*) \leq d$, then there are at most $2d$ positions i with $S_i(L) \neq S_i(L^*)$.*

Proof: Let $L = (\vec{\imath}, \vec{\jmath}, \vec{u}, \vec{v}, \vec{w})$ and let $L^* = (\vec{\imath}^*, \vec{\jmath}^*, \vec{u}^*, \vec{v}^*, \vec{w}^*)$. We consider each position where L and L^* might differ and see how each affects the functions $\vec{S}$ given in Definition 12.1.

1. If $i_k \neq i^*_k$, with $k \leq n$, then we might have that $S_{n+1+k}(L) = N_2(i_k) \neq N_2(i^*_k) = S_{n+1+k}(L^*)$, or that $S_{2n+2+k}(L) = \mathcal{K}_{1,2}(E_{i_k}[V_{j_{k,1}} \cap V_{j_{k,2}}]) \neq \mathcal{K}_{1,2}(E_{i^*_k}[V_{j^*_{k,1}} \cap V_{j^*_{k,2}}])S_{2n+2+k}(L^*)$.
2. If $i_{n+1} \neq i^*_{n+1}$, then we might have that $S_{2n+2}(L) = N_3(i_{n+1}) \neq N_3(i^*_{n+1}) = S_{2n+2}(L^*)$, or that $S_{3n+3}(L) = \mathcal{K}_{1,2}(E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]) \neq \mathcal{K}_{1,2}(E_{i^*_{n+1}}[V_{j^*_{n+1,1}} \cap V_{j^*_{n+1,2}} \cap V_{j^*_{n+1,3}}]) = S_{3n+3}(L^*)$.
3. If, for some $k \leq n$, $(j_{k,1}, j_{k,2}) \neq (j^*_{k,1}, j^*_{k,2})$ then we might have that $S_{2n+2+k}(L) = \mathcal{K}_{1,2}(E_{i_k}[V_{j_{k,1}} \cap V_{j_{k,2}}]) \neq \mathcal{K}_{1,2}(E_{i^*_k}[V_{j^*_{k,1}} \cap V_{j^*_{k,2}}]) = S_{2n+2+k}(L^*)$.
4. If $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \neq (j^*_{n+1,1}, j^*_{n+1,2}, j^*_{n+1,3})$ then we might have that
$$S_{3n+3}(L) = \mathcal{K}_{1,2}(E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]) \neq \mathcal{K}_{1,2}(E_{i^*_{n+1}}[V_{j^*_{n+1,1}} \cap V_{j^*_{n+1,2}} \cap V_{j^*_{n+1,3}}]) = S_{3n+3}(L^*)$$
5. Differences between (u_k, v_k, w_k) and (u^*_k, v^*_k, w^*_k) do not affect any of the S_i's.

■

Proof:(The calculations ensuring Property 3, of Lemma 19 as applied in the proof of Lemma 13 .)

1. Coordinates $1, \dots n+1$: $F_k(i_1, \dots i_{k-1}) = \{i_1, \dots i_{k-1}\}$ and $X_k = [m]$, therefore:
$$\begin{aligned} |F_k(L) \oplus F_k(L^*)| &= |\{i_1, \dots i_{k-1}\} \oplus \{i^*_1, \dots i^*_{k-1}\}| \leq d \\ &= \frac{d}{3n+3} \frac{3n+3}{m} m = \frac{d}{3n+3} \frac{3\gamma m}{m} m = \frac{3d\gamma}{3n+3} |X_k| \end{aligned}$$
2. For coordinates $n+2, \dots 2n+1$, $X_{n+1+k} = [2m+1]^2$ and
$$\begin{aligned} F_{n+1+k}(\vec{\imath}, (j_{1,1}, j_{1,2}), \dots (j_{k-1,1}, j_{k-1,2})) &= pm_{[2m+1]}(\{j_{1,1}, j_{1,2}, \dots j_{k-1,1}, j_{k-1,2}\}) \\ F_{n+1+k}(\vec{\imath}^*, (j^*_{1,1}, j^*_{1,2}), \dots (j^*_{k-1,1}, j^*_{k-1,2})) &= pm_{[2m+1]}(\{j^*_{1,1}, j^*_{1,2}, \dots j^*_{k-1,1}, j^*_{k-1,2}\}) \end{aligned}$$
Notice that for any X, Y, $pm_{[2m+1]}(X) \oplus pm_{[2m+1]}(Y) \subseteq pm_{[2m+1]}(X \oplus Y)$. On the other hand, $HD(L, L^*) \leq d$, so $|\{j_{1,1}, j_{1,2}, \dots j_{k-1,1}, j_{k-1,2}\} \oplus \{j^*_{1,1}, j^*_{1,2}, \dots j^*_{k-1,1}, j^*_{k-1,2}\}| \leq 2d$, and therefore
$$\begin{aligned} |F_{n+1+k}(L) \oplus F_{n+1+k}(L^*)| &\leq 2 \cdot 2d \cdot (2m+1) = \frac{4d}{(3n+3)(2m+1)}(3n+3)(2m+1)^2 \\ &= \frac{4d}{3n+3} \frac{3\gamma m}{2m+1} |X_{n+1+k}| < \frac{4d}{3n+3} \frac{3\gamma m}{2m} |X_{n+1+k}| = \frac{6d\gamma}{3n+3} |X_{n+1+k}| \end{aligned}$$

3. At coordinate $2n+2$, $X_{2n+2} = [2m+1]^3$ and

$$\begin{aligned} F_{2n+2}(\vec{\imath}, (j_{1,1}, j_{1,2}), \ldots (j_{n,1}, j_{n,2})) &= tm_{[2m+1]}\left(\{j_{1,1}, j_{1,2}, \ldots j_{n,1}, j_{n,2}\}\right) \\ F_{2n+2}(\vec{\imath}^*, (j^*_{1,1}, j^*_{1,2}), \ldots (j^*_{n,1}, j^*_{n,2})) &= tm_{[2m+1]}\left(\{j^*_{1,1}, j^*_{1,2}, \ldots j^*_{n,1}, j^*_{n,2}\}\right) \end{aligned}$$

Notice that for any X, Y, $tm_{[2m+1]}(X) \oplus tm_{[2m+1]}(Y) \subseteq tm_{[2m+1]}(X \oplus Y)$. On the other hand, $HD(L, L^*) \leq d$, so $|\{j_{1,1}, j_{1,2}, \ldots j_{n,1}, j_{n,2}\} \oplus \{j^*_{1,1}, j^*_{1,2}, \ldots j^*_{n,1}, j^*_{n,2}\}| \leq 2d$, and therefore

$$\begin{aligned} |F_{2n+2}(L) \oplus F_{2n+2}(L^*)| &\leq 3 \cdot 2d \cdot (2m+1)^2 = \frac{6d}{(3n+3)(2m+1)}(3n+3)(2m+1)^3 \\ &= \frac{6d}{3n+3}\frac{3n+3}{2m+1}(2m+1)^3 = \frac{6d}{3n+3}\frac{3\gamma m}{2m+1}(2m+1)^3 \\ &< \frac{6d}{3n+3}\frac{3\gamma m}{2m}(2m+1)^3 = \frac{9d\gamma}{3n+3}(2m+1)^3 \end{aligned}$$

4. For coordinates $2n+3, \ldots 3n+3$, $X_{2n+2+k} = [3m]^3$ and

$$F_{2n+2+k}(\vec{\imath}, \vec{\jmath}, (u_1, v_1, w_1), \ldots (u_{k-1}, v_{k-1}, w_{k-1})) = tm(\{u_1, v_1, w_1, \ldots u_{k-1}, v_{k-1}, w_{k-1}\})$$

Notice that for any finite sets X and Y: $tm_{[3m]}(X) \oplus tm_{[3m]}(Y) \subseteq tm_{[3m]}(X \oplus Y)$.

$$\begin{aligned} &|F_{2n+2+k}(L) \oplus F_{2n+2+k}(L^*)| \\ &= |tm_{[3m]}(\{u_1, v_1, w_1, \ldots u_{k-1}, v_{k-1}, w_{k-1}\}) \oplus tm_{[3m]}(\{u^*_1, v^*_1, w^*_1, \ldots u^*_{k-1}, v^*_{k-1}, w^*_{k-1}\})| \\ &\leq |tm_{[3m]}(\{u_1, v_1, w_1, \ldots u_{k-1}, v_{k-1}, w_{k-1}\} \oplus \{u^*_1, v^*_1, w^*_1, \ldots u^*_{k-1}, v^*_{k-1}, w^*_{k-1}\})| \\ &\leq 3 \cdot 3d \cdot (3m)^2 = \frac{9d}{3n+3}\frac{3n+3}{3m}(3m)^3 = \frac{9d}{3n+3}\frac{3\gamma m}{3m}(3m)^3 = \frac{9d\gamma}{3n+3}(3m)^3 \end{aligned}$$

Therefore, for every $i = 1, \ldots 3n$, $|F_i(L) \oplus F_i(L^*)| \leq \frac{9d\gamma}{3n+3}|X_i|$.

■

Lemma 23 *Let $\delta > 0$ be given, and let m be an integer $\geq 36/\delta$. Let $(\mathcal{V}_I, \mathcal{V}_{II})$ be a partition of $MVars_m$, with $\delta(\mathcal{V}_I, \mathcal{V}_{II}) \geq \delta$. Let D be as in the Proof of Lemma 12. Let G, N_2 and N_3 be as in Definition 8.1. Let U be the uniform distribution over $i_{n+1}, i_l \in [m]$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in [2m+1]^3$, and $j_{l,1}, j_{l,2}) \in [2m+1]^2$. Let A be the event that $i_{n+1} \in G$, $i_l \in G$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in N_3(i_p)$, and $(j_{n+1,2}, j_{l,1}, j_{l,2}) \in N_3(i_l)$.*

$$\mathbb{E}_U[D \cdot \chi_A] \geq \delta(\mathcal{V}_I, \mathcal{V}_{II})/2$$

Proof: Let B_0 be the event that either $j_{l,1} = j_{l,2}$, $j_{n+1,1} = j_{n+1,2}$, $j_{n+1,2} = j_{n+1,3}$, $j_{n+1,3} = j_{n+1,1}$, $j_{l,1} = j_{n+1,2}$, or $j_{l,1} = j_{n+1,2}$. Let B_1 be the event that $i_{n+1} \notin G$, let B_2 be the event that $i_l \notin G$, let B_3 be the event that $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \notin N_3(i_{n+1})$, and let B_4 be the event that $(j_{n+1,2}, j_{l,1}, j_{l,2}) \notin N_3(i_l)$. For each $i = 0, \ldots 4$, let $B^*_i = B_i \cap \bigcap_{j=0}^{i-1} B^c_j$. Because the B^*'s partition A^c we have that:

$$\mathbb{E}_U[D] = \mathbb{E}_U[D \cdot \chi_A] + \sum_{i=0}^{4} \mathbb{E}_U[D \cdot \chi_{B^*_i}]$$

Set $\delta^* = \delta(\mathcal{V}_I, \mathcal{V}_{II})$. The calculations below show that $U(B_0^*) \leq 6/(2m+1)$ and for each $i = 1, \ldots 4$, $E_U[D \cdot \chi_{B_i^*}] \leq (5\delta^*/12)U(B_i^*)$. Modulo those calculations, we have the lemma:

$$\begin{aligned}
\mathbb{E}_U[D \cdot \chi_A] &= \mathbb{E}_U[D] - \sum_{i=0}^{4} \mathbb{E}_U[D \cdot \chi_{B_i^*}] \geq \delta^* - 6/(2m+1) - \sum_{i=1}^{4} (5\delta^*/12)U(B_i^*) \\
&\geq \delta^* - 5\delta^*/12 - 6/(2m+1) \geq 7\delta/12 - 6/(2(36/\delta)) = 7\delta^*/12 - \delta/12 \geq \delta^*/2
\end{aligned}$$

For each of the six pairs $j_{l,1}$ and $j_{l,2}$, $j_{n+1,1}$ and $j_{n+1,2}$, $j_{n+1,2}$ and $j_{n+1,3}$, $j_{n+1,3}$ and $j_{n+1,1}$, $j_{n+1,2}$ and $j_{j_{l,1}}$, and $j_{n+1,2}$ and $j_{l,2}$, there is a collision with probability $1/(2m+1)$. Therefore by the union bound, $U(B_0^*) = U(B_0) \leq 6/(2m+1)$. We now bound the expectation over the pieces B_1^*, B_2^*, B_3^*, and B_4^*. Because these events are contained in B_0^c, for elements drawn from these sets, the tuples $(j_{l,1}, j_{l,2})$, $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3})$, and $(j_{n+1,2}, j_{l,1}, j_{l,2})$ each contain distinct elements. To denote this, we will use Z to denote the set of pairs tuples $(\vec{j_{n+1}}, \vec{j_l})$ with $j_{l,1} \neq j_{l,2}$, $j_{n+1,1} \neq j_{n+1,2}$, $j_{n+1,2} \neq j_{n+1,3}$, $j_{n+1,3} \neq j_{n+1,1}$, $j_{n+1,1} \neq j_{j_{l,1}}$, and $j_{n+1,1} \neq j_{l,2}$, let $[2m+1]_2$ denote all ordered pairs from $[2m+1]$ with two distinct values and let $[2m+1]_3$ denote all ordered triples from $[2m+1]$ with three distinct values. Finally, set $M = m^2(2m+1)^5$,

$$\begin{aligned}
\mathbb{E}_U[D \cdot \chi_{B_1^*}] &= \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{i_l \in [m]} \sum_{(\vec{j_{n+1}}, \vec{j_l}) \in Z} D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \\
&\leq \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{i_l \in [m]} \sum_{\substack{\vec{j_{n+1}} \in [2m+1]_3 \\ \vec{j_l} \in [2m+1]_2}} D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \\
&= \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in [2m+1]_3} \sum_{\substack{i_l \in [m] \\ \vec{j_l} \in [2m+1]_2}} D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \\
&= \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in N_3(i_{n+1})} \sum_{\substack{i_l \in [m] \\ \vec{j_l} \in [2m+1]_2}} D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \\
&\quad + \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in [2m+1]_3 \setminus N_3(i_{n+1})} \sum_{\substack{i_l \in [m] \\ \vec{j_l} \in [2m+1]_2}} D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \\
&\leq \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in N_3(i_{n+1})} \sum_{\substack{i_l \in [m] \\ \vec{j_l} \in [2m+1]_2}} 1 + \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in [2m+1]_3 \setminus N_3(i_{n+1})} \sum_{\substack{i_l \in [m] \\ \vec{j_l} \in [2m+1]_2}} (\delta/3) \\
&= \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in N_3(i_{n+1})} m(2m+1)^2 + \frac{1}{M} \sum_{i_{n+1} \notin G} \sum_{\vec{j_{n+1}} \in [2m+1]_3 \setminus N_3(i_{n+1})} (\delta/3)m(2m+1)^2 \\
&\leq \frac{1}{M} \sum_{i_{n+1} \notin G} (\delta/12)(2m+1)^3 m(2m+1)^2 + \frac{1}{M} \sum_{i_{n+1} \notin G} (2m+1)^3(\delta/3)m(2m+1)^2 \\
&= \frac{1}{M} \sum_{i_{n+1} \notin G} (\delta/12 + \delta/3)(2m+1)^3 m(2m+1)^2 = (5\delta/12)U(B_1^*)
\end{aligned}$$

To bound $E_U[D \cdot \chi_{B_2^*}]$ we need first show that for all $i_{n+1}, i_l \in [m]$, all $\vec{j_{n+1}} \in [2m+1]_3$, and all $\vec{j_l} \in [2m+1]_2 \setminus N_2(i_l)$ $D(i_{n+1}, i_l, \vec{j_{n+1}}, \vec{j_l}) \leq \delta/3$. To see this choose $j^* \in \{j_{n+1,1}, j_{n+1,2}, j_{n+1,3}\} \setminus$

$\{j_{l,1}, j_{l,2}\}$ and calculate:

$$\begin{aligned} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) &= \frac{|E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}] \cap E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}]|}{\binom{3m}{2}} \\ &\leq \frac{|E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}} \cap V_{j^*}]|}{\binom{3m}{2}} \leq \delta/3 \end{aligned}$$

$$\begin{aligned} \mathbb{E}_U[D \cdot \chi_{B_2^*}] &= \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} \sum_{(\vec{\jmath_{n+1}}, \vec{\jmath_l}) \in Z} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) \\ &\leq \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} \sum_{\vec{\jmath_l} \in N_2(i_l)} \sum_{\vec{\jmath_{n+1}} \in [2m+1]_3} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) \\ &\quad + \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} \sum_{\vec{\jmath_l} \in [2m+1]_2 \setminus N_2(i_l)} \sum_{\vec{\jmath_{n+1}} \in [2m+1]_3} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) \\ &\leq \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} \sum_{\vec{\jmath_l} \in N_2(i_l)} \sum_{\vec{\jmath_{n+1}} \in [2m+1]_3} 1 + \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} \sum_{\vec{\jmath_l} \in [2m+1]_2 \setminus N_2(i_l)} \sum_{\vec{\jmath_{n+1}} \in [2m+1]_3} (\delta/3) \\ &\leq \frac{1}{M} \sum_{i_{n+1} \in G} \sum_{i_l \notin G} (\delta/12)(2m+1)^5 + \frac{1}{M} \sum_{i_l \notin G} (\delta/3)(2m+1)^5 = (5\delta/12) U(B_2^*(i_{n+1})) \end{aligned}$$

To bound $\mathbb{E}_U[D \cdot \chi_{B_3^*}]$, note that for all $(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) \in B_3^*$, because $(j_{n+1,1}, j_{n+1,2}, j_{n+1,3}) \in [2m+1]_3 \setminus N_3(i_{n+1})$:

$$\begin{aligned} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) &= \frac{|E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}] \cap E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]|}{\binom{3m}{2}} \\ &\leq \frac{|E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]|}{\binom{3m}{2}} \leq \delta/3 \end{aligned}$$

Therefore $\mathbb{E}_U[D \cdot \chi_{B_3^*}] \leq (\delta/3) U(B_3^*)$.

Similarly, to bound $\mathbb{E}_U[D \cdot \chi_{B_4^*}]$, observe that for all $(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) \in B_4^*$, because $(j_{n+1,2}, j_{l,1}, j_{l,2}) \in [2m+1]_3 \setminus N_3(i_l)$:

$$\begin{aligned} D(i_{n+1}, i_l, \vec{\jmath_{n+1}}, \vec{\jmath_l}) &= \frac{|E_{i_l}[V_{j_{l,1}} \cap V_{j_{l,2}}] \cap E_{i_{n+1}}[V_{j_{n+1,1}} \cap V_{j_{n+1,2}} \cap V_{j_{n+1,3}}]|}{\binom{3m}{2}} \\ &\leq \frac{|E_{i_l}[V_{j_{n+1,2}} \cap V_{j_{l,1}} \cap V_{j_{l,2}}]|}{\binom{3m}{2}} \leq \delta/3 \end{aligned}$$

Therefore $\mathbb{E}_U[D \cdot \chi_{B_4^*}] \leq (\delta/3) U(B_4^*)$.

■

ECCC ISSN 1433-8092
http://eccc.hpi-web.de/