SIAM J. COMPUT.
Vol. 0, No. 0, pp. FOCS22-288–FOCS22-339

ON BOUNDED DEPTH PROOFS FOR TSEITIN FORMULAS ON THE GRID; REVISITED*

JOHAN HÅSTAD† AND KILIAN RISSE‡

Abstract. We study Frege proofs using depth-d Boolean formulas for the Tseitin contradiction on $n \times n$ grids. We prove that if each line in the proof is of size M, then the number of lines is exponential in $n/(\log M)^{O(d)}$. This strengthens a recent result of Pitassi, Ramakrishman, and Tan [2022 *IEEE* 62*nd Annual Symposium on Foundations of Computer Science,* 2022, pp. 445–456]. The key technical step is a multiswitching lemma extending the switching lemma of Håstad [*J. ACM,* 68 (2021), 1] for a space of restrictions related to the Tseitin contradiction. The strengthened lemma also allows us to improve the lower bound for standard proof size of bounded depth Frege refutations from exponential in $\widetilde{\Omega}(n^{1/59d})$ to exponential in $\widetilde{\Omega}(n^{1/d})$. This strengthens the bounds given in the preliminary version of this paper [J. Håstad and K. Risse, 2022 *IEEE* 63*rd Annual Symposium on Foundations of Computer Science,* 2022, pp. 1138–1149].

Key words. proof complexity, bounded depth Frege, Tseitin formulas

MSC code. 68Q05

DOI. 10.1137/22M153851X

1. Introduction.

Mathematicians like proofs, formal statements where each line follows by simple reasoning rules from previously derived lines. Each line derived in this manner, assuming that the reasoning steps are sound, can give us some insight into the initial assumptions of the proof. A particularly interesting consequence is contradiction. Deriving an obviously false statement allows us to conclude that the initial assumptions, also called axioms, are contradictory. We continue the study of Frege proofs of contradiction where each line in the proof is a Boolean formula of depth d. This subject has a long tradition, so let us start with a very brief history.

A very basic proof system is resolution: each line of such a proof simply consists of a disjunction of literals. The derivation rules of resolution are also easy to understand and simple to implement, but the proof system nevertheless gives rise to reasonably short proofs for some formulas. It is far from easy to give lower bounds for the size of proofs in resolution, but it has been studied for a long time and by now many strong bounds are known. An early paper by Tseitin [20] defined an important class of contradictions based on graphs that is central to this and many previous papers. For each edge there is a variable, and the requirement is that the parity of the variables incident to any given node sum to a particular bit which is called the charge of that node. If the sum of the charges is 1 modulo 2, this is a contradiction. For a subsystem of resolution, called regular resolution, Tseitin proved exponential lower bounds on

*Received by the editors December 1, 2022; accepted for publication (in revised form) July 18, 2025; published electronically October 16, 2025. A preliminary version of this paper appeared in the *Proceedings of the* 63*rd Annual IEEE Symposium on Foundations of Computer Science* (FOCS '22).
https://doi.org/10.1137/22M153851X
Funding: Supported by the Approximability and Proof Complexity project funded by the Knut and Alice Wallenberg Foundation. The work of the second author was supported by the Swiss National Science Foundation through project 200021-184656 and the Postdoc.Mobility fellowship P500-2_235298. Most of this work was done while the second author was affiliated with KTH Royal Institute of Technology and EPFL.
†KTH Royal Institute of Technology, Stockholm, Sweden (johanh@kth.se).
‡Lund University, LTH, Lund, Sweden (kilianr@kth.se).

refutations of these formulas. After this initial lower bound it took almost another two decades before the first strong lower bound for general resolution was obtained by Haken [5], whose lower bound applied to the pigeonhole principle (PHP). Many other resolution lower bounds followed, but as we are not so interested in resolution and rather intend to study the more powerful proof system with formulas of larger, though still bounded, depth d on each line, let us turn to such proof systems.

The study of proofs with lines limited to depth d dates back several decades. A pioneering result was obtained by Ajtai [1] who showed that the PHP cannot be proved in polynomial size for any constant depth d. Developments continued in the 1990s, and polynomial size proofs were ruled out for values of d up to $O(\log\log n)$ for both the PHP [12, 11] as well as the Tseitin contradiction defined over complete [21] and expander graphs [2].

These developments followed previous work where the computational power of the class of circuits[1] of depth d was studied [18, 4, 22, 6, 15, 19]. It is not surprising that it is easier to understand the computational power of a single circuit rather than to reason about a sequence of formulas giving a proof. This manifested itself in that while the highest value of d for which strong bounds were known for size of proofs remained at $O(\log\log n)$, the results for circuit size extended to almost logarithmic depth.

This gap was (essentially) closed in two steps. First Pitassi et al. [14] proved superpolynomial lower bounds for d up to $o(\sqrt{\log n})$, and then Håstad [8] extended this to depth $\Theta(\frac{\log n}{\log\log n})$ which, up to constants, matches the result for circuits.

The key technique used in most of the described results is the use of restrictions. These set most of the variables to constants which simplifies the circuit or formulas studied. If done carefully one can at the same time preserve the contradiction refuted or the function computed. Of course one cannot exactly preserve the contradiction, and to be more precise a contradiction with parameter n before the restriction turns into a contradiction of the same type but with a smaller parameter, n/T, after the restriction.

The simplification under a restriction usually takes place in the form of a switching lemma. This makes it possible to convert depth-d formulas to formulas of depth $d-1$. A sequence of restrictions is applied to reduce the depth to (essentially) zero, making the circuit or formula straightforward to analyze. The balance to be struck is to find a set of restrictions that leaves a large resulting contradiction but at the same time allows a switching lemma to be proved with good parameters.

In proof complexity the most commonly studied measure is the total size of a proof. There are two components to this size, the number of reasoning steps needed and the size of each line of the proof. In some cases, such as resolution, each line is automatically bounded in size and hence any lower bound for proof size is closely related to the number of proof steps. In some other situations the line sizes may grow and an interesting question is whether this can be avoided.

This line of investigation for Frege proofs with bounded depth formulas was recently initiated by Pitassi, Ramakrishnan, and Tan [13]. They consider the Tseitin contradiction defined over the grid of size $n \times n$, a setting where strong total size lower bounds for Frege refutations of bounded depth had previously been given by Håstad [8]. If each line of the refutation is limited to size M and depth d, then Pitassi, Ramakrishnan, and Tan [13] showed that the Frege proof must consist of at least $\exp(n/2^{O(d\sqrt{\log M})})$ many lines. For most interesting values of M this greatly

[1]When the depth is small, there is no major difference between circuits and formulas, so the reader should feel free to ignore this difference.

improves the bounds implied by the results for total proof size. In particular if M is a polynomial, the lower bounds are of the form $\exp(n^{1-o(1)})$, as long as $d = o(\sqrt{\log n})$, in contrast to the total size lower bounds of the form $\exp(n^{\Omega(1/d)})$. Pitassi, Ramakrishnan, and Tan [13] rely on the restrictions introduced by Håstad [8] but analyze them using the methods of Pitassi et al. [14].

We study the same Tseitin contradiction on the grid and improve the lower bounds to $\exp(n/(\log M)^{O(d)})$, a bound conjectured by Pitassi, Ramakrishnan, and Tan [13]. Note that if the size of each line is bounded by $M = O\big(n^{\mathrm{polylog}(n)}\big)$ and the depth is $d = o\big(\frac{\log n}{\log\log n}\big)$, then the length lower bound is of the form $\exp\big(n^{1-o(1)}\big)$. For this setting of parameters the bound is essentially optimal since there is a resolution upper bound of size $2^{O(n)}$.

For other settings of parameters we cannot match the lower bound. We do believe, though, that the bound obtained is tight for a wide range of parameters. While we cannot match the lower bounds with actual proofs, we can at least represent the intermediate results of a natural proof by formulas of the appropriate size. We discuss this in more detail below.

1.1. Overview of proof techniques. The structure of the proof of our main result follows the approach of [13] but relies on proving much sharper variants of the switching lemma.

In a standard application of a switching lemma to proof complexity one picks a restriction and demands that switching happens to all depth-2 formulas in the entire proof. Each formula switches successfully with high probability, and by an application of a union bound it is possible to find a restriction to get them all to switch simultaneously.

The key idea of [13] is that one need not consider all formulas in the proof at the same time. Rather one can focus on the sub-formulas of a given line. It is sufficient to establish that these admit what is called an ℓ-common partial decision tree of small depth. This is a decision tree with the property that at each leaf, each of the formulas can be described by a decision tree of depth ℓ. It turns out that this is enough to analyze the proof and establish that a short proof cannot derive contradiction. The key property is that it is sufficient to only look at the constant number of formulas involved in each derivation step and analyze each such step separately.

The possibility to compute a set of formulas by an ℓ-common partial decision tree after having been hit by a restriction is exactly what is analyzed by what has become known as a "multiswitching lemma" as introduced by [7, 10]. This concept was introduced in order to analyze the correlation of small circuits of bounded depth with parity but turns out to also be very useful in the current context.

Even though there is no general method, it seems like when it is possible to prove a standard switching lemma there is good hope of also proving a multiswitching lemma with similar parameters. This happens when going from [6] to [7] and when going from [14] to [13]. We follow the same approach here, and this paper very much builds on [8]. We need a slight modification of the space of restrictions and changes to some steps of the proof, but a large fraction of the proof remains untouched. Let us briefly touch on the necessary changes.

The switching lemma of Håstad [8] has a failure probability to not switch to a decision tree of depth s of the form $(As)^{\Omega(s)}$ where A depends on other parameters. As a first step one needs to eliminate the factor s in the base of the exponent. This triggers the above-mentioned change in the space of restrictions. This change enables us to prove a standard switching lemma with stronger parameters and, as a warm-up, we give this proof in the current paper. This results in an improvement of the lower

bound for total proof size from $\exp(\tilde{\Omega}(n^{1/58d}))$ to $\exp(\tilde{\Omega}(n^{1/d}))$. We believe that this lower bound is tight up to polylogarithmic factors in the exponent.

The high level idea of the proof of the multiswitching lemma is that for each of the formulas analyzed we try to construct a decision tree of depth ℓ. If this fails, then we take the long branch in the resulting decision tree and instead query these variables in the common decision tree.

1.2. Constructing small proofs. Let us finally comment on a possible upper bound: how to construct efficient refutations. If we are allowed to reason with linear equations modulo 2, then the Tseitin contradiction has efficient refutations. In particular on the grid we can sum all equations in a single column giving an equation containing $O(n)$ variables that must be satisfied. Adding the corresponding equation for the adjacent column maintains an equation of the same size, and we can keep adding equations from adjacent columns until we have covered the entire grid. We derive a contradiction, and we never use an equation containing more than $O(n)$ variables.

If we consider resolution, then it is possible to represent a parity of size m as a set of clauses. Indeed, looking at the equation $\sum_{i=1}^{m} x_i = 0$ we can replace this by the 2^{m-1} clauses of full width where an odd number of variables appear in negative form. Now replace each parity in the above proof by its corresponding clauses. It is not difficult to check that Gaussian elimination can be simulated by resolution. Given linear equation $L_1 = b_1$ and $L_2 = b_2$ with m_1, and m_2 variables, respectively, and both containing the variable x, we want to derive all clauses representing $L_1 \oplus L_2 = b_1 \oplus b_2$. We have 2^{m_1-1} clauses representing the first linear equation and the 2^{m_2-1} clauses representing the second linear equation. Now we can take each pair of clauses and resolve over x, and this produces a good set of clauses. If L_1 and L_2 do not have any other common variables, we are done. If they do contain more common variables, then additional resolution steps are needed, but these are not difficult to find and we leave it to the reader to figure out this detail. We conclude that Tseitin on the grid allows resolution proofs of length $2^{O(n)}$.

Let us consider proofs that contain formulas of depth d, and let us see how to represent a parity. Given $\sum_{i=1}^{m} x_i = 0$ we can divide the variables into groups of size $(\log M)^{d-1}$ and write down formulas of depth d and size M that represent the parity and the negation of the parity of each group. Assume that the output gate of each of these formulas is an or. We now use the above clause representation of the parity of the groups and get a set of $2^{m/(\log M)^{d-1}}$ formulas of size $mM/(\log M)^{d-1}$ that represent the linear equations This means that we can represent each line in the parity proof by about $2^{n/(\log M)^{d-1}}$ lines of size about M. We do not know how to syntactically translate a Gaussian elimination step to some proof steps in this representation, and thus we do not actually get a proof, only a representation of the partial results.

1.3. Organization. Let us outline the contents of this paper. We start in section 2 with some preliminaries. In section 3 we define the set of restrictions used in the current paper which are almost the same as in [8]. Next we show how to derive our two main theorems assuming the new switching lemmas in section 4. In section 5 we provide some further preliminaries and explain the proof idea of the standard switching lemma. The full proof of the standard switching lemma is given in section 6, and the extension to a multiswitching lemma is presented in section 7. We end with some conclusions in section 8.

2. Preliminaries. Logarithms are denoted by log and are always with respect to the base 2. For integers $n \geq 1$ we introduce the shorthand $[n] = \{1, \ldots, n\}$ and sometimes identify singletons $\{u\}$ with the element u. We identify *false* (*true*) with 0

(with 1) and let the binary *or* and *and* connective be denoted by $\vee$ and $\wedge$, while the unary *negation* connective is denoted by $\neg$.

Since Frege systems over the basis $\vee$, $\wedge$ and $\neg$ can polynomially simulate each other [3], it is not essential what Frege system we use. We choose to work with Schoenfield's system as previous work has [21, 14, 8, 13].

2.1. The Frege proof system. Schoenfield's Frege system works over the basis $\vee$ and $\neg$. We simulate a conjunction $A \wedge B$ by treating it as an abbreviation for the formula $\neg(\neg A \vee \neg B)$.

If A is a formula over variables $p_1, \ldots, p_m$, and σ maps the variables $p_1, \ldots, p_m$ to formulas $B_1, \ldots, B_m$, then $\sigma(A)$ is the formula obtained from A by replacing the variable p_i with $B_i = \sigma(p_i)$ for all i. A *rule* is a sequence of formulas written as $A_1, \ldots, A_k \vdash A_0$. If every truth assignment satisfying all of $A_1, \ldots, A_k$ also satisfies A_0, then the rule is *sound.* A formula C_0 is inferred from $C_1, \ldots, C_k$ by the rule $A_1, \ldots, A_k \vdash A_0$ if there is a function σ mapping the variables $p_1, \ldots, p_m$, over which $A_0, \ldots, A_k$ are defined, to formulas $B_1, \ldots, B_m$ such that $C_i = \sigma(A_i)$ for all i.

The Frege system we consider consists of the rules

$$\begin{aligned} \vdash p \vee \neg p \quad & \text{Excluded middle,} \\ p \vdash q \vee p \quad & \text{Expansion rule,} \\ p \vee p \vdash p \quad & \text{Contraction rule,} \\ p \vee (q \vee r) \vdash (p \vee q) \vee r \quad & \text{Association rule,} \\ p \vee q, \neg p \vee r \vdash q \vee r \quad & \text{Cut rule.} \end{aligned}$$

A *Frege proof* of a formula B from a formula $A = C_1 \wedge \cdots \wedge C_m$ is a sequence of formulas $F_1, F_2, \ldots, F_\ell$ such that $F_\ell = B$ and every formula F_i in the sequence is either one of $C_1, \ldots, C_m$ or inferred from formulas earlier in the sequence by one of the above rules. Since the above system is sound and complete, a formula B has a proof from a formula A if and only if B is implied by A. A *Frege refutation* of a formula A is a Frege proof of $\perp$ constant false.

The *size of a formula* is the number of connectives in it, and the *depth* of a formula A is the maximum number of alternations of $\vee$ and $\neg$ on any root-to-leaf path when A is viewed as a tree. The *size of a Frege proof* is the sum of the sizes of all formulas in the proof, and the *depth of a proof* is the maximum depth of any formula in it.

2.2. The grid. Throughout the paper we work over graphs $G_n = (V, E)$ with n^2 nodes which we call *the grid.* However, in order to avoid problems at the boundary, we in fact work over the 2-*dimensional torus:* each node $(i, j) \in V$ is indexed by two integers $i, j \in [n]$, and an edge $\{u, v\}$ is in E if and only if it connects two adjacent nodes, that is, if one of the coordinates of u and v is identical and the other differs by 1 modulo n.

For a set $U \subseteq V$ we say that a node v is at distance d from U if there is a node $u \in U$ such that the shortest path between u and v is of length d.

2.3. Tseitin formulas. The Tseitin formula $\mathrm{Tseitin}(G, \alpha)$ defined for a graph G and a vector $\alpha \in \{0,1\}^{V(G)}$ claims that there is a $\{0,1\}$-labeling of the edges of G such that the number of 1-labeled edges incident to each node v is equal to the *charge* α_v modulo 2. This is formalized with a Boolean variable x_e per edge $e \in E(G)$ and encoding the linear constraints

$$\sum_{e \ni v} x_e = \alpha_v \mod 2 \tag{1}$$

for each node $v \in V(G)$ as a conjunctive normal form (CNF) formula. The main case we consider is when $\alpha_v = 1$ for all nodes v. Let us denote this formula by $\text{Tseitin}(G)$. We use more general charges in intermediate steps, and hence the following lemma from [8] is useful. In order to be self-contained we provide a proof in Appendix A.

LEMMA 2.1. *Consider the Tseitin formula* $\text{Tseitin}(G_n, \alpha)$ *defined over the* $n \times n$ *grid. If* $\sum_v \alpha_v$ *is even, then* $\text{Tseitin}(G, \alpha)$ *is satisfiable and has* 2^{r_n} *solutions for a positive integer* r_n *that only depends on* n *and not on* α*.*

As a converse to the above lemma, if $\sum_v \alpha_v$ is odd, then by summing all equations it is easy to see that such a system is contradictory. In particular the Tseitin formulas with $\alpha_v = 1$ for all v are contradictions for graphs with an odd number of nodes. We note that all Tseitin formulas $\text{Tseitin}(G_n, \alpha)$ over the grid graph can be written as a 4-CNF formula with 8 clauses of width 4 for each node.

2.4. Local consistency of assignments. We are interested in solutions to subsystems of the Tseitin formula $\text{Tseitin}(G_n)$. From Lemma 2.1 it follows that if we drop the constraints of a single node, then we obtain a consistent system of linear equations with many solutions. Denote by X the variables that $\text{Tseitin}(G_n)$ is defined over and say that a partial assignment $\alpha\colon X \to \{0, 1, *\}$ assigns a variable $x \in X$ if $\alpha(x) \in \{0, 1\}$.

The *support* of a partial assignment α, denoted by $\text{supp}(\alpha)$, is the set of nodes incident to assigned variables. We say that α is *complete* on a set of nodes $U \subseteq V(G_n)$ if α assigns all variables incident to U and no others. Note that the support of an assignment complete on U also includes the neighbors of U.

We consider partial assignments that give values to few variables. More specifically we are interested in assignments that are complete on a set $U \subseteq V(G_n)$ of size at most $|U| \leq 2n/3$. Note that such a U cannot touch all rows or columns of the grid. Denote by $U^c = V(G_n) \setminus U$ the complement of U.

Since $|U| \leq 2n/3$, the sub-graph $G_n[U^c]$ induced by U^c has a *giant component* that contains almost all nodes of the grid: there are at least $n/3$ complete rows and columns in U^c, and all the nodes of these rows and columns are connected. It is important to control assignments on the other components, *small components,* of $G_n[U^c]$ as they may fail to extend in a consistent manner to these small components. For a set U let the *closure of* U*,* denoted by $\text{closure}(U) \subseteq V(G_n)$, consist of all nodes in U along with all the nodes in the small components of $G_n[U^c]$. Note that $\text{closure}(U)^c$ contains exactly the set of nodes that are in the giant component of $G_n[U^c]$.

DEFINITION 2.2 (local consistency). *A partial assignment* α *with* $U = \text{supp}(\alpha)$ *is* locally consistent *if it can be extended to an assignment complete on* $\text{closure}(U)$ *such that all parity constraints on* $\text{closure}(U)$ *are satisfied. We extend this notion to say that a pair of assignments is* pairwise locally consistent *if they do not give different values to the same variable and the union of the two assignments is locally consistent.*

The following lemma from [8] is used throughout the article.

LEMMA 2.3. *If* α *is a locally consistent assignment satisfying* $|\text{supp}(\alpha)| \leq n/2$*, then for any variable* x_e *there is a locally consistent assignment* $\alpha' \supseteq \alpha$ *with* x_e *in its domain.*

For completeness we provide a proof in Appendix A.

DEFINITION 2.4 (local implication). *Let* α *be a locally consistent assignment. A variable* x *is* locally implied *by* α *if there is a unique* $b \in \{0, 1\}$ *such that the partial assignment* $\alpha \cup \{x \mapsto b\}$ *is locally consistent.*

In particular if a locally consistent assignment α assigns a variable x, then x is locally implied by α.

2.5. Restrictions. Let $\tau\colon \{x_1,\dots,x_m\} \to \{0,1,*\}$ be a partial assignment, and denote by F a formula over the variables in the domain of τ. The formula F *restricted by* τ, denoted by $F\restriction_\tau$, is the formula obtained from F by replacing each variable x_i by $\tau(x_i)$ unless $\tau(x_i) = *$ in which case we leave x_i untouched. More generally, if σ maps variables $x_1,\dots,x_m$ to formulas $A_1,\dots,A_m$, then the formula F *restricted by* σ, denoted by $F\restriction_\sigma$, is the formula obtained from F by replacing the variable x_i with $A_i = \sigma(x_i)$ for all $i \in [m]$.

2.6. Decision trees. A *decision tree* is a directed tree such that every node has either out-degree 2 (an *internal node*) or 0 (a *leaf node*) and all nodes have in-degree 1 except the designated *root node* which has in-degree 0. Edges and leaves are labeled 0 or 1, while internal nodes are labeled with a variable. A *branch* of a decision tree T is a root-to-leaf path in T and the *depth* of T, denoted by $\mathrm{depth}(T)$, is the length of the longest branch in T. Throughout we implicitly assume that internal node labels (variables) of any branch are distinct.

A 1-*branch* (a 0-*branch*) is a branch with a leaf labeled 1 (labeled 0), and a 1-*tree* (a 0-*tree*) is a decision tree where all leaves are labeled 1 (labeled 0). Special cases of b-trees are trees of depth 0. We sometimes write $T = b$ if T is a b-tree.

Given a Boolean assignment τ, we can evaluate a decision tree T: start at the root node v. If v is a leaf, then output its label. Otherwise v is labeled by some variable x. Let $b = \tau(x)$, and recurse on the node that the out-edge b of v points to.

Since every branch B has a minimal partial assignment τ such that any extension of τ traverses B, we interchangeably identify a branch by the root-to-leaf path B, by τ, and by the unique leaf in B.

For a partial assignment α and a decision tree T we obtain the decision tree T restricted by α by iteratively removing internal nodes $v \in T$ labeled $x_i \in \alpha^{-1}\big(\{0,1\}\big)$ and replacing them by the node that the out-edge $\alpha(x_i)$ of v points to.

Equivalently, if we view a decision tree T as a set of branches, then T restricted by α consists of all branches $\tau \in T$ consistent with α (in the standard sense: τ and α do not assign a variable to opposite value). These branches fit nicely into a tree-structure once all information about α is removed.

Let us stress the obvious: the restriction as defined above *always* produces a valid decision tree. Throughout the manuscript we do *not* use the above notion of a restriction. In the following we define the restrictions used.

Decision trees and local consistency. In the following we consider decision trees on the variables of the Tseitin formula $\mathrm{Tseitin}(G_n)$ defined over the $n \times n$ grid.

Definition 2.5 (local consistency for branches). *Let T be a decision tree on the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$. *A branch τ in T is* locally consistent *if the partial assignment τ is locally consistent as an assignment (see Definition* 2.2*) and T is* locally consistent *if all branches τ of T are locally consistent.*

The following is a direct consequence of Lemma 2.3.

Corollary 2.6. *Let T be a decision tree on the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$. *If* $\mathrm{depth}(T) \le n/4$, *then T contains a locally consistent branch.*

Throughout this article we assume that all considered decision trees are of depth at most one fourth of the dimension of the grid we are considering. We may thus assume that all decision trees have a locally consistent branch.

We are going to maintain the even stronger property that all the considered decision trees T are locally consistent, that is, *all branches* of T are locally consistent. This is easy to maintain during the creation of a decision tree: when extending a decision tree at some leaf τ we simply disallow queries to a variable x if x is locally implied by τ.

We further intend to maintain this property when a decision tree T is hit by a locally consistent assignment α. To this end we trim T aggressively during the restriction: denote by $T\restriction_\alpha$ the decision tree that consists of all the branches $\tau \in T$ that are *pairwise locally consistent* with α as defined in Definition 2.2. If there are indeed some such branches τ, then these fit again into a tree-structure once any information about variables x locally implied by α is removed.

However, if there are *no* branches τ pairwise locally consistent with α, then the above restriction fails to return a decision tree. The following lemma, a direct consequence of Lemma 2.3, states that if α is small and the tree T is of low depth, then the restriction $T\restriction_\alpha$ does not fail, that is, the restricted tree $T\restriction_\alpha$ is indeed a locally consistent decision tree.

COROLLARY 2.7. *Let T be a decision tree on the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$*, and denote by α a locally consistent assignment. If* $|\mathrm{supp}(\alpha)| + 2 \cdot \mathrm{depth}(T) \leq n/2$*, then $T\restriction_\alpha$ is a locally consistent decision tree.*

DEFINITION 2.8 (functional equivalence of decision trees). *Denote by T_1 and T_2 two locally consistent decision trees of depth at most $n/8$ defined over the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$*. The decision trees T_1 and T_2 are* functionally equivalent *if for every branch τ of T_1 ending in a leaf labeled b it holds that $T_2\restriction_\tau = b$ and vice versa.*

The assumption in Definition 2.8 on the depth of T_1 and T_2 ensures that the restricted trees are well defined.

LEMMA 2.9. *Let T_1 and T_2 be two locally consistent decision trees defined over the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$*, and denote by α a locally consistent assignment. Suppose $T_1\restriction_\alpha$ is a b-tree. If T_1 and T_2 are functionally equivalent and* $|\mathrm{supp}(\alpha)| + 2\big(\mathrm{depth}(T_1) + \mathrm{depth}(T_2)\big) \leq n/2$*, then $T_2\restriction_\alpha$ is a b-tree.*

Proof. Suppose that $T_2\restriction_\alpha$ has a $\neg b$-branch τ_2. Since T_1 and T_2 are functionally equivalent it holds that $T_1\restriction_{\alpha\cup\tau_2}$ is a $\neg b$-tree, where we use Corollary 2.7. This contradicts the assumption that $T_1\restriction_\alpha$ is a b-tree. ∎

For decision trees $T, T_1, T_2, \ldots, T_m$ we say that T *represents* $\bigvee_{i=1}^m T_i$ if for every branch τ of T ending in a leaf labeled 1, it holds that there is an $i \in [m]$ such that $T_i\restriction_\tau = 1$, and if τ ends in a leaf labeled 0, then for all $i \in [m]$ it holds that $T_i\restriction_\tau = 0$.

Recall that the key idea of [13] is that one need not consider all formulas in the proof at the same time. Rather one can focus on the sub-formulas of a given line and establish that these admit what is called an ℓ-common partial decision tree of small depth. This is a decision tree with the property that at each leaf each of the formulas can be described by a decision tree of depth ℓ. The formal definition follows.

DEFINITION 2.10 (common partial decision tree). *Let $T_1, \ldots, T_m$ be decision trees over the variables of the Tseitin formula* $\mathrm{Tseitin}(G_n)$*. A decision tree $\mathcal{T}$ is said to be an* ℓ-common partial decision tree for $T_1, \ldots, T_m$ of depth t *if*

1. *the depth of $\mathcal{T}$ is bounded by t, and*
2. *for every T_i and branch $\tau \in \mathcal{T}$ there are decision trees $T(i, \tau)$ of depth ℓ satisfying the following. Let $\mathcal{T}_i$ be the decision tree obtained from $\mathcal{T}$ by appending*

the trees $T(i,\tau)$ at the corresponding leaf τ of $\mathcal{T}$. Then, if a branch $\tau' \in \mathcal{T}_i$ ends in a leaf labeled b, it holds that $T_i\lceil_{\tau'} = b$.

Let $m_1,\ldots,m_M \in \mathbb{N}^+$ for some integer M. Consider decision trees T_i^j for $i \in [m_j]$ and $j \in [M]$. For $j \in [M]$ let T_j be a decision tree that represent $\bigvee_{i=1}^{m_j} T_i^j$. An ℓ-common partial decision tree $\mathcal{T}$ of depth t *represents* the sequence $\left(\bigvee_{i=1}^{m_j} T_i^j\right)_{j=1}^{M}$ if it is an ℓ-common partial decision tree for $T^1,\ldots,T^M$ of depth t.

2.7. Evaluations. The concept of an *evaluation* was introduced by Krajíček, Pudlák, and Woods [11] and is a very convenient tool for proving lower bounds on Frege proof size. The content of this section is standard, and we follow the presentation of Urquhart and Fu [21] while using the notation of Håstad [8]. We need a generalization of previous notions as introduced by Pitassi, Ramakrishnan, and Tan [13].

DEFINITION 2.11 (evaluation). *The set of formulas Γ has a t-*evaluation *φ, mapping formulas from Γ to locally consistent decision trees of depth at most t if the following holds.*

1. *The mapping φ assigns constants (variables) to the corresponding decision trees of depth 0 (of depth 1),*
2. *axioms are assigned to 1-trees,*
3. *if $\varphi(F) = T$, then $\varphi(\neg F)$ is the same decision tree as T except that the leaf-labels are negated, and*
4. *if $F = \bigvee_{i\in[s]} F_i$, then $\varphi(F)$ represents $\bigvee_{i\in[s]} \varphi(F_i)$.*

Eventually we will associate each line of a Frege proof with its own t-evaluation. In order to argue about the proof we require that these different t-evaluations are functionally equivalent as explained in the following. Let us say that two formulas are *isomorphic* if they only differ in the order of the binary $\vee$ connectives.

DEFINITION 2.12 (functional equivalence of evaluations). *Suppose that φ is a t-evaluation defined over a set of formulas Γ, and similarly let φ' be a t-evaluation defined over the set of formulas Γ'. The two t-evaluations φ and φ' are* functionally equivalent *if all isomorphic formulas $F \in \Gamma$ and $F' \in \Gamma'$ satisfy that the decision trees $\varphi(F)$ and $\varphi'(F')$ are functionally equivalent.*

We say that a *Frege proof has a t-evaluation* if each line ν in the proof has a t-evaluation φ^ν for all sub-formulas occurring on ν and for all lines ν,ν' it holds that φ^ν and $\varphi^{\nu'}$ are functionally equivalent.

The following lemma is central. It states that if we have a $t(k)$-evaluation for a Frege proof with $t(k) \le n/16$, then all lines in the proof are represented by 1-trees. As constant false is represented by a 0-tree (Definition 2.11, property 1) it is thus not possible to derive contradiction. Hence any Frege refutation is large, respectively, long, in the case of Frege proofs of bounded line size.

LEMMA 2.13. *Let $n,t \in \mathbb{N}$ such that $t \le n/16$, and suppose that we have a Frege proof of a formula A from the Tseitin formula* $\mathrm{Tseitin}(G_n)$ *defined over the $n\times n$ grid. If this proof has a t-evaluation, then each line in the derivation is mapped to a 1-tree. In particular $A \neq \bot$, that is, contradiction cannot be derived.*

The proof of Lemma 2.13 follows by standard arguments. For completeness we provide a proof in Appendix A.

3. Full restrictions. In this section we introduce a space of restrictions that we use to turn the Tseitin contradiction $\mathrm{Tseitin}(G_{n_1})$ defined over the $n_1 \times n_1$ grid into the Tseitin contradiction $\mathrm{Tseitin}(G_{n_2})$ over the smaller $n_2 \times n_2$ grid. Throughout we

assume that n_1 and n_2 are odd integers such that the mentioned Tseitin formulas are indeed contradictions.

Let $D = \lfloor n_1/n_2 \rfloor$, and partition the columns (rows) of the $n_1 \times n_1$ grid into n_2 almost equal sized sets $\mathcal{Q} = \{Q_i \mid i \in [n_2]\}$ (sets $\mathcal{R} = \{R_i \mid i \in [n_2]\}$) such that each set Q_i (set R_i) contains D or $D+1$ consecutive columns (rows). The *central columns* of $Q \in \mathcal{Q}$ consist of columns $q \in Q$ such that Q has at least $D/8$ columns to the left and right of q, and similarly we let the *central rows* of $R \in \mathcal{R}$ consist of rows $r \in R$ such that R has at least $D/8$ rows above and below r. For each set $Q \in \mathcal{Q}$ (set $R \in \mathcal{R}$) we designate $\Delta = \lfloor D/5 \rfloor$ of the central columns in Q (central rows in R) to be the *center columns* (*center rows*). These center columns (center rows) are chosen evenly spaced from the central columns (central rows), and hence there are always at least 2 central columns (central rows) between each consecutive pair of center columns (center rows).

The partitions $\mathcal{Q}$ and $\mathcal{R}$ naturally induce a partition of the $n_1 \times n_1$ grid into n_2^2 *sub-squares*: sub-square (i,j) is defined as the sub-graph induced by the nodes in $R_i \cap Q_j$. The *central area* of a sub-square (i,j) consists of the nodes in the intersection of the central rows of R_i and the central columns of Q_j. The ℓth *center* of a sub-square is the node in the intersection of the ℓth center row of R_i and the ℓth center column of Q_j. Hence each sub-square has Δ centers. A schematic picture is given in Figure 1.

A restriction chooses one center per sub-square which, eventually, will be the nodes of the smaller $n_2 \times n_2$ grid. For this to make sense we need to explain (1) how to connect such centers by paths and (2) how these paths correspond to variables in the smaller instance.

Let us specify the paths used to connect centers in adjacent sub-squares. Suppose we are given a center c_i and a center c_j in the sub-square below. Since there are at least $2 \cdot \lceil D/8 \rceil \geq \Delta$ rows between the two central areas we can designate for each center c_i a unique row row_i in the middle area.

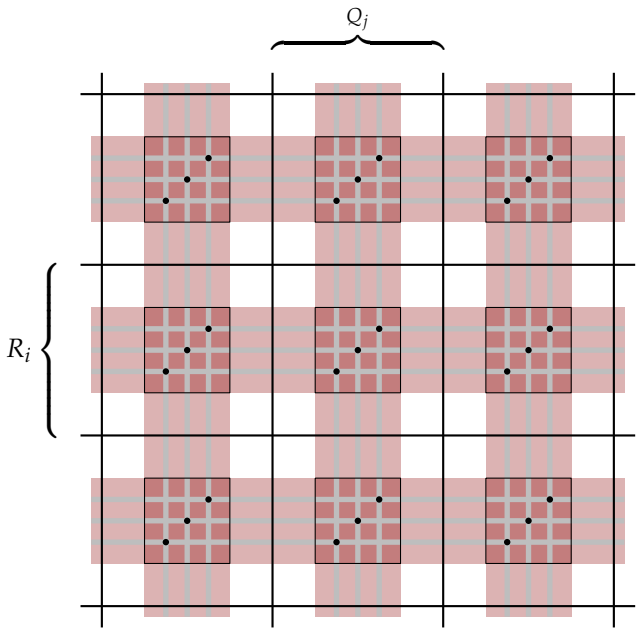

FIG. 1. *Centers and central areas: Central columns and central rows are highlighted red, while center columns and center rows are shaded gray. (Figure available in color online.)*

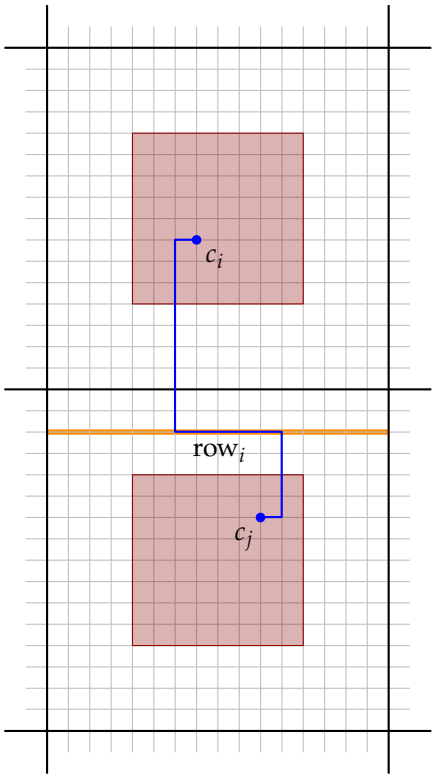

FIG. 2. *A path connecting c_i to a center c_j in the sub-square below with the central areas highlighted red and the designated row* row_i *highlighted orange. (Figure available in color online.)*

To connect c_i to c_j we start at c_i, first go 1 step to the left and then straight down to row_i. This is complemented by starting at c_j, going 1 step to the right, and then straight up to row_i. The appropriate segment from row_i completes the path. An illustration is provided in Figure 2.

Connecting c_i to a center c_j in a sub-square to the right is done in an analogous way: there is a unique column col_i associated with the center c_i. The path consists of five nonempty segments. The first segment consists of the vertical edge down from c_i, while the last segment consists of the vertical edge up from c_j. We add two horizontal segments connecting the first and the last segment to the designated column col_i and use the appropriate middle segment from col_i.

This completes the discussion of the paths. Let us turn our attention to how a path corresponds to a variable in the smaller Tseitin instance. Recall that $n_1 > n_2$ are odd.

A restriction $\sigma \in \Sigma(n_1, n_2) = \Sigma_{\mathcal{Q},\mathcal{R}}(n_1, n_2)$ is defined by one center in each sub-square (the so-called *chosen centers* C_σ of σ) and an assignment σ_0 to the edges of the $n_1 \times n_1$ grid that satisfies the Tseitin formula with 0 charges at the chosen centers and 1 charges at all other nodes. Since the number of chosen centers is odd, by Lemma 2.1, such an assignment exists.

Let us call a path that connects two chosen centers a *chosen path* and note that the set of chosen paths is pairwise edge-disjoint. For each chosen path P we introduce a new variable y_P and define the *full restriction* σ as

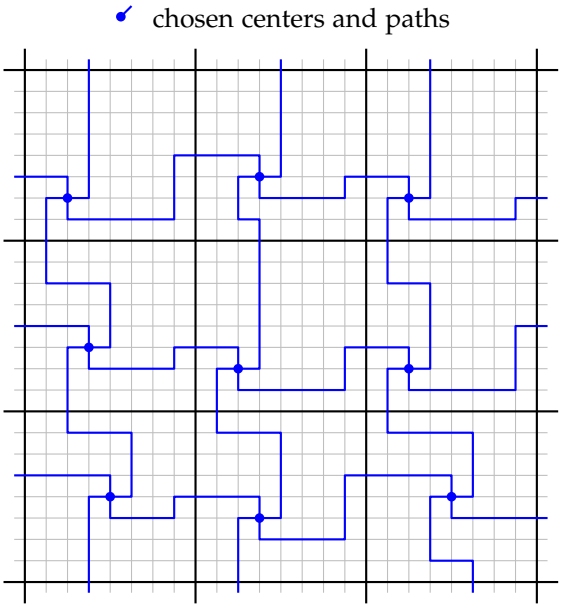

FIG. 3. *A full restriction* σ.

$$\sigma(x_e) = \begin{cases} \sigma_0(x_e) & \text{if } e \text{ is not on a chosen path,} \\ y_P & \text{if } e \text{ is on a chosen path } P \text{ and } \sigma_0(x_e) = 1, \\ \neg y_P & \text{if } e \text{ is on a chosen path } P \text{ and } \sigma_0(x_e) = 0. \end{cases} \tag{2}$$

The value given by σ_0 to a variable that is not on a chosen path is called the *final value.* See Figure 3 for an illustration.

We claim that $\text{Tseitin}(G_{n_1})\lceil_\sigma$ is the Tseitin contradiction $\text{Tseitin}(G_{n_2})$ defined over the smaller $n_2 \times n_2$ grid. Let us check that under σ all the axioms of $\text{Tseitin}(G_{n_1})$ are either mapped to true (and can be removed) or to an axiom of the smaller instance $\text{Tseitin}(G_{n_2})$:

- The axioms of a node v not on a chosen path are satisfied since σ_0 assigns an odd number of incident edges to 1.
- The axioms of an interior node v of a chosen path P are reduced to tautologies: the axioms are true independent of the value of y_P since flipping the value of y_P changes the value of two variables incident to v.
- The axioms of a chosen center $v \in C_\sigma$ turn into axioms of the smaller instance. Note that the charge is still 1: the restriction σ_0 assigns an even number of edges incident to v to 0, and hence there is an even number of negated variables $\neg x_P$ incident to v. Thus the constraint $\sum_{e \ni v} \sigma(x_e) = 1 \mod 2$ is equivalent to $\sum_{P \ni v} y_P = 1 \mod 2$.

Note that a full restriction σ is really an *affine restriction* in the vocabulary of Rossman, Servedio, and Tan [17] since σ not only assigns values to variables but also identifies old variables with new variables or the negations thereof.

3.1. A distribution over full restrictions. For odd integer k sample $\sigma \sim \mathcal{D}_k\big(\Sigma(n_1, n_2)\big)$ as follows. Uniformly at random choose a set of k centers from the set of all k-subsets of the Δn_2^2 centers with the property that every sub-square contains $(1 \pm 0.01)k/n_2^2$ centers. These are the so-called *alive* centers. In each sub-square the

alive center with the lowest numbered row becomes a *chosen center.* Sample uniformly at random an assignment σ_0 from the space of solutions to the Tseitin formula with charges 0 at the chosen centers and 1 at all other nodes. Define σ from σ_0 as in (2).

3.2. Differences from previous work. We use a very similar space of random restriction compared to [8] but make two changes. First, we make the number k of alive centers *independent* of the depth of the considered decision trees: in [8] the number of alive centers is $Cn_2^2 s$ for s the depth of the considered decision trees, whereas we have $Cn_2^2 \log n$ alive centers (independent of s). This change allows us to prove a multiswitching lemma. Second, we define full restrictions by designating rows and columns that are unique to a single center instead of pairs of centers as done in [8].

While the first change already appeared in the preliminary version of this work [9], the second modification is presented here for the first time. This change allows us to further strengthen the size lower bound by a factor 2 in the second exponent.

3.3. Decision trees and full restrictions. In this section we define $T\restriction_\sigma$: the restriction of a decision tree T by a full restriction σ. Recall that for a partial assignment α the decision tree $T\restriction_\alpha$ only consists of branches pairwise locally consistent with α (see subsection 2.6). In the following we define a notion of local consistency for full restrictions so that we can define $T\restriction_\sigma$ as for partial assignments.

While the initial decision tree T queries variables x_e, the resulting decision tree $T\restriction_\sigma$ queries path variables y_P defined on the smaller grid. The idea of pairwise local consistency of a full restrictions $\sigma \in \Sigma(n_1, n_2)$ and a partial assignment τ is not complicated: we want to ensure that if a variable is assigned a constant by both τ and σ, then these agree. Furthermore we require that the restriction induced by τ on the smaller $n_2 \times n_2$ grid is locally consistent. The following definition formalizes this notion.

DEFINITION 3.1 (pairwise local consistency for full restrictions). *Consider the Tseitin formula* $\mathrm{Tseitin}(G_{n_1})$ *defined over the* $n_1 \times n_1$ *grid, denote by* X *the variables of* $\mathrm{Tseitin}(G_{n_1})$*, let* $\tau\colon X \to \{0,1,*\}$ *be a partial assignment, and denote by* $\sigma \in \Sigma(n_1, n_2)$ *a full restriction. We say that* τ *and* σ *are* pairwise locally consistent *if the following holds.*

1. *For all variables* $x \in X$ *it holds that if* x *is assigned to a constant by both* τ *and* σ*, then* $\sigma(x) = \tau(x)$*.*
2. *Suppose* σ *maps* x_1 *and* x_2 *to the same variable. If* $x_1, x_2 \in \tau^{-1}\big(\{0,1\}\big)$*, then*
   (a) $\tau(x_1) = \tau(x_2)$ *if* $\sigma(x_1) = \sigma(x_2)$*,*
   (b) $\tau(x_1) = \neg\tau(x_2)$ *if* $\sigma(x_1) = \neg\sigma(x_2)$*.*
3. *If* τ_σ *denotes the minimal partial assignment such that for all* $x \in \mathrm{dom}(\tau)$ *it holds that*

$$\tau_\sigma\big(\sigma(x)\big) = \tau(x),$$

   then we require that τ_σ *is locally consistent with respect to the* $n_2 \times n_2$ *grid as defined in Definition* 2.2*.*

Note that property 2 of the above definition ensures that τ_σ is well defined, that is, it ensures that there are no two edges on a chosen path P such that τ_σ assigns 0 and 1 to y_P. Note that τ_σ, as defined in property 3 of Definition 3.1, can simply be thought of as the restriction that τ induces on the smaller grid. In case a more explicit description of τ_σ is sought, the partial assignment τ_σ may equivalently be defined as

$$\tau_\sigma(y_P) = \begin{cases} \tau(x_e) & \text{if there is an edge } e \in P \text{ such that } x_e \in \tau^{-1}\big(\{0,1\}\big) \text{ and} \\ & \sigma(x_e) = y_P, \\ \neg\tau(x_e) & \text{if there is an edge } e \in P \text{ such that } x_e \in \tau^{-1}\big(\{0,1\}\big) \text{ and} \\ & \sigma(x_e) = \neg y_P, \\ * & \text{otherwise.} \end{cases}$$

With the notion of pairwise local consistency for full restrictions σ in place we are ready to define the restriction of a decision tree T by a full restriction σ, denoted by $T\restriction_\sigma$, as follows. Consider a branch $\tau \in T$ that is pairwise locally consistent with σ. The restricted decision tree $T\restriction_\sigma$ contains the branch τ_σ as defined in property 3 of Definition 3.1 for each such τ. By definition each such branch τ_σ is locally consistent with respect to the smaller grid. Furthermore, these branches fit into a tree-structure once duplicate queries to the same variable are removed.

However, if there is *no* branch $\tau \in T$ pairwise locally consistent with σ, then the above restriction fails to return a decision tree. For shallow trees we are guaranteed to obtain locally consistent decision trees as summarized in the following.

LEMMA 3.2. *Let T be a decision tree on the variables of the Tseitin formula* $\mathrm{Tseitin}(G_{n_1})$*, and denote by $\sigma \in \Sigma(n_1, n_2)$ a full restriction. If* $\mathrm{depth}(T) = t \leq n_2/4$*, then $T\restriction_\sigma$ is a locally consistent decision tree of depth at most t.*

Proof. We need to argue that there is a branch $\tau \in T$ pairwise locally consistent with σ. We construct τ inductively starting with v being the root node of T. Denote by τ^v the partial assignment from the root to v, and let τ^v_σ be the assignment induced by τ^v on the smaller grid as defined in property 3 of Definition 3.1. Suppose v is an internal node labeled x_e. If σ assigns x_e to a constant, then recurse on the node the out-edge of v labeled $\sigma(x_e)$ points to. Otherwise σ assigns x_e to a variable.

1. If τ^v_σ assigns the variable y_P that $\sigma(x_e)$ maps to, then recurse on the node the out-edge of v labeled
   (a) $\tau^v_\sigma(y_P)$ points to, assuming $\sigma(x_e) = y_P$, or
   (b) $\neg\tau^v_\sigma(y_P)$ points to, assuming $\sigma(x_e) = \neg y_P$.
2. Else τ^v_σ does not assign y_P. Let $b \in \{0,1\}$ such that $\tau^v_\sigma \cup \{y_P \mapsto b\}$ is a locally consistent assignment on the $n_2 \times n_2$ grid. Recurse on the node the out-edge of v labeled b points to.

If the node v is a leaf, then we have found a branch $\tau = \tau^v$ in T that is pairwise consistent with σ.

The above process may fail in step 2 if there is no $b \in \{0,1\}$ such that $\tau^v_\sigma \cup \{y_P \mapsto b\}$ is locally consistent. Since $\mathrm{depth}(T) \leq n_2/4$ by Lemma 2.3 there is always such a choice. The statement follows. ∎

LEMMA 3.3. *Let T and T' be two functionally equivalent decision trees on the variables of the Tseitin formula* $\mathrm{Tseitin}(G_{n_1})$*, and denote by $\sigma \in \Sigma(n_1, n_2)$ a full restriction. If* $\mathrm{depth}(T), \mathrm{depth}(T') \leq n_2/8$*, then $T\restriction_\sigma$ and $T'\restriction_\sigma$ are locally equivalent.*

Proof. Consider a b-branch $\tau \in T$ pairwise locally consistent with σ. Since τ and σ are pairwise consistent, the decision tree $T\restriction_\sigma$ contains the b-branch τ_σ as defined in property 3 of Definition 3.1.

As T and T' are functionally equivalent, the decision tree $T'\restriction_\tau$ is a b-tree. Since for all $x \in \mathrm{dom}(\tau)$ it holds that $\tau_\sigma\big(\sigma(x)\big) = \tau(x)$, it follows that the decision tree $\big(T'\restriction_\sigma\big)\restriction_{\tau_\sigma}$ is also a b-tree. Here we rely on Corollary 2.7 to argue that the decision tree $T'\restriction_\sigma$ restricted by τ_σ is well defined. The statement follows. ∎

3.4. Evaluations and full restrictions. Given a t-evaluation φ for Γ and a full restriction σ, we denote by $\varphi\restriction_\sigma$ the t-evaluation for $\Gamma\restriction_\sigma$ defined by $\varphi\restriction_\sigma(F\restriction_\sigma) = \varphi(F)\restriction_\sigma$ for all $F \in \Gamma$.

Consider a Frege proof of depth d, and for a line ν in the proof let us denote by Γ^ν the set of sub-formulas occurring on line ν. We intend to construct a sequence of full restrictions $\sigma_1, \sigma_2, \ldots, \sigma_d$ with the following property. Denote by σ_k^* the concatenation of the first k restrictions in the sequence, and let $t(k)$ be a function growing with k to be fixed—it will depend on the application. From the sequence of restrictions we require that all sub-formulas occurring in the proof of depth at most k have functionally equivalent $t(k)$-evaluations after hitting them with the restriction σ_k^*. In more detail, for every line ν we want a $t(k)$-evaluation for the formulas in

$$\Gamma_k^\nu = \{F\restriction_{\sigma_k^*} \mid F \in \Gamma^\nu \wedge \operatorname{depth}(F) \le k\} \tag{3}$$

and require that any pair of these $t(k)$-evaluations is functionally equivalent. We construct these $t(k)$-evaluations by induction on k. To ensure that the domain of the t-evaluations does not decrease when we apply another restriction we rely on the following lemma.

LEMMA 3.4. *Let $n_1, n_2, t \in \mathbb{N}$ such that $n_2 \le n_1$ and $t \le n_2/8$. Denote by φ a t-evaluation defined over the set of formulas Γ, let φ' be a t-evaluation defined over the set of formulas Γ', and denote by $\sigma \in \Sigma(n_1, n_2)$ a full restriction. If φ and φ' are functionally equivalent, then $\varphi\restriction_\sigma$ and $\varphi'\restriction_\sigma$ are also functionally equivalent t-evaluations with domains $\operatorname{dom}(\varphi\restriction_\sigma) = \Gamma\restriction_\sigma$ and $\operatorname{dom}(\varphi'\restriction_\sigma) = \Gamma'\restriction_\sigma$.*

Proof. Fix a formula $F \in \Gamma$, and let $T = \varphi(F)$. By definition we have that $\varphi\restriction_\sigma(F\restriction_\sigma) = T\restriction_\sigma$. By Lemma 3.2 we see that $T\restriction_\sigma$ is a locally consistent decision tree with respect to the $n' \times n'$ grid and it holds that $\operatorname{depth}(T\restriction_\sigma) \le t$. We need to check that $T\restriction_\sigma$ satisfies properties 1–4 of Definition 2.11. Properties 1–3 are immediate since the process of a restriction neither depends on the labels of the leaves nor does it change them.

We are left to show property 4. Suppose $F = \bigvee_{i\in[m]} F_i$, let $T_i = \varphi(F_i)$, and consider a b-branch τ in T that is pairwise locally consistent with σ. Since φ is a t-evaluation, it holds that if $b = 0$, then $T_i\restriction_\tau = 0$ for all $i \in [m]$ and if $b = 1$, then there is an $i \in [m]$ such that $T_i\restriction_\tau = 1$.

Since τ and σ are locally consistent the decision tree $T\restriction_\sigma$ contains the b-branch τ_σ as defined in property 3 of Definition 3.1. Recall that restricting first by σ and then τ_σ sets the variables in $\operatorname{dom}(\tau)$ to the same constants as τ does, that is, for $x \in \operatorname{dom}(\tau)$ we have that $\tau_\sigma(\sigma(x)) = \tau(x)$. Hence if $T_i\restriction_\tau$ is a b-tree, then the decision tree $T_i\restriction_\sigma$ under the restriction τ_σ is also a b-tree. For the last statement we relied on Corollary 2.7. This yields property 4.

The claimed functional equivalence follows from Lemma 3.3. This establishes the claim. ∎

The important step of the argument is to use a switching lemma to extend the domain of the $t(k)$-evaluation from Γ_k^ν to Γ_{k+1}^ν. We give that argument in the next section.

4. Proofs of the main theorems. We first reprove the main theorem of [8] with improved parameters.

THEOREM 4.1. *For $d = O\big(\frac{\log n}{\log\log n}\big)$ it holds that any depth-d Frege refutation of the Tseitin formula $\operatorname{Tseitin}(G_n)$ with odd charges at all nodes of the $n \times n$ grid requires size*

$$\exp\Big(\Omega\big(n^{1/d}/\log^4 n\big)\Big).$$

As outlined in subsection 3.4 we construct a t-evaluation for all sub-formulas occurring in a short and shallow Frege proof. By Lemma 2.13 we then conclude that all shallow Frege proofs of the Tseitin contradiction must be long. For the total size lower bound we in fact do not create distinct t-evaluations per line but rather a single one, used on each line. Such a t-evaluation is clearly functionally equivalent and hence satisfies our needs. In order to extend the t-evaluation to larger depth we use the following switching lemma.

LEMMA 4.2 (switching lemma). *There are absolute constants $A, C, n_0 > 0$ such that for integer $n \geq n_0$ the following holds. Let $k, m, n', s, t \in \mathbb{N}^+$ satisfy $n/n' \geq At\log^4 n$, $k = n'^2(1 \pm 0.01)C\log n'$ be odd, and $t \leq s \leq n'/32$. Then for any decision trees $T_1, \ldots, T_m$ of depth at most t querying edges of the $n \times n$ grid it holds that if $\sigma \sim \mathcal{D}_k\big(\Sigma(n, n')\big)$, then the probability that $\bigvee_{i=1}^m T_i\restriction_\sigma$ cannot be represented by a decision tree of depth s is bounded by*

$$\left(\frac{At\log^4 n'}{n/n'}\right)^{s/64}.$$

The proof of Lemma 4.2 is given in section 6. Let us verify that Theorem 4.1 indeed follows from Lemma 4.2.

Proof of Theorem 4.1. Suppose π is a refutation of size $N \leq \exp\big(n^{1/d}/c_1\log^4 n\big)$ for some large constant $c_1 > 0$. Denote by Γ the set of sub-formulas occurring in π. We proceed by induction on $i = 0, 1, 2, \ldots, d-1$ as follows.

Assume by induction that we are given a sequence of restrictions $\sigma_1, \sigma_2, \ldots, \sigma_{i-1}$, denote by σ^*_{i-1} the composition thereof, and suppose that we have a t_i-evaluation φ_i defined on all formulas

$$\Gamma_i = \{F\restriction_{\sigma^*_{i-1}} \mid F \in \Gamma \text{ and } \operatorname{depth}(F) \leq i\} \tag{4}$$

of original depth at most i. Sample a full restriction $\sigma_i \in \mathcal{D}_k\big(\Sigma(n_i, n_{i+1})\big)$, and extend φ_i to a t_{i+1}-evaluation φ_{i+1} defined on all formulas Γ_{i+1} of original depth at most $i+1$.

Let us implement the above plan. We choose $t_0 = 1$ and $n_0 = n$, set $s = 152\log N$, and let $t_i = s$ and $n_i = \lfloor n_{i-1}/4At_{i-1}\log^4 n_{i-1}\rfloor$ for $i \in [d]$. The constant A is chosen as in Lemma 4.2.

Initially, for $i = 0$, each formula is either a variable which is mapped by φ_0 to the depth-1 decision tree evaluating it or a constant which is mapped to the appropriate depth-0 decision tree. For the inductive step we may assume that we have a t_i-evaluation φ_i for all formulas in Γ_i.

Sample $\sigma_i \in \mathcal{D}_k\big(\Sigma(n_i, n_{i+1})\big)$. We need to extend the φ_i to a t_{i+1}-evaluation φ_{i+1} defined on all formulas Γ_{i+1} of original depth at most $i+1$. Consider any such formula $F \in \Gamma$. We define φ_{i+1} as follows.

1. If F is of depth at most $\operatorname{depth}(F) \leq i$, then let $\varphi_{i+1}(F\restriction_{\sigma^*_i}) = \varphi_i(F\restriction_{\sigma^*_{i-1}})\restriction_{\sigma_i}$. By Lemma 3.4 the restricted t_i-evaluation $\varphi_i\restriction_{\sigma_i}$ is defined on such formulas.
2. If $F = \neg F'$ is of depth $i+1$, then $\varphi_{i+1}\big(F\restriction_{\sigma^*_i}\big)$ is defined in terms of $\varphi_{i+1}\big(F'\restriction_{\sigma^*_i}\big)$ by negating all the leaf-labels.
3. If $F = \bigvee_j F_j$ is of depth $i+1$ and each F_j is of depth at most $\operatorname{depth}(F_j) \leq i$, then we appeal to Lemma 4.2 to obtain a decision tree T of depth $\operatorname{depth}(T) \leq t_{i+1}$ representing $\bigvee_j T_j$ for $T_j = \varphi_i(F_j\restriction_{\sigma^*_{i-1}})$.

The only place where the extension might fail is in item 3. By Lemma 4.2 and our choice of parameters we see that the failure probability is bounded by

$$\left(\frac{At_i \log^4 n_{i+1}}{n_i/n_{i+1}}\right)^{s/76} \leq \left(\frac{At_i \log^4 n_{i+1}}{2At_i \log^4 n_i}\right)^{2\log N} \leq N^{-2} \ . \tag{5}$$

We union bound over at most N sub-formulas and thus succeed with probability $1 - N^{-1}$ in every step.

This completes the induction, and we thus obtain a refutation of $\mathrm{Tseitin}(G_{n_d})$ with a t_d-evaluation φ_d. Note that $n_d \geq n/\log^{d-1}(N)(c_2 \log^4 n)^d$ for some constant $c_2 > 0$. On the other hand we have $t_d = 152 \log N$. Thus if $\log N \leq n^{1/d}/c_1 \log^4 n$ for constant $c_1 > 0$ large enough, then we get a contradiction to Lemma 2.13. The claimed lower bound follows. ∎

We turn our attention to the main result of the paper.

THEOREM 4.3. *For any Frege refutation of the Tseitin formula* $\mathrm{Tseitin}(G_n)$ *with odd charges at all nodes of the* $n \times n$ *grid the following holds. If each line of the refutation is of size* M *and depth* d*, then the number of lines in the refutation is*

$$\exp\left(\Omega\left(\frac{n}{\left((\log n)^{O(1)} \log M\right)^d}\right)\right).$$

Note that the lower bounds obtained from Theorem 4.3 are much stronger than the lower bounds obtained from Theorem 4.1 if the line size M is bounded. For example, if $M = O\big(n^{\mathrm{polylog}(n)}\big)$ and the depth $d = o\big(\frac{\log n}{\log \log n}\big)$, then we obtain a length lower bound of $\exp\big(n^{1-o(1)}\big)$. This lower bound is essentially optimal since there is a resolution refutation of length $2^{O(n)}$.

The strategy of the proof is similar to the proof of Theorem 4.1: we again build a t-evaluation for a supposed Frege proof. The main difference is that instead of creating a single t-evaluation for the entire proof we in fact independently create t-evaluations for each line. These t-evaluations turn out to be functionally equivalent, as defined in Definition 2.12. We obtain the claimed bounds by an appeal to Lemma 2.13.

Suppose we are given a Frege refutation of the Tseitin principle defined over the $n \times n$ grid consisting of N lines, where each line is a formula of size M and depth d. Denote by Γ^ν the set of sub-formulas of line $\nu \in [N]$ in the proof. We construct a sequence of restrictions $\sigma_1, \sigma_2, \ldots, \sigma_d$ such that all formulas of depth at most $\eta \in [d]$ have functionally equivalent $t(\eta)$-evaluations if hit by the concatenation σ_η^* of the first η restrictions in the sequence, where $t(\eta)$ is some function dependent on η to be fixed later. That is, for every line ν we have a $t(\eta)$-evaluation φ_η^ν for all formulas in the set

$$\Gamma_\eta^\nu = \{F\lceil_{\sigma_\eta^*} \mid F \in \Gamma^\nu \wedge \mathrm{depth}(F) \leq \eta\}, \tag{6}$$

and all these $t(\eta)$-evaluations are functionally equivalent. In addition to these $t(\eta)$-evaluations, for each line ν we also maintain a decision tree $\mathcal{T}_\eta(\nu)$. We maintain the property that $\mathcal{T}_\eta(\nu)$ is a t-common partial decision tree for all $t(\eta)$-evaluations $\varphi_\eta^\nu(\Gamma_\eta^\nu)$ of bounded depth.

These common partial decision trees $\mathcal{T}_\eta(\nu)$ are useful in order to extend the $t(\eta)$-evaluations φ_η^ν to larger depths. In each such step, increasing η, we apply for each branch τ from $\mathcal{T}_\eta(\nu)$ the following multiswitching lemma to the set of decision trees $\varphi_\eta^\nu(\Gamma_\eta^\nu)\lceil_\tau$ of depth at most $t(\eta)$. We then extend $\mathcal{T}_\eta(\nu)$ in each leaf τ by the common partial decision tree from the lemma to obtain $\mathcal{T}_{\eta+1}(\nu)$ of slightly larger depth.

LEMMA 4.4 (multiswitching lemma). *There are constants $A, c_1, c_2, n_0 > 0$ such that for integer $n \geq n_0$ the following holds. Let $k, M, n', s, t \in \mathbb{N}^+$ satisfy $n/n' \geq At\log^{c_1} n$, $k = n'^2(1 \pm 0.01)C\log n'$ be odd, and $t \leq s \leq n'/32$. For $m_1, \ldots, m_M \in \mathbb{N}^+$ and any decision trees T_i^j of depth at most t, where $j \in [M]$ and $i \in [m_j]$, it holds that if $\sigma \sim \mathcal{D}_k\big(\Sigma(n, n')\big)$, then the probability that $\big(\bigvee_{i=1}^{m_j} T_i^j\restriction_\sigma\big)_{j=1}^{M}$ cannot be represented by an ℓ-common partial decision tree of depth s is bounded by*

$$M^{s/\ell}\left(\frac{At\log^{c_1} n}{n/n'}\right)^{s/c_2}.$$

We defer the proof of Lemma 4.4 to section 7. In the following we explain how Theorem 4.3 follows from Lemma 4.4.

We apply Lemma 4.4 with mostly the same parameters. Let us fix these. We choose $\ell = t = \log M$, let $n_0 = n$, and set $n_\eta = \lfloor n_{\eta-1}/A_1 \cdot t \cdot \log^{c_1} n_{\eta-1}\rfloor$ for $\eta \in [d]$ and a sufficiently large constant A_1. The parameter s depends on η and is fixed to $s = s_\eta = 2^{\eta-1}\log N$. With these parameters in place we can finally also fix $t(\eta) = \sum_{i\leq\eta} s_i + \log M \leq 2^\eta \log N + \log M$.

LEMMA 4.5. *Suppose that for every line $\nu \in [N]$ we have functionally equivalent $t(\eta-1)$-evaluations $\varphi_{\eta-1}^\nu$ for formulas in $\Gamma_{\eta-1}^\nu$ along with a t-common partial decision tree $\mathcal{T}_{\eta-1}(\nu)$ for $\varphi_{\eta-1}^\nu(\Gamma_{\eta-1}^\nu)$ of depth $\sum_{i<\eta} s_i$. Suppose that $t(\eta) \leq n_\eta/16$. For $\sigma_\eta \sim \mathcal{D}_k\big(\Sigma(n_{\eta-1}, n_\eta)\big)$ with probability $1-N^{-1}$, for every line $\nu \in [N]$ there are functionally equivalent $t(\eta)$-evaluations φ_η^ν for formulas in Γ_η^ν and a t-common partial decision tree $\mathcal{T}_\eta(\nu)$ for $\varphi_\eta^\nu(\Gamma_\eta^\nu)$ of depth $\sum_{i\leq\eta} s_i$.*

Proof. Let us first extend the common partial decision trees and then explain how to obtain the evaluation φ_η^ν for different lines $\nu \in [N]$.

The interesting formulas of original depth η to consider are the ones with a top $\vee$ gate. Let us fix a line $\nu \in [N]$ and consider all sub-formulas $\{F^j = \bigvee_{i=1}^{m_j} F_i^j\}_{j=1}^{M_\nu}$ of line ν of original depth η with a top $\vee$ gate under the restriction $\sigma_{\eta-1}^*$. As the original depth of every formula F_i^j is at most $\mathrm{depth}(F_i^j) \leq \eta - 1$, all these formulas are in the domain of the evaluation $\varphi_{\eta-1}^\nu$. Let us further fix a branch τ in $\mathcal{T}_{\eta-1}(\nu)$ and recall that all decision trees $\varphi_{\eta-1}^\nu(F_i^j)\restriction_\tau$ are of depth at most t.

For every $\nu \in [N]$ and branch τ of $\mathcal{T}_{\eta-1}(\nu)$ we apply Lemma 4.4 to the set of formulas $F_i^j\restriction_\tau$ with associated trees $\varphi_{\eta-1}^\nu(F_i^j)\restriction_\tau$ of depth at most t. The probability of failure of a single application is bounded by

$$M^{s_\eta/\ell}\left(\frac{At\log^{c_1} n_{\eta-1}}{n_{\eta-1}/n_\eta}\right)^{s_\eta/c_2} \leq M^{s_\eta/\log M}\left(\frac{At\log^{c_1} n_{\eta-1}}{A_1 t\log^{c_1} n_{\eta-1}}\right)^{s_\eta/c_2} \leq 2^{-4s_\eta} = N^{-2^{\eta+1}}, \tag{7}$$

assuming that the constant A_1 is large enough. As we invoke Lemma 4.4 at most $N \cdot 2^{\sum_{i<\eta} s_i} \leq N^{2^\eta}$ times, by a union bound, with probability at least $1 - N^{-1}$, there is a full restriction σ_η such that for every line $\nu \in [N]$ and every branch $\tau \in \mathcal{T}_{\eta-1}(\nu)$ we get a t-common partial decision tree of depth at most s_η for the formulas $(F^j\restriction_{\tau\sigma_\eta})_{j=1}^{M_\nu}$. Let us denote this common decision tree by $\mathcal{T}(\nu, \tau)$ and attach it to $\mathcal{T}_{\eta-1}(\nu)$ at the leaf τ to obtain $\mathcal{T}_\eta(\nu)$. The trees $\mathcal{T}_\eta(\nu)$ are of depth at most $\sum_{i\leq\eta} s_i$ as required.

Let us explain how to define the evaluation φ_η^ν for a fixed line $\nu \in [N]$. Consider any formula F in Γ_η^ν.

- If F is of depth less than η, then F is in the domain of $\varphi_{\eta-1}^\nu$ and we can appeal to Lemma 3.4.

- If $F = \neg F'$ is of depth η, then $\varphi^\nu_\eta(F)$ is defined from $\varphi^\nu_\eta(F')$ by negating the labels at the leaves.
- For $F = \bigvee_i F_i$ of depth η we use the previously constructed common partial decision trees. We define $\varphi^\nu_\eta(F)$ to be the decision tree whose first $\sum_{i\le\eta} s_i$ levels are equivalent to $\mathcal{T}_\eta(\nu)$ followed by t levels unique to F obtained from the multiswitching lemma.

Let us check that the decision trees $\mathcal{T}_\eta(\nu)$ are indeed t-common partial decision trees for $\varphi^\nu_\eta(\Gamma^\nu_\eta)$. By construction this clearly holds for formulas of depth η with a top $\vee$ gate. Since $\mathcal{T}_\eta(\nu)$ is equivalent to $\mathcal{T}_{\eta-1}(\nu)$ on the upper levels, and restrictions only decrease the depth of decision trees, by the initial assumptions this also holds for formulas of depth less than η. As the $t(\eta)$-evaluations of formulas of depth η with a top $\neg$-gate are defined in terms of formulas of depth less than η, we also see that $\mathcal{T}_\eta(\nu)$ is a t-common partial decision tree for such formulas.

Last we need to check that each φ^ν_η is a $t(\eta)$-evaluation plus that these are pairwise functionally equivalent.

By Lemma 3.4 all the properties hold for formulas of depth less than η. Let us verify the $t(\eta)$-evaluation properties for formulas of depth η. That is, we need to check that constants (variables) are mapped to the corresponding decision trees of depth 0 (of depth 1), that axioms are assigned to 1-trees, that the negation of a formula is assigned to the same decision tree as the formula is except that the leaf-labels are negated, and that if a formula F is the or of some sub-formulas, then the decision tree that F is mapped to represents the or of the decision trees that the sub-formulas are mapped to. Let us check these properties.

Property 1 of Definition 2.11 is immediate as $\eta > 0$. As we only consider locally consistent decision trees as defined in Definition 2.5, property 2 also follows. Further, property 3 is satisfied by construction. Property 4 can be established by checking the property for each branch τ in $\mathcal{T}_{\eta-1}(\nu)$ separately; for a fixed τ we see by Lemma 4.4 that this indeed holds.

Finally we need to establish that two $t(\eta)$-evaluations φ^ν_η and $\varphi^{\nu'}_\eta$ are functionally equivalent for formulas of depth η. By the inductive hypothesis isomorphic formulas with a top $\neg$-gate are functionally equivalent. Hence we are left to check functional equivalence for isomorphic formulas of depth η with a top $\vee$ gate.

Let $F = \bigvee_i F_i$ and $F' = \bigvee_i F'_i$ be two isomorphic formulas from Γ^ν_η and $\Gamma^{\nu'}_\eta$, respectively. For the sake of contradiction suppose that $\varphi^\nu_\eta(F)\restriction_\tau = 1$, but $\varphi^{\nu'}_\eta(F')\restriction_\tau = 0$ for some assignment τ. In the following we use that $t(\eta) \le n_\eta/16$ to argue that there are locally consistent branches as claimed.

By property 2 we know that for some F_i it holds that $\varphi^\nu_\eta(F_i)\restriction_\tau = 1$. Since formulas F and F' are isomorphic formulas, we know that there is an F'_j such that F_i and F'_j are isomorphic formulas. As such formulas have functionally equivalent decision trees (by induction and Lemma 3.4) we get $\varphi^{\nu'}_\eta(F'_j)\restriction_\tau = 1$. But this cannot be as by property 4 of a $t(\eta)$-evaluation this implies that $\varphi^{\nu'}_\eta(F')\restriction_\tau = 1$. This establishes that the different $t(\eta)$-evaluations are functionally equivalent, as required. ∎

With all pieces in place we are ready to prove Theorem 4.3.

Proof of Theorem 4.3. For the sake of contradiction suppose that we are given a proof of length $N = \exp(n/((\log n)^c \log M)^d)$ for some constant $c > 0$. We may assume that $M \le \exp(n^{1/d-1/d(d-1)})$, as otherwise we can apply Theorem 4.1.

In order to create the functionally equivalent $t(\eta)$-evaluations φ^ν for each line $\nu \in [N]$ we consecutively apply Lemma 4.5 d times. We start with the evaluation φ^ν_0

which maps constants to the appropriate depth-0 decision tree and variables to the corresponding depth-1 decision trees. The common partial decision trees $\mathcal{T}_0(\nu)$ are all empty.

After applying Lemma 4.5 d times we are left with a $t(d)$-evaluation for the proof. We need to ensure that $t(d)$ is upper bounded by the dimension of the final grid, $t(d) \leq 2^d \log N + \log M$, while the final side length of the grid is $n \cdot (2A_1(\log n)^{c_1} \log M)^{-d}$. For our choice of N and the assumption on M this indeed holds and by Lemma 2.13 the theorem follows. ∎

5. Switching lemma: Proof outline and further preliminaries. In this section we revisit full restrictions and define some bookkeeping objects that are used in the proof of the switching lemma. The actual proof of Lemma 4.2 is carried out in section 6.

In order to motivate the following definitions we give a very high-level proof outline of Lemma 4.2 in the following section.

5.1. High level proof outline. We are given m decision trees T_i of depth at most $\text{depth}(T_i) \leq t$ that query the edges of the $n \times n$ grid. We sample a full restriction $\sigma \sim \mathcal{D}_k\big(\Sigma(n, n')\big)$ and want to argue that the probability that there is no decision tree of depth s representing $\bigvee_{i=1}^m T_i\lceil_\sigma$ is exponentially small in s.

We bound this probability by constructing the so-called *extended canonical decision tree* $\mathcal{T}$ that represents $\bigvee_{i=1}^m T_i\lceil_\sigma$ and bounding the probability that $\mathcal{T}$ is of depth $\text{depth}(\mathcal{T}) > s$.

For now we can think of $\mathcal{T}$ being constructed like the canonical decision tree: proceed in stages. In each stage a branch τ of $\mathcal{T}$ is extended by querying the variables of the first 1-branch ψ in the decision trees $T_1\lceil_{\sigma\tau}, T_2\lceil_{\sigma\tau}, \ldots, T_m\lceil_{\sigma\tau}$. Once queried we check in each new leaf of the tree whether we traversed the path ψ. If so, then we label the leaf with a 1 and otherwise we continue with the next stage. If there are no 1-branches left, then we label the leaf with a 0.

It is not so hard to see that this process results in a decision tree $\mathcal{T}$ that represents $\bigvee_{i=1}^m T_i\lceil_\sigma$: for each leaf τ of $\mathcal{T}$ that is labeled 1 it holds that there is an $i \in [m]$ such that $T_i\lceil_{\sigma\tau} = 1$ and if the branch τ is labeled 0, then for all $i \in [m]$ we have that $T_i\lceil_{\sigma\tau}$ is a 0-tree. It remains to argue that the decision tree $\mathcal{T}$ is of depth at most s except with probability exponentially small in s.

We analyze this event using the labeling technique of Razborov [16]. The idea of this technique is to come up with an (almost) bijection F mapping restrictions σ that give rise to an extended canonical decision tree $\mathcal{T}$ of depth $\text{depth}(\mathcal{T}) \geq s$ to a space of restrictions Σ^* that is much smaller than the space Σ from which we sampled the full restriction σ. If we manage to exhibit such an F to a space Σ^* that is an exponential in s factor smaller than Σ, then the claimed upper bound on the failure probability follows.

Let us sketch the construction of such an (almost) bijection. Consider a full restriction σ and decision trees $T_1, \ldots, T_m$ that give rise to an extended canonical decision tree $\mathcal{T}$ of depth s. Let $\tau \in \mathcal{T}$ be a 0-branch of length s, and denote by $\psi_1, \ldots, \psi_g$ the 1-branches used in the stages that constructed the branch τ. We know that τ does not agree with any of $\psi_1, \ldots, \psi_g$ as otherwise τ would end in a 1-leaf. Let $\tau_1, \ldots, \tau_g \subseteq \tau$ be the partial assignments to the variables of the corresponding ψ_j, that is, the domain $\text{dom}(\tau_j) = \text{dom}(\psi_j)$ of these assignments for all $j \in [g]$ are equal.

Razborov [16] maps σ to $F(\sigma) = \sigma^*$: the restriction σ^* is σ composed with the partial assignments $\psi_1, \ldots, \psi_g$, that is, we somehow "add" the restrictions ψ_j to σ such that the branch ψ_1 is traversed by the resulting restriction σ^*. The crucial insight

is that with the help of the shallow decision trees $T_1, \ldots, T_m$ the mapping F can be cheaply inverted by "redoing" the construction of τ: we proceed in stages $j = 1, \ldots, g$. At the beginning of stage j we assume that we have the restriction $\sigma^*_{\geq j}$ which is σ composed with $\tau_1, \ldots, \tau_{j-1}, \psi_j, \ldots, \psi_g$ such that the branch ψ_j is traversed. Since $\sigma^*_{\geq j}$ by construction traverses ψ_j we can identify ψ_j for free: it is the first 1-branch in $T_1, \ldots, T_m$ traversed by $\sigma^*_{\geq j}$. We intend to recover τ_j so that we can "remove" ψ_j from $\sigma^*_{\geq j}$ and "add" τ_j to obtain $\sigma^*_{\geq j+1}$. Since ψ_j is a branch of depth at most t, using only $\log t$ bits of external information per variable we can point out all the variables where ψ_j and τ_j differ.

Thus using in total at most $s \log t$ bits we seem to be able to reconstruct σ from σ^*. If we could further argue that the space of restrictions Σ^* that F maps to is a factor $(n/n')^{-s}$ smaller than $\Sigma(n, n')$, then we could bound the failure probability by

$$\left(\frac{t}{n/n'}\right)^s. \tag{8}$$

This completes the high-level proof outline.

It is not immediately clear how to implement the above proof outline for the Tseitin contradictions $\mathrm{Tseitin}(G_n)$ defined over the $n \times n$ grid. One of the ideas of [8] is, given a full restriction σ, to try to reduce the number of centers in σ_0 with an even charge, where σ_0 is the assignment used in the construction of σ (see section 3). This approach does not work immediately: there are *more* assignments to the Tseitin formula with an even charge at all chosen centers except two (chosen freely) than there are assignments σ_0 with an even charge at all chosen centers.

We follow [8] and define *partial restrictions* ρ which have a large number of centers with an even charge. We can think of $\mathrm{Tseitin}(G_n)\restriction_\rho$ as an intermediate formula between $\mathrm{Tseitin}(G_n)$ and the formula $\mathrm{Tseitin}(G_n)\restriction_\sigma$ restricted by a full restriction σ. These partial restrictions have the property that the number of restrictions decreases as the number of centers with an even charge decreases. This allows us to implement the above proof outline.

Organization. In subsection 5.2 we define the concept of an associated center which allows us to recover a center from a discovered variable on a branch ψ_j. In subsection 5.3 we then introduce the notion of a *partial restriction.* Finally, in subsection 5.4, we introduce the main bookkeeping objects of the proof: *information pieces.*

5.2. Associated centers. Recall that each path connecting two centers in adjacent sub-squares consists of 5 nonempty *segments*: The first and the last segment are within the central area and of length 1, the middle segment is contained in between the central areas on the designated row or column, and segments 2 and 4 pass from the central areas to the area in between. Illustrations can be found in Figures 2 and 4. The key property of these paths is stated in the following lemma.

LEMMA 5.1. *If an edge e lies on multiple paths as described, then all these paths have a common endpoint.*

Proof. Consider the set of paths $\mathcal{P}$ connecting the centers of a fixed sub-square s to centers in adjacent sub-squares. Denote by $P \in \mathcal{P}$ a path that connects a center $c \in s$ to a center c' in the sub-square below or to the right of s. Recall that in between two consecutive centers there are at least two rows/columns with no centers.

Consider the 5 segments of P. If some path $P' \in \mathcal{P}$ shares an edge with P on segments 1, 2 or 3, then P and P' share c as a common endpoint. Similarly, if P' shares an edge on segments 4 or 5 with P, then they have c' as a common endpoint. The statement follows by symmetry. ∎

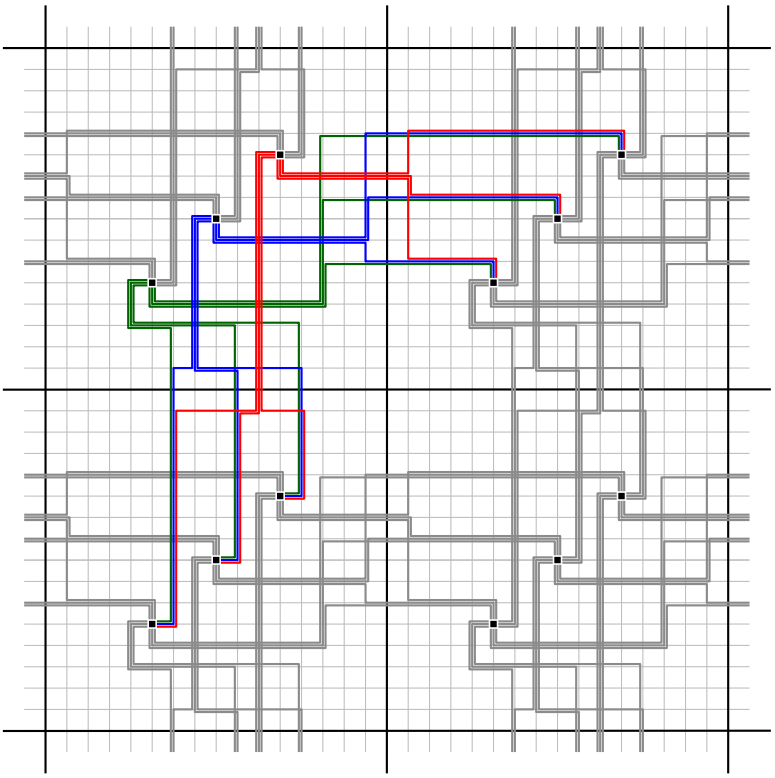

FIG. 4. *The paths from the top-left sub-square going right and down are highlighted.*

We let the *associated center* of an edge e denote the endpoint common to all paths that contain e. This notion is naturally extended to variables: the *associated center* of a variable x_e is the associated center of the edge e.

Let us introduce some notation. Given a fixed center v we say that a path P connecting v to some other center u goes to the δ, where δ is one of the directions *left, right, up*, or *down* if v lies in the sub-square to the δ of u. Let us stress that this notion of direction is relative to the fixed center v: if we fix u and then consider the same path P, then P has the opposite direction of δ. Finally note that if we fix the associated center of an edge e, then all the paths passing through e have the same direction.

Recall from the proof outline provided in subsection 5.1 that during the recovery of the assignment τ_j it is possible to cheaply identify variables x_e on the short branch ψ_j. The notion of an associated center allows us to pass from such a variable x_e to a center. This enables us to do a counting argument in terms of centers. A more detailed proof outline is provided in subsection 6.1.

5.3. Partial restrictions and pairings. Recall that we sample a full restriction $\sigma \sim \mathcal{D}_k\big(\Sigma(n,n')\big)$ by first sampling k *alive* centers such that each sub-square contains $(1 \pm 0.01)k/n'^2$ alive centers and then fixing the alive centers with the lowest numbered row in each sub-square to be the *chosen* centers C_σ of σ. We then sample an assignment σ_0 from the space of solutions to the Tseitin formula with 0-charges at chosen centers and 1-charges at all other nodes. We define σ from σ_0 as done in (2).

In the following we define a so-called *partial restriction* ρ. A partial restriction is in some sense one half of a full restriction σ: we split a full restriction σ into a partial restriction ρ and a *pairing* π. The partial restriction ρ is an (affine) restriction similar to σ, but it maps to more variables: while σ maps to variables y_P for P a chosen path, the partial restriction ρ maps to variables z_P where P is any path connecting two *alive* centers. The pairing π is then the (affine) restriction such that $\mathrm{Tseitin}(G_n)\restriction_\rho\restriction_\pi = \mathrm{Tseitin}(G_n)\restriction_\sigma$. Let us define ρ and π more formally by describing an alternative way of sampling a full restriction $\sigma \sim \mathcal{D}_k\big(\Sigma(n,n')\big)$.

As before we start by sampling uniformly at random k *alive centers* from the set of subsets of centers of size k satisfying that each sub-square contains $(1\pm 0.01)k/n'^2$ alive centers. Sample ρ_0 from the space of solutions to the Tseitin formula with 0-charges at alive centers and 1-charges at all other nodes. Since we have an odd number k of alive centers, by Lemma 2.1, this is indeed possible. We have variables z_P for each path P connecting two alive centers (an *alive path*), denote by $\oplus$ the exclusive or, and define ρ from ρ_0 by

$$\rho(x_e) = \begin{cases} \rho_0(x_e) & \text{if } e \text{ is not on an alive path,} \\ \bigoplus_{P\ni e} z_P & \text{if } e \text{ is on some alive path(s) and } \rho_0(x_e)=1, \\ \neg\bigoplus_{P\ni e} z_P & \text{if } e \text{ is on some alive path(s) and } \rho_0(x_e)=0. \end{cases} \tag{9}$$

The assignments given by ρ_0 to variables not on alive paths are called the *final values.*

It remains to define the pairing π such that ρ combined with π gives a full restriction σ. The alive center with the lowest numbered row in each sub-square is called the *chosen center* (as before), and the other alive centers are called *nonchosen centers.* Similarly we denote paths that are alive but not chosen as the *nonchosen paths* and let centers that are not alive simply be the *dead centers.*

The main task of the pairing π is to ensure that the nonchosen centers have an odd charge in the final full restriction. We can thus think of π as an assignment to the nonchosen paths such that each nonchosen center has an odd number of incident nonchosen paths that are set to 1. For reasons to become apparent later it is convenient to have small 1-components in π. Let a star of size 4 be the graph with a central node of degree 3 and three nodes of degree 1 connected to the central node by an edge.

Definition 5.2 (graphical pairing). *A* graphical pairing π_0 *is a graph supported on the nonchosen centers. Each component of* π_0 *is either a single edge or a star of size* 4. *The centers of a component are in distinct but adjacent sub-squares.*

The following lemma follows from the proof of [8, Lemma 4.3]. For completeness we provide a proof in Appendix A.

Lemma 5.3 (see [8, Lemma 4.3]). *For large enough integer* $a \in \mathbb{N}$ *the following holds. If each sub-square has* $(1\pm 0.01)a$ *alive centers, then there is a graphical pairing* π_0.

For each choice of alive centers we fix a graphical pairing π_0. Let us stress that there is *no* randomness in the choice of π_0 once we have sampled a partial restriction ρ.

We have variables y_P for each chosen path P connecting two chosen centers. We define the *pairing* π from a graphical pairing π_0 by

$$\pi(z_P) = \begin{cases} 1 & \text{if } P \text{ is nonchosen and in } \pi_0 \text{ as an edge,} \\ 0 & \text{if } P \text{ is nonchosen and not in } \pi_0, \\ y_P & \text{if } P \text{ is a chosen path.} \end{cases} \tag{10}$$

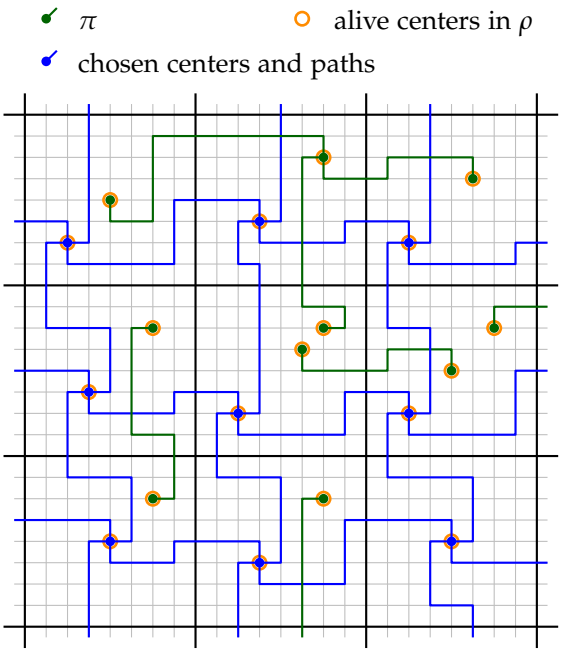

FIG. 5. *A part of the grid with π and the chosen centers and paths highlighted.*

See Figure 5 for an illustration. With π and ρ defined we let $\sigma(x_e) = \pi\big(\rho(x_e)\big)$.

We claim that sampling σ through ρ and π as outlined above is equivalent to sampling it directly as explained in subsection 3.1. This is readily verified: we may assume that we sample the same set of alive centers. Since π, which only depends on the set of alive centers, defines a bijection between assignments ρ_0 and σ_0 as sampled in the respective processes, we see that the two distributions obtained from these processes are indeed equivalent.

The sampling of ρ is the main probabilistic event that we analyze in the proof of the switching lemma. Since σ can be defined in terms of ρ and π, we write $\sigma = \sigma(\rho, \pi)$ whenever we want to stress this fact.

Instead of sampling the k alive centers uniformly from all subsets of centers of size k satisfying that each sub-square has $(1 \pm 0.01)k/n'^2$ alive centers, we can instead simply sample k centers uniformly from the set of centers and then sample ρ as above. Denote by $R(k, n, n')$ the space of partial restrictions obtained by choosing k alive centers uniformly from the set of centers, and write $R^{\mathrm{reg}}(k, n, n')$ for the space of partial restrictions with k alive centers such that each of the $n' \times n'$ sub-squares contains $(1 \pm 0.01)k/n'^2$ many alive centers.

LEMMA 5.4. *For $n, n' \in \mathbb{N}$ large enough there is a constant C such that for odd $k \geq Cn'^2 \log n'$ it holds that $R^{\mathrm{reg}}(k, n, n') \geq (1 - 1/n')R(k, n, n')$.*

Proof. Sample ρ uniformly from $\mathcal{U}\big(R(k, n, n')\big)$, and consider the random variables $X_{(i,j)}$ defined for each sub-square (i, j) counting the number of alive centers of ρ in the sub-square (i, j). Each such $X_{(i,j)}$ is the sum of the indicator random variables $Y^{\nu}_{(i,j)}$ for $\nu \in [\Delta]$ that are 1 if and only if the νth center of the sub-square (i, j) is alive in ρ. Note that these random variables are negatively correlated since we choose exactly k alive centers overall. Since the Chernoff bounds continue to hold for

negatively correlated random variables, it holds that $X_{(i,j)} \in (1 \pm 0.01)k/n'^2$ except with probability $1/n'^3$ for a large enough constant $C > 0$. The claim follows by a union bound over all sub-squares. ∎

LEMMA 5.5. *Let $n, n' \in \mathbb{N}^+$ be large enough such that $n \geq 20n'C \log n'$, and suppose that $k = Cn'^2 \log n'$. For integer $s \geq 1$ it holds that*

$$\frac{|R(k-2s,n,n')|}{|R^{\mathrm{reg}}(k,n,n')|} \leq \left(\frac{13C\log n'}{n/n'}\right)^{2s}.$$

The whole proof of the switching lemma hinges on this exponential in s factor $(\frac{13C\log n'}{n/n'})^{2s}$: it allows us to implement the proof plan as outlined in subsection 5.1 with $\Sigma = R^{\mathrm{reg}}(k,n,n')$ and $\Sigma^* = \bigcup_{i=\Omega(s)}^{(k-1)/2} R(k-2i,n,n')$.

Proof. By Lemma 5.4 it holds that $|R^{\mathrm{reg}}(k,n,n')| \geq (1-1/n)|R(k,n,n')|$. It thus suffices to show that $|R(k-2s,n,n')| \leq (\frac{12C\log n'}{n/n'})^{2s} \cdot |R(k,n,n')|$ to establish the claim. Recall from Lemma 2.1 that the number of solutions of the Tseitin formula $\mathrm{Tseitin}(G_n, \alpha)$ only depends on the parity of the sum of the charges $\sum_{v \in V(G)} \alpha_v$ and n: the space of restrictions $R(k-2s,n,n')$ is of size $2^{r_n}\binom{m}{k-2s}$, where $m = n'^2\Delta$ is the number of centers and r_n is as in Lemma 2.1. It thus holds that

$$\frac{|R(k-2s,n,n')|}{|R(k,n,n')|} = \frac{2^{r_n}\binom{m}{k-2s}}{2^{r_n}\binom{m}{k}} = \prod_{i=0}^{2s-1} \frac{k-i}{m-k+i} \leq \left(\frac{k}{m-k}\right)^{2s} \tag{11}$$

$$= \left(\frac{C\log n'}{\Delta - C\log n'}\right)^{2s} \tag{12}$$

$$\leq \left(\frac{2C\log n'}{\Delta}\right)^{2s}, \tag{13}$$

using the assumption that $n \geq 20n'C\log n'$ and hence $\Delta \geq 2C\log n'$. The claimed bound follows from the fact that $\Delta \geq n/6n'$. ∎

Let us establish some nomenclature. As the original grid is also a graph with edges, from now on we use the term *grid-edges* to refer to edges in the original grid. We only consider paths in the original grid and keep the short term *path* for these. An *edge* refers to an alive path, that is, an *edge* is a connection between two alive centers and corresponds to an alive path (possibly a chosen path) in the original grid. Edges between chosen centers are *new grid-edges*, and we say that two chosen centers are neighbors if they lie in adjacent sub-squares.

A partial restriction is usually denoted by ρ, and since we mostly discuss partial restrictions we simply call them *restrictions*, while we use the term *full restrictions* when that is what we have in mind.

5.4. Information pieces. As an intermediate between ρ and a full restriction σ we have ρ along with some information in the form of the existence or absence of edges (alive paths) incident to alive centers. We have the following definition.

DEFINITION 5.6 (information piece). *An* information piece *is either*

1. *an edge $\{v, w\}$ where v, w are centers in adjacent sub-squares, or*
2. *of the form $(v, \delta, \perp)$ for a center v and a direction δ, that is, δ is either* left, right, up, *or* down.

The former says that there is an edge from v to w, while the latter says that there is no edge from v in direction δ.

We can think of information pieces as always being on alive centers. Suppose we are given a restriction ρ together with an information piece $\{v, w\}$ for alive centers v and w. The tuple $(\rho, \{v, w\})$ corresponds to the restriction obtained from ρ by additionally restricting

1. the variable $z_{P_{vw}} = 1$, where P_{vw} is the alive path connecting the centers v and w, and
2. setting variables $z_{P'} = 0$ for any alive path $P' \neq P_{vw}$ which lies in the same direction as P_{vw} does from either v or w.

Let us stress that the second point above is by choice: it would be perfectly fine to allow multiple paths set to 1 in one direction incident to a single center. In the following we do not consider such assignments.

Note that there is some asymmetry in information pieces: if we are instead given ρ along with $(v, \text{down}, \perp)$, then this corresponds to the restriction obtained from ρ where we additionally restrict all variables $z_P = 0$, where P is an alive path going down from v.

We make the intuition of what restriction one obtains from ρ with some information pieces more formal in Definition 5.9. It is convenient to have the abstraction of an information piece since it allows us to think of certain alive paths as restricted while only knowing one endpoint. We also use sets of information pieces.

DEFINITION 5.7 (information set). *An* information set *I is a collection of information pieces. The* support *of I, denoted by* $\text{supp}(I)$*, is the set of centers mentioned in these pieces.*

A partial assignment to new grid-edges naturally corresponds to a set of information pieces: an assignment of 0 to a new grid-edge P_{uv} corresponds to two nonedge information pieces in the appropriate directions at the chosen centers u and v. An assignment of 1 corresponds to an information piece in the form of an edge between the two chosen centers u, v connected by the path P_{uv}.

DEFINITION 5.8 (local consistency for information sets). *An information set I is* locally consistent *if*

1. *the set I does not have two different information pieces in one direction from the same center, and*
2. *if I has information in all four directions from a center v, then it has an odd number of edges incident to v.*

Two information sets I and J are pairwise locally consistent *if the union $I \cup J$ is locally consistent.*

We use the term *locally consistent* both for sets of information pieces and partial assignments. Local consistency for assignments requires an odd number of 1's incident to any node if it assigns all the incident variables. This corresponds exactly to the local consistency property of information pieces. Hence a locally consistent assignment gives rise to a locally consistent information set. Note, though, that the converse is not true since a locally consistent assignment needs to satisfy further properties (see Definition 2.2).

Jointly with ρ an information set forces the values of some more variables as follows.

DEFINITION 5.9 (forcing). *Let ρ be a restriction, denote by ρ_0 the assignment used in the construction of ρ, and let I be an information set. A variable x_e is* forced *by (ρ, I) if and only if either*

1. *the associated center v of x_e is dead in ρ, or*
2. *the information set I has information pieces in all directions incident to v.*

In the former case the variable x_e is always forced to $\rho_0(x_e)$, while in the later case it is forced to $\neg\rho_0(x_e)$ if the information piece $i \in I$ incident to v in the direction of x_e is

1. *an edge $i = \{v, w\}$, and*
2. *the grid-edge e lies on the alive path P connecting v to w.*

Otherwise, as in the first case, the variable x_e is forced to $\rho_0(x_e)$.

Remark 5.10. Readers familiar with [8, 9] should note that our notion of forcing differs from the notion used in previous work: a variable with an alive center is considered forced only if the information set I contains information pieces in *all* directions incident to the associated center. In [8, 9] a variable x_e is considered forced if I contains an information piece incident to the associated center in the direction of x_e.

There are other situations where the value of a variable might be determined by ρ and I such as the lack, or scarcity, of alive centers in a sub-square. We do not use such information in the reasoning below. We need a notion of a closed information set.

DEFINITION 5.11 (closed information set). *An information set I is* closed at a node u *if I is locally consistent and has information pieces in all four directions incident to u. The information set I is* closed *if I is closed at all nodes $u \in \operatorname{supp}(I)$.*

The definition implies that if an information set I is closed, then for any $u \in \operatorname{supp}(I)$ and direction δ where there is not an element of $\operatorname{supp}(I)$ we have a nonedge $(u, \delta, \perp)$. When considered as a graph such an information set is an odd-degree graph (with degrees one and three) on the centers of $\operatorname{supp}(I)$. See Figure 6 for an illustration of a closed information set. Going forward we will usually think of each connected component of the pairing π as a closed information set.

Consider an information set I supported on alive centers that is closed on U, and suppose that on centers $v \in \operatorname{supp}(I) \setminus U$ the set I has an even number of incident edges. Note that the variables forced by (ρ, I) can be described by a restriction with the centers U killed: negate the values of any variable on any path in I, and then forget that the centers in U were alive. More precisely, if ρ_0 denotes the assignment used in the construction of ρ, then the restriction ρ^* defined by

$$\rho^*(x_e) = \begin{cases} b & \text{if } (\rho, I) \text{ forces } x_e \text{ to } b \in \{0,1\}, \text{ and} \\ \rho(x_e) & \text{otherwise} \end{cases} \tag{14}$$

assigns the same variables to the same constants as (ρ, I) forces. Thus, if we let such an information set I operate on a restriction ρ, we obtain a restriction with fewer alive centers where the killed centers are precisely the centers in U.

6. Proof of the switching lemma. This section is dedicated to the proof of the switching lemma restated for convenience.

LEMMA 4.2 (switching lemma). *There are absolute constants $A, C, n_0 > 0$ such that for integer $n \geq n_0$ the following holds. Let $k, m, n', s, t \in \mathbb{N}^+$ satisfy $n/n' \geq At \log^4 n$, $k = n'^2(1 \pm 0.01)C \log n'$ be odd, and $t \leq s \leq n'/32$. Then for any decision trees $T_1, \ldots, T_m$ of depth at most t querying edges of the $n \times n$ grid it holds that if $\sigma \sim \mathcal{D}_k\big(\Sigma(n, n')\big)$, then the probability that $\bigvee_{i=1}^m T_i\lceil_\sigma$ cannot be represented by a decision tree of depth s is bounded by*

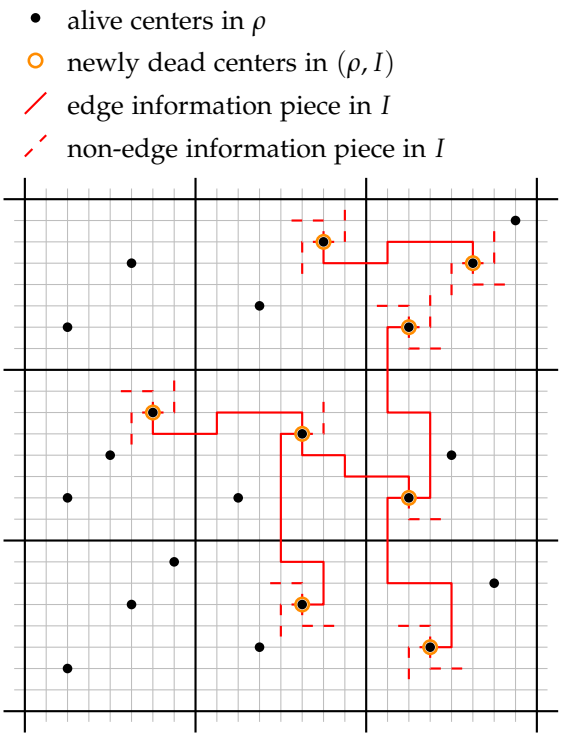

FIG. 6. *A closed information set I and the newly dead centers in* $\mathrm{supp}(I)$.

$$\left(\frac{At\log^4 n'}{n/n'}\right)^{s/64}.$$

The proof of Lemma 4.2 closely follows the argument of [8] with some simplifications and changes. In the following section we give a detailed proof outline. The details of the proof are fleshed out in subsections 6.2–6.5.

6.1. Detailed proof outline. Let us recall the setup. We have a full restriction $\sigma = \sigma(\rho, \pi)$ as defined in section 3 that consists of a restriction ρ and a pairing π as introduced in subsection 5.3. The restriction ρ has $(1 \pm 0.01)C\log(n)$ alive centers in each sub-square for some (large) constant $C > 0$. For $i \in [m]$ we have decision trees T_i of depth at most $\mathrm{depth}(T_i) \le t$ querying grid-edges. We want to bound the probability that there is no decision tree of depth $s \ge t$ representing $\bigvee_{i=1}^m T_i\lceil_\sigma$.

In the following we construct a decision tree $\mathcal{T}$ that represents $\bigvee_{i=1}^m T_i\lceil_\sigma$ which is with high probability over the choice of ρ of depth at most s. Let us stress that in contrast to the decision trees T_i that query grid-edges, this decision tree $\mathcal{T}$ queries edges, that is, path variables.

The decision tree $\mathcal{T}$ is constructed in a similar manner to the construction of a canonical decision tree: we proceed in stages where in each stage a branch τ of $\mathcal{T}$ is extended by querying variables related to the first 1-branch ψ in the restricted trees $T_1\lceil_{\sigma\tau}, T_2\lceil_{\sigma\tau}, \ldots, T_m\lceil_{\sigma\tau}$. For now it is not so important what the "related variables of ψ" precisely are and we can simply think of these as the variables on the branch ψ. Once all these variables have been queried we check in each new leaf of the tree whether we traversed the path ψ. If so, then we label the leaf with a 1 and otherwise

continue with the next stage. If there are no 1-branches left, then we label the leaf with a 0.

As argued in subsection 5.1 it is quite immediate that the above process indeed results in a decision tree $\mathcal{T}$ that represents $\bigvee_{i=1}^{m} T_i\lceil_\sigma$. It remains to argue that $\mathcal{T}$ is with high probability of depth at most s.

We analyze this event using the labeling technique of Razborov [16]. The idea of this technique is to come up with an (almost) bijection from restrictions ρ that give rise to a decision tree $\mathcal{T}$ of depth larger than s to a set of restrictions that is much smaller than the set of all restrictions. In a bit more detail, given such a bad ρ, we create a restriction ρ^* with fewer alive centers such that with a bit of extra information we can recover the restriction ρ from ρ^*. As ρ^* has roughly s fewer alive centers than ρ, we obtain our statement by Lemma 5.5.

Let us explain in a bit more detail how to construct ρ^* from a ρ that gives rise to a decision tree $\mathcal{T}$ of depth $\mathrm{depth}(\mathcal{T}) \geq s$. To this end we first need to slightly refine the construction process of $\mathcal{T}$. Namely, we need to discuss what the "related variables of a branch ψ" are. Instead of thinking of this as a set of variables we rather want to think of it as an information set J, as introduced in subsection 5.4. The information set J is a minimal set that forces, along with the already collected information set on the branch τ, the branch ψ. Once we identified such a set J, we then query all necessary variables to see whether we agree with J (along with some further variables).

Recall that we are trying to explain how to construct a restriction ρ^* from a restriction ρ that gives rise to a decision tree $\mathcal{T}$ of large depth. Fix a long branch $\tau \in \mathcal{T}$, and consider the sets $J_1, J_2, \ldots, J_g$ identified in the different stages of the construction of the long branch τ. For this proof overview, let us assume that each J_j is closed and that the support of these information sets are pairwise disjoint. Let us stress that this is a slight simplification. Assuming this holds, note that the union $J^* = \cup_{i=1}^{g} J_j$ is also closed, and recall from subsection 5.4 that all variables forced by (ρ, J^*) can be described by a restriction where the centers in $\mathrm{supp}(J^*)$ are killed. This defines the restriction ρ^*: it is the restriction that forces all variables forced by (ρ, J^*). Assuming that the support of J^* is large we see that ρ^* has much fewer alive centers. Using Lemma 5.5 we obtain an upper bound on the failure probability, assuming that we can easily recover ρ from ρ^*.

It remains to argue that ρ can be recovered from ρ^* with little extra information. The idea is to remove the set J_j, starting with $j = 1$, one-by-one from ρ^*. To do this cheaply we use the decision trees $T_1, \ldots, T_m$. Recall that the information set J_1 determines all variables on the first 1-branch ψ_1. This implies in particular that ρ^* traverses the branch ψ_1. Hence identifying ψ_1 is for free: it is the first 1-branch in $T_1, \ldots, T_m$ traversed by ρ^* (since the set J_1 is pairwise disjoint from all later sets J_j). Once we identified the branch ψ_1 we want to recover the first part of the long branch τ so that we can repeat this argument with J_2. As ψ_1 is of length at most t, using only $\log t$ bits per variable, we indicate which variables are different on τ from J_1. This lets us cheaply recover τ along with the centers killed by J_1. Repeating this argument g times allows us to recover the restriction ρ.

This completes the proof overview. We allowed ourselves some simplifications and left out a number of details.

Organization. The proof of Lemma 4.2 spans the following four subsections. In subsection 6.2 we define the extended canonical decision tree $\mathcal{T}$, and in the subsequent subsection 6.3 we prove some crucial properties of these decision trees. In subsection 6.4 we explain how Lemma 4.2 follows from the encoding argument as outlined above.

The proof of the encoding lemma is the final part of the proof of the switching lemma and is given in subsection 6.5.

6.2. Extended canonical decision trees. Sample a full restriction $\sigma = \sigma(\rho, \pi) \sim \mathcal{D}_k\big(\Sigma(n, n')\big)$ as defined in section 3, and denote by C_σ the chosen centers of σ. Recall that ρ has $(1 \pm 0.01)C \log n$ many alive centers in each sub-square. Let $T_1, \ldots, T_m$ be decision trees of depth at most t querying grid-edge variables of the $n \times n$ grid.

We intend to construct the *extended canonical decision tree* $\mathcal{T}$ that represents $\bigvee_{i=1}^m T_i\lceil_\sigma$. Note that, in contrast to the decision trees T_i that query variables of the unrestricted formula, that is, they query grid-edges, the decision tree $\mathcal{T}$ queries variables of the *restricted* formula, that is, path variables y_P where P is a chosen path (a path connecting two chosen centers in adjacent sub-squares).

Intuition. Before formally defining the extended canonical decision tree let us give an intuitive (and flawed) outline of the construction. We would like to first construct a decision tree $\widetilde{\mathcal{T}}$ representing $\bigvee_{i=1}^m T_i$ restricted by ρ (instead of σ) in the usual manner: proceed in stages, and in each stage extend a branch τ of $\widetilde{\mathcal{T}}$ by querying variables related to the first 1-branch ψ in the trees $T_1\lceil_{\rho\tau}, T_2\lceil_{\rho\tau}, \ldots, T_m\lceil_{\rho\tau}$. For the moment, as in subsection 6.1, we may think of the related variables as the variables on the branch ψ. Once we have queried all these variables we check in each newly created leaf of $\widetilde{\mathcal{T}}$ whether we traversed ψ: if so, then the leaf becomes a 1-leaf. Otherwise we proceed with the next stage. If no further 1-branches are left, then the leaf is labeled 0.

Note that such a decision tree $\widetilde{\mathcal{T}}$ not only queries variables y_P for chosen paths P but queries variables associated with *any* path connecting two alive centers: while $\mathcal{T}$ knows all of σ and hence only needs to query variables associated with chosen paths, the tree $\widetilde{\mathcal{T}}$ is unaware[2] of the pairing π and hence does not know the identity of the chosen centers. Once we have $\widetilde{\mathcal{T}}$ we would like to define $\mathcal{T}$ (the decision tree representing $\bigvee_{i=1}^m T_i\lceil_\sigma$) as the restriction of $\widetilde{\mathcal{T}}$ by π, that is, $\mathcal{T} = \widetilde{\mathcal{T}}\lceil_\pi$.

Instead of analyzing the event that $\mathcal{T}$ has a branch of length s, we would like to analyze the event that $\widetilde{\mathcal{T}}$ has a branch $\tilde{\tau}$ of length s which is pairwise locally consistent with π, that is, the restricted branch $\tilde{\tau}\lceil_\pi$ is a traversable branch in $\mathcal{T}$. If we manage to upper bound the probability of the latter event, then this upper bound clearly also applies to the former.

The actual construction differs from the above intuitive description. We do *not* take the detour via the decision tree $\widetilde{\mathcal{T}}$. Instead we directly define the decision tree $\mathcal{T}$ since we want to treat chosen path variables in a slightly different manner than a path variable whose corresponding path simply connects alive centers. The formal construction follows.

Setup. We need to introduce two sets that guide the construction of the extended canonical decision tree $\mathcal{T}$. We start with $\mathcal{T}$ being the empty decision tree. We extend the decision tree $\mathcal{T}$ in *stages.* In each stage we extend $\mathcal{T}$ at some branch. By the end of a stage each branch τ of $\mathcal{T}$ is associated with

1. a subset of the alive centers $S = S(\tau, \sigma)$ called the *exposed centers,* and
2. an information set $I = I(\tau, \sigma)$ supported on alive centers.

Initially the sets $S(\emptyset, \sigma)$ and $I(\emptyset, \sigma)$ are empty. Throughout the construction of $\mathcal{T}$ we maintain the following invariants.

[2]Let us remark that the pairing π is a function of the restriction ρ. Hence given ρ the pairing π is known and hence this description makes little sense formally. For intuition, however, it is a quite insightful view.

INVARIANT 6.1. *Throughout the creation of $\mathcal{T}$ the following properties of S and I are maintained.*

1. *No element is ever removed from S or I. In other words, the sets S and I only become larger throughout the creation of a branch τ of $\mathcal{T}$.*
2. *The information set I is locally consistent and closed on S.*
3. *If a center of a connected component of the pairing π is in the set of exposed centers S, then the entire component is in S.*
4. *The part of the information set I on the nonchosen centers $S \setminus C_\sigma$ is a subset of the connected components of the pairing π in the form of a closed information set.*
5. *For every exposed chosen center $v \in S \cap C_\sigma$ all the variables y_P incident to v are queried by the branch τ. The answers to these queries is precisely the information in I: 1-answers are recorded in the form of an edge, while the 0-answers are recorded as a nonedge in the appropriate direction.*

Let us stress that information about the pairing π comes from the restriction $\sigma(\rho, \pi)$ and hence, in order to maintain Invariant 6.1, property 4, we do not need to query a variable in $\mathcal{T}$. This is in contrast to Invariant 6.1, property 5: querying a variable y_P associated with a chosen path P causes a query in the decision tree $\mathcal{T}$. There is another subtle difference between property 4 and property 5: on the nonchosen centers we only have information pieces in the information set I that are incident to the exposed centers S. On the other hand I may contain information pieces incident to chosen centers that are not exposed, that is, information pieces incident to chosen centers that are not in $S \cap C_\sigma$. Finally note that the information set I never contains a path between a chosen center and a nonchosen center.

We need one last definition before we can formally define the extended canonical decision tree $\mathcal{T}$. Recall that the construction of the decision tree $\mathcal{T}$ commences in stages. In each stage we extend a branch τ of $\mathcal{T}$ by querying variables associated with a 1-branch ψ of a decision tree T_i pairwise locally consistent with the branch τ and the full restriction σ. Hence there is a unique minimum partial assignment to the variables of ψ pairwise locally consistent with τ and σ to reach the corresponding leaf. In the following we define the analogue of this minimum partial assignment in terms of information pieces.

Intuitively a *possible forcing information* J is a minimal information set that jointly with the information set $I(\tau, \sigma)$ and the partial restriction ρ forces[3] all variables on the branch ψ to take the values given by this partial assignment. We require some further properties as summarized in the following definition.

DEFINITION 6.2 (possible forcing information). *Let ψ be a branch, let I be an information set, let S be a set of centers, and denote by $\sigma(\rho, \pi)$ a full restriction. A* possible forcing information *for ψ is a minimal information set J that satisfies the following.*

1. *The information sets I and J are pairwise locally consistent.*
2. *If an associated center u of a variable on the branch ψ is not in S, then J is closed at u.*
3. *All variables on ψ are forced by $(\rho, I \cup J)$ such that the leaf of ψ is reached.*
4. *The part of the information set J on the nonchosen centers is a subset of the connected components of the pairing π in the form of a closed information set.*

[3]According to Definition 5.9 a variable is forced if and only if the associated center is dead or we have information pieces in all directions at its associated center.

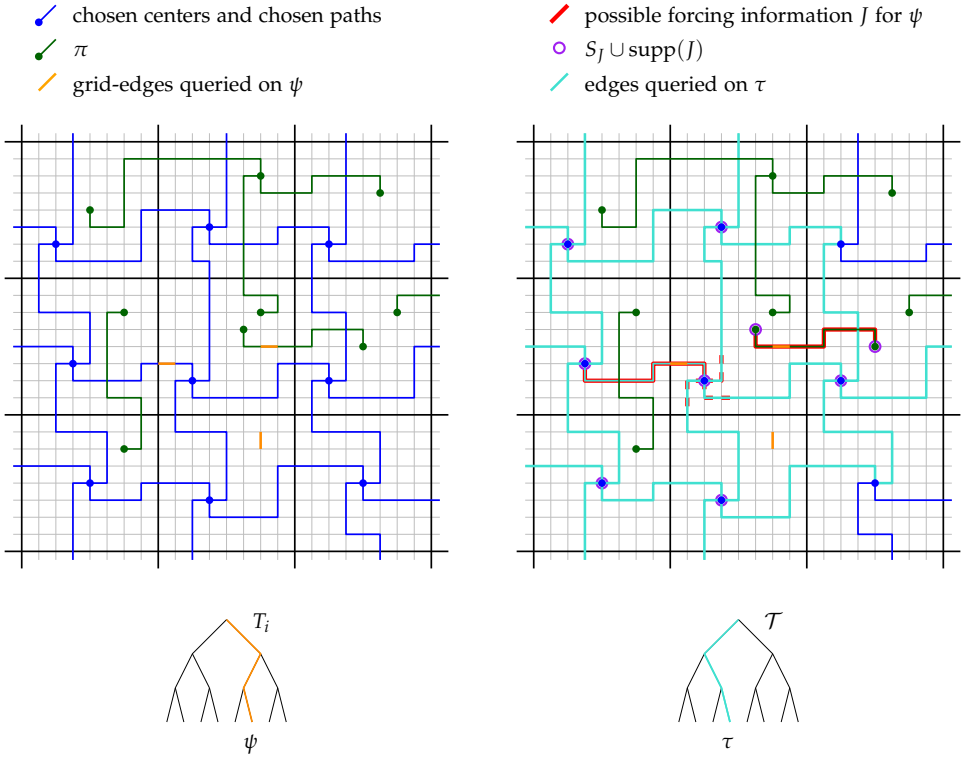

FIG. 7. *Depiction of the first stage in the construction of the extended canonical decision tree $\mathcal{T}$. For clarity we omitted the nonedges in J on nonchosen centers.*

Let us stress that a possible forcing information J never contains an edge between a chosen center and a nonchosen center. Note that a possible forcing information J may not be unique for a given branch ψ. If there are several sets as described above, choose one in a fixed but otherwise arbitrary manner. While the choice is not essential, we do need to establish that whenever some decision tree T_i can still reach a 1-leaf, then there is a possible forcing information J. We postpone this to the following section (see Lemma 6.3) and for now assume that such an information set J exists whenever we have a branch ψ as described. With these definitions in place we are ready to formally define the extended canonical decision tree $\mathcal{T}$.

Construction. We provide pseudocode (Algorithm B.1) in case of ambiguity in the following verbal description as well as an illustration of the process in Figure 7. Initially $\mathcal{T} = \emptyset$ is the empty tree. In each stage we fix a branch τ in $\mathcal{T}$ and go over the decision trees $T_1, T_2, \ldots, T_m$ one by one. Suppose we consider the decision tree T_i. Let ψ be the first (in some arbitrary but fixed order) 1-branch of T_i such that

1. the branch ψ and the full restriction σ are pairwise locally consistent, and
2. the assignment[4] ψ_σ induced by ψ on the smaller $n' \times n'$ grid is locally consistent with τ.

If there is no such branch, then proceed with the decision tree T_{i+1}. If there is no such branch ψ for any decision tree T_i, then label the leaf τ of $\mathcal{T}$ by 0 and continue with a different branch τ' of $\mathcal{T}$ until all leaves of $\mathcal{T}$ are labeled. For the remainder

[4]As defined in Property 3 of Definition 3.1.

of this section let us assume that there is such a 1-branch ψ. Denote by J a possible forcing information for the *forceable branch* ψ.

Let I_π be the information set from π incident to nonchosen centers in $\mathrm{supp}(J)$, and denote by S_J all chosen centers at distance at most 1 from $\mathrm{supp}(J) \cap C_\sigma$ with respect to the smaller $n' \times n'$ grid.

Extend the decision tree $\mathcal{T}$ at the leaf τ by querying all variables incident to S_J. Let τ' be a newly created extension of τ. Denote by $I_{\tau' \setminus \tau}$ the information set consisting of the answers to the queries on $\tau' \setminus \tau$ and the locally implied answers; that is, 1-answers are recorded as an edge and 0-answers as a nonedge incident to the exposed center.

Update the bookkeeping objects

1. $S(\tau', \sigma) = S(\tau, \sigma) \cup S_J \cup \mathrm{supp}(J)$ and
2. $I(\tau', \sigma) = I(\tau, \sigma) \cup I_{\tau' \setminus \tau} \cup I_\pi$.

Finally check whether the information set $I(\tau', \sigma)$ traverses the forceable branch ψ of T_i. Since all variables on ψ have their associated center in S and I is closed on S, this can be easily checked. If the forceable branch ψ is indeed followed, then label the leaf τ' with a 1. Otherwise, that is, if the forceable branch is not followed, then proceed with the next stage.

This completes the description of the creation of the extended canonical decision tree $\mathcal{T}$ for $\bigvee_{i=1}^m T_i\lceil_\sigma$. If in the construction of $\mathcal{T}$ a branch τ is of length at least s, then we stop the construction of $\mathcal{T}$. In the following we bound the probability of ever reaching a stage such that the construction is stopped.

It is straightforward to check that the invariants hold after every completed stage; we only need to recall that we enforce that all the newly created branches τ' are *locally consistent* as assignments on the smaller grid (see Definition 3.1 and the discussion thereafter). Hence the information sets created are locally consistent, that is, all the information sets $I(\tau', \sigma)$ are locally consistent. The other invariants hold by construction.

6.3. Some properties of extended canonical decision trees. We need to show that the extended canonical decision tree $\mathcal{T}$ represents $\bigvee_{i=1}^m T_i\lceil_\sigma$. This will be a direct consequence of the postponed claim that if it is possible to reach a 1-leaf in some decision tree T_i, then there is a possible forcing information J.

Observe that at any point when forming the extended canonical decision tree the information I comes partly from the pairing π and from queries already done in the decision tree $\mathcal{T}$ with answers τ. Remember that $\sigma = \sigma(\rho, \pi)$ includes all the information from the pairing π.

LEMMA 6.3. *If there is a 1-branch ψ in $T_i\lceil_\sigma$ that is pairwise locally consistent with τ, then there is a possible forcing information J for ψ.*

Proof. Extend ψ (an assignment to the $n' \times n'$ grid) to an assignment $\psi^+ \supseteq \psi$ such that

1. the assignments ψ^+ and τ are pairwise locally consistent, and
2. the assignment ψ^+ is complete on all associated centers u of variables on ψ except on the associated centers u on which τ is already complete.

According to Lemma 2.3 such an extension ψ^+ exists, assuming that $|\mathrm{supp}(\psi)| \le n'/32$ and $|\mathrm{supp}(\tau)| \le n'/4$.

Let ψ_0 be the branch in T_i that gives rise to ψ. Note that ψ_0 is an assignment to the $n \times n$ grid. Let us construct a possible forcing information J such that ψ_0 is followed.

Add information pieces next to chosen centers as indicated by ψ^+. The information pieces next to nonchosen centers are the relevant information pieces from π.

We claim that the information set J that contains the just-mentioned information pieces is a valid possible forcing information for ψ_0 if we ignore the minimality condition. Let us check the other properties of Definition 6.2. Let I be the information set corresponding to τ.

Note that since τ and ψ^+ are locally consistent, so are I and J. Hence Property 1 follows and Properties 2–4 follow by construction. Hence any minimal subset of J with the above properties constitutes a forcing information for ψ_0. This completes the proof of the lemma. ∎

As an immediate corollary we have that the decision tree $\mathcal{T}$ is indeed a legitimate choice for $\bigvee_{i=1}^m T_i\lceil_\sigma$.

COROLLARY 6.4. *The extended canonical decision tree $\mathcal{T}$ represents $\bigvee_{i=1}^m T_i\lceil_\sigma$.*

We have three auxiliary lemmas regarding forcing information and the size of the set of exposed centers S.

LEMMA 6.5. *In each stage at most $4|\mathrm{supp}(J)|$ centers are added to the set of exposed centers S.*

Proof. We add $\mathrm{supp}(J) \cup S_J$ to the set of exposed centers S. Recall that S_J is the set of chosen centers at distance at most 1 from $\mathrm{supp}(J) \cap C_\sigma$. It suffices to argue that every chosen center in $\mathrm{supp}(J)$ is adjacent to at least one other chosen center in $\mathrm{supp}(J)$: if this holds, then for every chosen center in $\mathrm{supp}(J)$ we add at most three chosen centers to $S_J \setminus \mathrm{supp}(J)$.

By minimality of J each nonexposed chosen center $u \in \mathrm{supp}(J) \cap C_\sigma$ is either an associated center of a variable on ψ or is adjacent to such an associated center. In the latter case we are done. If u is an associated center, then the forcing information J contains information pieces in all directions incident to u and hence, since J is locally consistent, there is at least one edge incident to u in J. The claim follows. ∎

COROLLARY 6.6. *In each stage at most $16t$ centers are added to the set of exposed centers S.*

Proof. A forceable branch ψ is of length at most t as the decision trees T_i are of depth at most t. For each variable x_e on ψ there are at most four chosen centers in $\mathrm{supp}(J)$ if the associated center of x_e is chosen and at most four nonchosen centers if the associated center is nonchosen. Hence $|\mathrm{supp}(J)| \le 4t$. The statement follows by Lemma 6.5. ∎

LEMMA 6.7. *Consider a branch $\tau \in \mathcal{T}$ along with the possible forcing information $J_1, J_2, \ldots$ used in the different stages of the construction of τ. For $j \neq j'$ it holds that $\mathrm{supp}(J_j)$ and $\mathrm{supp}(J_{j'})$ are disjoint.*

Proof. Denote by $1, 2, \ldots$ the stages used in the construction of the branch τ, and let S^*_{j-1} be the set of exposed centers S at the beginning of stage j. Suppose $j' < j$.

By minimality of J_j it holds that $S^*_{j-1} \cap \mathrm{supp}(J_j)$ only contains chosen centers that are adjacent to a nonexposed chosen center. Since in stage j' all centers at distance at most 1 from $\mathrm{supp}(J_{j'}) \cap C_\sigma$ are added to the set of exposed centers, it holds that (1) $\mathrm{supp}(J_{j'}) \subseteq S^*_{j-1}$ and (2) all chosen centers adjacent to $\mathrm{supp}(J_{j'}) \cap C_\sigma$ are in S^*_{j-1}. Hence $\mathrm{supp}(J_j)$ cannot contain a center from $\mathrm{supp}(J_{j'})$. The claim follows. ∎

6.4. Encoding ρ. We want to bound the number of restrictions ρ that give rise to an extended canonical decision tree $\mathcal{T}$ of depth $\mathrm{depth}(\mathcal{T}) \geq s$. Fix such a restriction ρ along with the extended canonical decision tree $\mathcal{T}$ and a branch τ of length at least s. Denote by $1, 2, \ldots$ the stages in which the branch τ was constructed.

Since the branch τ is of length at least s there is a first stage g such that by the end of stage g at least $s/4$ centers are exposed: only variables incident to exposed chosen centers are queried, and each exposed center causes at most 4 queries on the branch τ. In other words, if we let $\tau_g \subseteq \tau$ be the branch constructed by the end of stage g, then the set of exposed centers $S_g^* = S(\tau_g, \sigma)$ is for the first time of size $|S_g^*| \geq s/4$. We analyze the event of ever reaching such a stage g.

Note that $|S_g^*| < s/4 + 16t$ by Corollary 6.6 and $g \leq s/4$ as in each stage at least one center is added to the set of exposed centers S. For $j \in [g]$ we let the forceable branch of stage j in the decision tree T_{i_j} be denoted by ψ_j, let J_j be the corresponding possible forcing information, and let $\tau_j \subseteq \tau_g$ be the branch in $\mathcal{T}$ created by the end of stage j. Let $S_j^* = S(\tau_j, \sigma)$, denote the information set added to I in stage j by I_j, and let $I_j^* = I(\tau_j, \sigma)$, or equivalently $I_j^* = \cup_{i=1}^{j} I_i$, be the information set gathered during the first j stages.

Let $K_j \supseteq J_j$ be the information set obtained from J_j by adding nonedge information pieces in direction δ incident to chosen centers $v \in \mathrm{supp}(J_j) \cap C_\sigma$ if

1. the information set J_j has an odd number of edge information pieces incident to v, and
2. there is no information piece in J_j incident to v in direction δ.

Note that since $\mathrm{supp}(J_j) = \mathrm{supp}(K_j)$ by Lemma 6.7 for $j \neq j'$ it holds that $\mathrm{supp}(K_j)$ and $\mathrm{supp}(K_{j'})$ are disjoint. This implies in particular that the information pieces in K_j and $K_{j'}$ are not in direct contradiction.

Let $K^* = \bigcup_{j=1}^{g} K_j$, and define ρ^* to be the restriction as defined in (15) that forces the same variables as (ρ, K^*) does. Observe that alive centers v of ρ with an odd number of edge information pieces incident in some information set J_j are now dead in ρ^*. The centers alive in ρ but dead in ρ^* are the *disappearing centers.*

Let a_j be the number of associated centers of variables on ψ_j that are also in $\mathrm{supp}(J_j) \setminus S_{j-1}^*$. Note that by Property 2 of Definition 6.2 all these associated centers are disappearing centers. Denote by b_j the number of additionally disappearing centers in K_j, and let $a = \sum_{j=1}^{g} a_j$, $b = \sum_{j=1}^{g} b_j$. We have the following relation between these parameters.

LEMMA 6.8. *It holds that $b \leq 3a$, and $s/64 \leq a$.*

Proof. For each stage $j \in [g]$ it holds that $\mathrm{supp}(J_j)$ contains at most $4a_j$ centers: each chosen center $u \in \mathrm{supp}(J_j)$ is either an associated center of a variable on ψ_j and in $u \in \mathrm{supp}(J_j) \setminus S_{j-1}^*$ or u is adjacent to such an associated center. For each of the former there are at most 3 of the latter since J_j is locally consistent. Similarly, since π has components of size 4, each nonchosen associated center causes at most 3 other nonchosen centers to be included in $\mathrm{supp}(J_j)$. Since $a_j + b_j \leq |\mathrm{supp}(K_j)| = |\mathrm{supp}(J_j)| \leq 4a_j$ we obtain the desired inequality $b \leq 3a$.

It remains to establish the inequality $s/64 \leq a$. By Lemma 6.5 and the just-established inequality $|\mathrm{supp}(J_j)| \leq 4a_j$ we get that $|S_g^*| \leq 4\sum_{i=1}^{g} |\mathrm{supp}(J_j)| \leq 16a$. Since $|S_g^*| \geq s/4$ we obtain the desired inequality. ∎

In the following section we prove the next lemma stating that a restriction ρ that causes the extended canonical decision tree to have a path of length at least s can be encoded using few bits, given ρ^* and $T_1, \ldots, T_m$. Put differently, the mapping from ρ

to ρ^* can be inverted with a bit of extra information. Recall that $\Delta = \Theta(n/n')$ is the number of centers in each sub-square.

LEMMA 6.9. *There is a constant $A > 0$ such that the following holds. Suppose we are given decision trees $T_1, \dots, T_m$ of depth at most t and ρ^*. Then*

$$a \log t + b \log \Delta + s \cdot A$$

many bits suffice to encode ρ.

Before diving into the proof of Lemma 6.9 let us verify that Lemma 4.2 indeed follows.

Proof of Lemma 4.2. We analyze the probability that a restriction ρ chosen uniformly from $R^{\mathrm{reg}}(k, n, n')$ gives rise to an extended canonical decision tree of length at least s. Let $m = n'^2\Delta$ be the total number of centers, and recall that the odd integer k is the total number of alive centers.

According to Lemma 6.9, for some absolute constant A, the number of restrictions ρ that give rise to an extended canonical decision tree of depth at least s can be upper bounded by the number of ways to choose a restriction ρ^* with $k - a - b$ alive centers times $t^a \Delta^b A^s$. Using Lemma 5.5 we can bound the probability of sampling such a ρ by

$$\sum_{a,b} \frac{|R(k-a-b, n, n')| \cdot t^a \Delta^b A^s}{|R^{\mathrm{reg}}(k, n, n')|} \leq \sum_{a,b} \left(\frac{A_0 \log n'}{\Delta} \right)^{a+b} \cdot t^a \Delta^b A^s \tag{15}$$

$$\leq \sum_a A_1^s \left(\frac{t \log n'}{\Delta} \right)^a \cdot \sum_b \log^b n' \tag{16}$$

$$\leq \sum_a A_2^s \left(\frac{t \log^4 n'}{\Delta} \right)^a \tag{17}$$

for appropriate constants A_0, A_1, and A_2. The final inequality relies on the bound $b \leq 2a$ from Lemma 6.8. As Lemma 6.8 further guarantees that $a \geq s/64$ and since the sum in (18) is a geometric series the claimed bound on the probability of the extended canonical decision tree reaching depth at least s follows. ∎

6.5. Proof of Lemma 6.9.

Outline. On a very high level, we want to remove the information set K^* from the partial restriction ρ^*. We commence in stages. In each stage we remove a single information set K_j from ρ^* by utilizing the shallow decision trees $T_1, \dots, T_m$ and some extra information. We need some further notation and a simple observation to give a detailed proof outline.

For convenience let $I_0^* = \emptyset$, and for $i > j$ denote by $I_j^i \subseteq I_j$ the information set obtained from I_j by removing any information piece that occurs (identically) in some possible forcing information $J_{j'}$ for $j' \geq i$. In other words, we let $I_j^i = I_j \setminus \left(\bigcup_{j'=i}^g J_{j'} \right)$ and hence it holds that $I_j^{j+1} \subseteq I_j^{j+2} \subseteq \cdots \subseteq I_j^{g+1} = I_j$. Let $I_{j-1}^{*-} = \bigcup_{i=1}^{j-1} I_i^j$. By the previous observation we have that $I_g^{*-} = I_g^*$.

For $i < j$ denote by $K_j^i \subseteq K_j$ the information set obtained from K_j by removing any information piece in direct contradiction with I_i^*: remove information pieces $(v, \delta, \perp)$ if and only if I_i^* contains an edge incident to v in direction δ. Since $J_j \subseteq K_j$ is pairwise locally consistent with I_{j-1}^*, it holds that $J_j \subseteq K_j^{j-1} \subseteq K_j^{j-2} \subseteq \cdots \subseteq K_j^0 = K_j$. Let $K_{\geq j}^{*-} = \bigcup_{i=j}^g K_i^{j-1}$, and note that $K_{\geq 1}^{*-} = \bigcup_{i=1}^g K_i^0 = \bigcup_{i=1}^g K_i = K^*$.

Denote by ρ_j^* the restriction obtained from composing ρ with the information set $I_{j-1}^{*-} \cup K_{\geq j}^{*-}$ as done in (15). Note that ρ_j^* forces the same variables as $(\rho, I_{j-1}^{*-} \cup K_{\geq j}^{*-})$ forces, observe that $\rho_1^* = \rho^*$, and note that ρ_{g+1}^* forces the same variables as (ρ, I_g^*) forces.

LEMMA 6.10. *The restriction ρ_j^* traverses the forceable branch ψ_j of stage j.*

Proof. Recall from Definition 6.2, property 3, that $(\rho, I_{j-1}^* \cup J_j)$ forces all the variables on the forceable branch ψ_j of stage j such that ψ_j is traversed. As the information set K_j^{j-1} extends the forcing information J_j and is not in direct contradiction with I_{j-1}^* we observe that also $(\rho, I_{j-1}^* \cup K_j^{j-1})$ traverses ψ_j. Furthermore, since $K_{\geq j}^{*-}$ extends K_j^{j-1} and is also not in direct contradiction with I_{j-1}^*, we conclude that $(\rho, I_{j-1}^* \cup K_{\geq j}^{*-})$, equivalently ρ_j^*, traverses the forceable branch ψ_j. ∎

Lemma 6.10 allows us to pursue the following high-level plan. We proceed in stages $j = 1, 2, \ldots, g$. At the beginning of each stage j we assume that we know the restriction ρ_j^* and the information set I_{j-1}^{*-}. Since $I_0^{*-} = \emptyset$ and $\rho_1^* = \rho^*$ we have the necessary information to start with stage $j = 1$. Furthermore, if we can complete these g stages, we obtain the restriction ρ_{g+1}^* from which, along with $I_g^{*-} = I_g^*$, we can recover the sought-after restriction ρ: since ρ_{g+1}^* forces the same variables as (ρ, I_g^*) forces, we can "remove" I_g^* from ρ_{g+1}^* to obtain ρ.

Let us consider a stage $j \in [g]$. By Lemma 6.10 the restriction ρ_j^* traverses the forceable branch ψ_j. Let us assume, for now, that ψ_j is the first 1-branch traversed. This assumption allows us to identify ψ_j for free. Once identified we can use the branch ψ_j to cheaply recover a good fraction of the support of the forcing information J_j: since the branch ψ_j is of length at most t we can point out the variables on the branch ψ_j forced by J_j at cost $\log t$ each. From these variables we can recover their unique associated centers for *free*. Each of these associated centers u is a disappearing center: by property 2 of Definition 6.2 the set J_j is closed at u and since J_j is locally consistent (property 1 of Definition 6.2) it has an odd number of edges incident to u.

To find the other centers in $\mathrm{supp}(J_j)$ and to recover the structure of J_j, i.e., whether there are edges or nonedges in between centers of $\mathrm{supp}(J_j)$, we rely on external information: we read $\log \Delta$ bits per disappearing center and a constant number of bits per potential edge. Once we have recovered J_j it is straightforward to obtain K_j^{j-1}. "Remove" K_j^{j-1} from ρ_j^*, and add all information pieces from J_j to I_{j-1}^{*-} that are common to J_j and I_{j-1}^*. By adding all these information pieces to I_{j-1}^{*-} we obtain the information set $\bigcup_{i=1}^{j-1} I_i^{j+1}$. Before we can proceed to stage $j+1$ we need to recover I_j^{j+1} so that we can create the information set $I_j^{*-} = \bigcup_{i=1}^{j} I_i^{j+1}$.

The support of the information set I_j consists of the centers $S_j = S_j^* \setminus S_{j-1}^*$ added to the set of exposed centers and some further centers at distance 1 from these newly exposed centers S_j. Most of the centers in S_j are readily identified as follows. Recall that the set S_j consists of $\mathrm{supp}(J_j)$ along with the chosen centers at distance at most 1 from $\mathrm{supp}(J_j) \cap C_\sigma$. For each chosen center in S_j at distance 1 from $\mathrm{supp}(J_j) \cap C_\sigma$ we read one bit of extra information to determine whether it has disappeared (since these may be in the support of some forcing information $K_{j'}$ for $j' > j$). If it has not disappeared, then it is readily verified as the alive center with the lowest numbered row. Otherwise we can afford to read $\log \Delta$ extra information per such center to identify it. This identifies the exposed centers in the support of I_j.

Recall that there may be some edge information pieces in I_j between an exposed chosen center $u \in S_j$ and a chosen center $v \notin S_j^*$ that is not exposed. Either

1. the edge $\{u, v\}$ is shared with a forcing information $J_{j'}$ for $j' > j$ and is thus not in I_j^{j+1}, or
2. the information piece $\{u, v\}$ contradicts some $K_{j'}$ for $j' > j$, or
3. the chosen center v is in no support $\mathrm{supp}(K_{j'})$ for $j' > j$ and is thus alive.

In case 1 we do not have to identify the chosen center v since we will recover it in stage j'. In case 3 the chosen center v is identified for free since it is the alive center with the lowest numbered row. For case 2 note that since $J_{j'}$ and I_j are locally consistent (property 1 of Definition 6.2), the contradiction is due to an information piece in $K_{j'} \setminus J_{j'}$. This implies that the center v is *not* an associated center of stage j': by property 2 of Definition 6.2 the forcing information $J_{j'}$ has information pieces in all directions incident to such centers. We may thus identify v by reading $\log \Delta$ bits of extra information to then "fix" the contradiction by removing the nonedge and adding the edge $\{u, v\}$.

All that remains is to recover the structure of I_j^{j+1}. We read a constant amount of extra information per potential edge. This identifies I_j^{j+1}. Proceed with stage $j+1$.

Let us tally the amount of extra information read. We have $a \log t$ bits per disappearing center that is also an associated center of a variable on a forceable branch, for other disappearing centers we pay $b \log \Delta$, and finally for the structure of the different information sets we need another $A|S_g^*|$ bits for some constant A. Thus in total, as claimed, we need at most $a \log t + b \log \Delta + A|S_g^*|$ bits to recover ρ. This completes the proof overview.

Setup. Unfortunately there are some complications. Recall that the forceable branch ψ_j is defined to be the next *pairwise locally consistent* branch with τ_{j-1} and σ in stage j of the construction of the decision tree $\mathcal{T}$. In particular, the branch ψ_j along with τ_{j-1} needs to be pairwise locally consistent as assignments on the smaller $n' \times n'$ grid. At this point it is not clear how to determine whether a given branch satisfies this property. Hence the first branch traversed by ρ_j^* is not necessarily the forceable branch ψ_j of stage j. We need a way to cheaply tell that a given branch is not the forceable branch. To this end[5] we introduce signatures.

DEFINITION 6.11 (signature). *Let v be a center in the support of J_j. The* signature *of such a center v consists of* 9 *bits.*

1. *The first bit is* 1 *if and only if v is a chosen center.*
2. *The following* 4 *bits indicate in what directions J_j has information pieces incident to v.*
3. *The final* 4 *bits indicate for each direction whether there is an edge in J_j incident to v.*

Remark 6.12. Note that the stage j, although mentioned in the definition, is *not* part of the signature (as it was in [8]). This change is mandated by our desire to get a tighter bound which requires that the signature of a single center is of constant size. Furthermore, note that the signature of v does not include the identity of v. In the following we only read signatures in combination with a fixed center v; we assume that the signatures are ordered on the auxiliary information as we process these centers.

Since the information sets J_j and $J_{j'}$ are disjoint (see Lemma 6.7) and we additionally have that $\mathrm{supp}(J_j) = \mathrm{supp}(K_j)$, it holds that each disappearing center in the support of K^* has a unique signature. Also, since $(\rho, I_{j-1}^* \cup J_j)$ forces all variables on

[5] We also use signatures to ensure that only branches forced by J_j are considered and not branches forced by information pieces in $K_j^{j-1} \setminus J_j$.

the forceable branch ψ_j, every variable on ψ_j that is not forced by (ρ, I^*_{j-1}) has an associated center in J_j with a signature. Finally note that only nodes in the support of K^* require signatures. As such we can afford to read all the signatures: at most $9|\mathrm{supp}(K^*)| \le 200s$ bits need to be read.

Note that a chosen center v along with its signature sign defines a partial assignment to the incident path variables: the first set of 4 bits of sign indicates the direction and the final 4 bits indicate the assignment to the incident path-variables.

The idea is that signatures indicate what the next forceable branch should look like: they indicate where the branch is supposed to have variables and what the corresponding assignment on the smaller grid looks like. If a 1-branch ψ contradicts this information, then we say that ψ is in *conflict* with these signatures. The formal definition follows.

DEFINITION 6.13 (conflict). *Let I be an information set, denote by ψ a branch, and let E be a set of tuples (v, sign) each consisting of a center v along with its signature* sign. *Denote by $E_\psi \subseteq E$ the subset of tuples (v, sign) such that v is a chosen center (i.e., the first bit of* sign *is* 1*) and there is a variable x_e on ψ such that v is the associated center of x_e. The set E is in* conflict *with ψ and I if and only if either*

1. *there is an associated center v of a variable on ψ such that the partial assignment induced by I along with the signature of v is not defined in all directions incident to v, or*
2. *the partial assignment on chosen path variables obtained from I jointly with the assignments defined by the tuples in E_ψ is not pairwise locally consistent as assignments on the smaller grid.*

Reconstruction. Let us explain how signatures are used to recover the forceable branch ψ_j of stage j. See Algorithm B.2 for pseudocode of the procedure described in the following. Throughout the procedure we maintain the following objects: a counter $j = 1, 2, \ldots, g$ of the current stage to be reconstructed, the restriction ρ^*_j, the information set I^{*-}_{j-1}, the exposed centers S^*_{j-1}, and a set E of (prematurely identified) disappearing centers along with their signatures. Initially we set $j = 1$, $\rho^*_1 = \rho^*$, and $S^*_0 = I^{*-}_0 = E = \emptyset$. We proceed as follows.

1. Find the next 1-branch ψ traversed by the restriction ρ^*_j.
2. If ψ and I^{*-}_{j-1} are in conflict with E, then go to step 1.
3. Read a bit b of extra information to determine whether there is a disappearing center that is the associated center of a variable on ψ.
4. If $b = 1$, then read an integer i of magnitude at most t. This identifies the associated center v of the ith variable on ψ as a disappearing center. Read the signature sign of v and add (v, sign) to E. If E is in conflict with ψ and I^{*-}_{j-1}, then go to step 1. Otherwise go to step 3.
5. If $b = 0$, then ψ is the forceable branch of stage j. Recover J_j, the information set K^{j-1}_j, as well as I^{j+1}_j essentially as discussed in the proof overview (details provided below). Update ρ^*_j to ρ^*_{j+1}, I^{*-}_{j-1} to I^{*-}_j, and S^*_{j-1} to S^*_j, remove all associated centers of ψ from E, and set $j = j + 1$. Ensure that E contains all the signatures (v, sign) of chosen centers $v \in S^*_{j-1} \cap \mathrm{supp}(K^{*-}_{\ge j})$. If $|S^*_{j-1}| \ge s/4$, then terminate. Otherwise go to step 1.

This completes the description of the procedure used to recover the forceable branch of stage j.

The following lemma guarantees that the above procedure is correct, that is, that it identifies the forceable branch of stage j.

LEMMA 6.14. *Let E be the set of all tuples (v, sign) that consist of a disappearing center $v \in \text{supp}(K^{*-}_{\geq j})$ or an exposed chosen center in $v \in S^*_{j-1} \cap \text{supp}(K^{*-}_{\geq j})$ along with their signature* sign. *If ψ is the first* 1*-branch traversed by ρ^*_j such that E is* not *in conflict with I^{*-}_{j-1} and ψ, then ψ is the forceable branch ψ_j of stage j.*

Proof. We need to establish that E is in conflict with I^{*-}_{j-1} and all 1-branches ψ occurring before ψ_j. Toward contradiction suppose otherwise: let ψ be a 1-branch before ψ_j such that

1. the restriction ρ^*_j forces all variables on ψ such that the respective leave is reached, and
2. the set E is not in conflict with I^{*-}_{j-1} and ψ.

Note that also the restriction defined by composing ρ with $I^*_{j-1} \cup \bigcup^g_{j'=j} J_{j'}$ would traverse ψ: a center u that does not have information pieces in all directions incident in $I^*_{j-1} \cup \bigcup^g_{j'=j} J_{j'}$ does not force any incident variables. In this case the assignment induced by I^{*-}_{j-1} along with the signature sign of u is not defined in all directions incident to u. Hence since E is not in conflict with I^{*-}_{j-1} and ψ, it holds that all variables forced by $K^{*-}_{\geq j}$ on ψ are incident to centers $u \in \text{supp}(J_{j'})$ for $j' \geq j$ with information pieces in all directions incident to u present in $I^*_{j-1} \cup J_{j'}$.

Let us construct a possible forcing information J'_j that could have been used in stage j of the construction of the extended canonical decision tree to force the branch ψ. On the nonchosen centers the set J'_j contains the pieces of π needed to force all variables on ψ. On the chosen centers the set J'_j consists of information pieces as given by the partial assignments defined by signatures $(v, \text{sign}) \in E$ such that there is a variable on ψ whose associated center is v. These information pieces are pairwise locally consistent with I^{*-}_{j-1} as E is not in conflict with I^{*-}_{j-1} and ψ. Furthermore, these force the input to traverse ψ as these information pieces are the same as used in $I^*_{j-1} \cup \bigcup^g_{j'=j} J_{j'}$. ∎

Lemma 6.14 shows that the above procedure, once it reaches step 5, indeed identifies the forceable branch ψ_j of stage j. It remains to argue that the information sets J_j, K^{j-1}_j, and I^{j+1}_j can be recovered. We closely follow the argument presented in the proof outline, and there is pseudocode in Appendix B in case of ambiguity in the following verbal discussion.

We are given the restriction ρ^*_j, the information set I^{*-}_{j-1}, and all the signatures of the disappearing centers that are also associated centers of variables on the forceable branch ψ_j of stage j. We need to construct the restriction ρ^*_{j+1} and the information set I^{*-}_j.

We start with the reconstruction of the forcing information J_j, then explain how to obtain K^{j-1}_j from J_j, and finally argue that we can recover I^{j+1}_j from J_j.

The unique associated centers of the branch ψ_j identify a good part of the support of J_j. For each associated center $u \in \text{supp}(J_j)$ that is also *chosen* we read the up to three incident centers used to make J_j closed at u. Each is read at cost $\log \Delta$ unless it has already been identified. If it has already been identified, then it is the center in the appropriate sub-square whose first bit in the signature is 1; we can identify it at cost at most $\log t \leq \log \Delta$. Reading one bit per information piece on chosen centers in J_j, we obtain the structure of J_j on the chosen centers.

This identifies all of J_j on the chosen centers. On the nonchosen centers we may need to complete some connected components from the pairing π. We encode the structure of the connected component using a constant number of bits. Each node in such a connected component can again be recovered at cost at most $\log \Delta$. This identifies all of J_j.

We obtain K_j^{j-1} from J_j by adding nonedge information pieces incident to a center $u \in \mathrm{supp}(J_j)$ in direction δ if and only if

1. the center u has an odd number of edges incident in J_j, and
2. there are no information pieces in $I_{j-1}^{*-} \cup J_j$ in direction δ from u.

"Undo" the information set K_j^{j-1} from ρ_j^* by flipping the assignment along all the edges in K_j^{j-1} and add all information pieces from J_j to I_{j-1}^{*-} that are common between J_j and I_{j-1}^*. This results in the information set $\bigcup_{i=1}^{j-1} I_i^{j+1}$. It remains to recover I_j^{j+1}.

Recall that the support of the information set I_j consists of the centers $S_j = S_j^* \setminus S_{j-1}^*$ added to the set of exposed centers and some further chosen centers at distance 1 from S_j. The nonchosen centers in S_j are readily identified: these are $\mathrm{supp}(J_j) \setminus C_\sigma$. It remains to identify the chosen centers in S_j at distance precisely 1 from $\mathrm{supp}(J_j) \cap C_\sigma$. Check E for whether any of the remaining chosen centers has already been identified (we know the relevant sub-squares; check whether there is a tuple $(v, \mathrm{sign}) \in E$ such that v is in one of these sub-squares and the first bit of sign is 1). These are identified at cost $\log t \le \log \Delta$. For each of the remaining sub-squares we read 1 bit of extra information to determine whether the chosen center is alive. If so, then it is the alive center with the lowest numbered row. Otherwise we read $\log \Delta$ extra information to identify it. Note that since such a center was exposed in stage j it cannot be a disappearing associated center of some stage $j' > j$ and we can thus afford to read these $\log \Delta$ bits. This identifies the exposed centers in the support $\mathrm{supp}(I_j)$.

Recall that there may be some edge information pieces in I_j between an exposed chosen center $u \in S_j$ and a chosen center $v \notin S_j^*$ that is not exposed. Either

1. the edge $\{u, v\}$ is shared with a forcing information $J_{j'}$ for $j' > j$ and is thus not in I_j^{j+1}, or
2. the information piece $\{u, v\}$ contradicts some $K_{j'}$ for $j' > j$, or
3. the chosen center v is in no support $\mathrm{supp}(K_{j'})$ for $j' > j$ and is thus alive.

In case 1 we do not have to identify the chosen center v since we will recover it in stage j'. In case 3 the chosen center v is identified for free since it is the alive center with the lowest numbered row. In case 2 we identify v at cost $\log \Delta$ (unless it has already been found and is in E). Since in case 2 the chosen center v does not have information pieces incident in all directions in J_j, it is not an associated center (by property 2 of Definition 6.2) and we can hence afford to pay $\log \Delta$ bits for its identification.

It remains to recover the structure of I_j^{j+1}. We read a constant amount of extra information per potential edge. This identifies I_j^{j+1}. This concludes the discussion of how one recovers J_j and I_j^{*-}. Let us tally the external information needed.

Recall that a_j is the number of disappearing centers outside S_{j-1}^* that are an associated center of a variable of ψ_j, b_j is the number of other disappearing centers of K_j, and we let $a = \sum_{j=1}^g a_j$ and $b = \sum_{j=1}^g b_j$. The following summarizes the amount of external information needed.

- The disappearing centers discovered as an associated center of a forceable branch contribute $a \log t$ bits.
- The other disappearing centers contribute at most $b \log \Delta$ bits.
- For each center in $\mathrm{supp}(K^*)$ we may have to read the signature. These are at most $9|S_g^*|$ bits.
- All other centers are discovered at constant cost; we only read $A_1|S_g^*|$ bits for some constant A_1.

- For the graph structure of J_j and I_j^{*-} we need another $A_2|S_g^*|$ bits for some constant A_2.
- There is a constant $A_3 > 0$ such that in the above procedure at most $s+16t+s/4 = s \cdot A_3$ bits b are read:
  - there are at most $s+16t$ bits that are 1 as each time a disappearing variable is discovered, and this is bounded by Corollary 6.6, and
  - at most s bits that are 0 as a stage is ended each time and $g \le s/4$.

Since $|S_g^*| \le s/4 + 16t \le s \cdot A_4$ for some other constant A_4 this completes the proof of Lemma 6.9. As Lemma 6.9 is the last missing piece of the proof of the switching lemma, we thereby also establish Lemma 4.2.

7. Proof of the multiswitching lemma. The purpose of this section is to prove the multiswitching lemma, restated here for convenience.

LEMMA 4.4 (multiswitching lemma). *There are constants $A, c_1, c_2, n_0 > 0$ such that for integer $n \ge n_0$ the following holds. Let $k, M, n', s, t \in \mathbb{N}^+$ satisfy $n/n' \ge At\log^{c_1} n$, $k = n'^2(1 \pm 0.01)C\log n'$ be odd, and $t \le s \le n'/32$. For $m_1, \dots, m_M \in \mathbb{N}^+$ and any decision trees T_i^j of depth at most t, where $j \in [M]$ and $i \in [m_j]$, it holds that if $\sigma \sim \mathcal{D}_k\big(\Sigma(n,n')\big)$, then the probability that $\big(\bigvee_{i=1}^{m_j} T_i^j\restriction_\sigma\big)_{j=1}^{M}$ cannot be represented by an ℓ-common partial decision tree of depth s is bounded by*

$$M^{s/\ell}\left(\frac{At\log^{c_1} n}{n/n'}\right)^{s/c_2}.$$

The proof of Lemma 4.4 very much follows the proof of Lemma 4.2. The first section gives a short proof overview, followed by subsection 7.2 that formally explains how to construct a common partial decision tree. In subsection 7.3 we finally prove Lemma 4.4.

7.1. Proof overview. The high-level proof outline is as follows. Consider the disjunctions of decision trees $\bigvee_{i=1}^{m_1} T_i^1, \bigvee_{i=1}^{m_2} T_i^2, \dots, \bigvee_{i=1}^{m_M} T_i^M$ in order. If such a disjunction $\bigvee_{i=1}^{m_j} T_i^j$ is not turned into a decision tree of depth ℓ, then find a branch λ^j in the extended canonical decision tree of $\bigvee_{i=1}^{m_j} T_i^j$ of length at least ℓ. Put the variables on λ^j in the common partial decision tree, query these variables as well as some extra variables, and recurse.

As in the proof of the switching lemma we consider any full restriction $\sigma = \sigma(\rho, \pi)$ for which Lemma 4.4 fails. We then turn ρ into a restriction ρ^* such that the mapping can be inverted with little extra information to argue that there are few full restrictions for which Lemma 4.4 fails.

The construction of ρ^* is analogous to the construction used in the proof of the switching lemma: fix a long branch τ in the ℓ-common partial decision tree, and consider the long branches $\lambda^1, \lambda^2, \dots$ from the respective extended canonical decision trees used to construct τ. Each such λ^j was constructed with the help of possible forcing information $J_1^j, J_2^j, \dots$. Close up each such J_i^j as before to obtain K_i^j, and let $\rho^* = (\rho, \bigcup_{i,j} K_i^j)$.

We recover ρ from ρ^* by iteratively recovering the possible forcing information $J_1^j, J_2^j, \dots$ as in the standard switching lemma, to obtain information sets $I_1^j, I_2^j, \dots$ describing the long branch λ^j. We then read a bit of extra information to recover the sub-assignment τ^j of τ to the variables assigned by λ^j.

There is one minor complication to handle: the recovered information sets $\bigcup_i I_i^j$ of a branch λ^j may be *in*consistent with future forcing information; a set I_i^j may be

inconsistent on chosen centers with some $J_{i'}^{j'}$ for $j' > j$. This is so because the set $J_{i'}^{j'}$ is consistent with τ^j but not necessarily with the branch λ^j.

We handle this complication by not just querying the variables of λ^j in the common partial decision tree but by querying all variables with an associated center at distance at most 1 from the exposed chosen centers of λ^j. Analogously to how we proved in Lemma 6.7 that different possible forcing information pieces have disjoint support, it can be shown that future $J_{i'}^{j'}$ have a support disjoint of the support of some I_i^j with $j < j'$. This implies in particular that $J_{i'}^{j'}$ cannot contain information pieces in direct contradiction with I_i^j.

7.2. Common partial decision trees. Let us explain how to construct the ℓ-common partial decision tree $\mathcal{T}$ of $\bigvee_{i=1}^{m_1} T_i^1\restriction_\sigma, \ldots, \bigvee_{i=1}^{m_M} T_i^M\restriction_\sigma$. Start with $\mathcal{T}$ empty. We proceed in *rounds.* In each round we consider a leaf τ of $\mathcal{T}$ such that there is a $\bigvee_{i=1}^{m_j} T_i^j$ that cannot be represented by a depth ℓ decision tree under σ and τ. Extend $\mathcal{T}$ at τ as follows.

Let j be minimum such that $\bigvee_{i=1}^{m_j} T_i^j\restriction_{\sigma\tau}$ cannot be represented by a depth-ℓ decision tree. Create the extended canonical decision tree T^j of $\bigvee_{i=1}^{m_j} T_i^j\restriction_{\sigma\tau}$ essentially as in subsection 6.2—more details follow. Denote by λ a branch of length at least ℓ in T^j, and extend $\mathcal{T}$ at τ by querying all variables on λ. Modulo the precise definition of the extended canonical decision tree used this describes the entire creation process of an ℓ-common partial decision tree.

Let us discuss how to construct the extended canonical decision trees in the above procedure. The only difference from the definition in subsection 6.2 is that we initialize the set of exposed centers S and the information set I used in the creation of the extended canonical decision tree with information from previous rounds. Let us explain this in more detail.

Throughout the creation of the ℓ-common partial decision tree $\mathcal{T}$ we maintain the following sets for each leaf τ of $\mathcal{T}$:

1. a set of exposed centers $S = S(\tau, \sigma)$, and
2. a set of information pieces $I = I(\tau, \sigma)$.

Initially the set of exposed centers S and the information set I are empty $S(\emptyset, \sigma) = T(\emptyset, \sigma) = \emptyset$. Throughout the creation of $\mathcal{T}$ and the various extended canonical decision trees we maintain the same invariants as in Invariant 6.1.

In each round of the construction of $\mathcal{T}$, when building the extended canonical decision tree T^j of $\bigvee_{i=1}^{m_j} T_i^j$, we initialize the sets S and I used in the creation of T^j with the corresponding objects maintained for the creation of the common partial decision tree $\mathcal{T}$. Other than that, the creation of T^j follows subsection 6.2: in each stage we find a new forceable branch ψ_i^j and the corresponding forcing information J_i^j and add all nodes in $\operatorname{supp}(J_i^j)$ to S along with the chosen centers adjacent to $\operatorname{supp}(J_i^j) \cap C_\sigma$.

To find out whether the forceable branch ψ_i^j is followed we get information sets I_i^j consisting of pieces from π and answers from T^j.

We continue with the next stage until at least $\ell/4$ centers have been added in this round to the set of exposed centers $S(\lambda^j, \sigma)$ for some leaf λ^j of T^j. We know that this happens as $\bigvee_{i=1}^{m_j} T_i^j\restriction_{\sigma\tau}$ could not be decided by a decision tree of depth ℓ (recall that τ is the leaf of $\mathcal{T}$ we are considering).

For such a long branch $\lambda^j \in T^j$ denote by S_{λ^j} all chosen centers at distance at most 1 from the newly exposed chosen centers, that is, S_{λ^j} consists of all chosen centers at distance at most 1 from $\big(S(\lambda^j, \sigma) \setminus S(\tau, \sigma)\big) \cap C_\sigma$. Extend the common partial decision tree $\mathcal{T}$ at τ by querying all variables incident to S_{λ^j}. For each newly

created leaf $\tau' \in \mathcal{T}$ we need to update the set of exposed centers S and the information set I: let $S = S(\lambda^j, \sigma) \cup S_{\lambda^j}$ and set I according to the answers on $\tau' \setminus \tau$ while also including the same information about π as is present in $I(\lambda^j, \sigma)$. This completes the description of the construction of the common partial decision tree $\mathcal{T}$.

Note that at the end of each round the information pieces in I are determined by the branch τ of $\mathcal{T}$ and the matching π. The information pieces $I_1^j, I_2^j, \ldots$ on the chosen centers used to determine λ^j in T^j are "forgotten." These answers were only used to find the long branch λ^j.

Clearly the above process creates an ℓ-common partial decision tree $\mathcal{T}$. We need to analyze the probability that we obtain a tree of depth at least $\text{depth}(\mathcal{T}) \geq s$.

7.3. Encoding cost. Once we have set up the machinery the proof parallels the proof of the standard switching lemma. We need to verify that it works and no new complications arise.

As in subsection 6.4 we extend each possible forcing information J_i^j to information sets K_i^j by adding nonedges incident to nodes in $\text{supp}(J_i^j)$ that have an odd number of edges incident in J_i^j. The restriction ρ^* is obtained by applying these K_i^j to ρ. We need to specify the information needed to invert this mapping, that is, the information needed to recover ρ from ρ^*.

The inversion commences in rounds. Each round essentially corresponds to the inversion process of the standard switching lemma (see subsection 6.5). The following extra information is read per round.

- We require $\log M$ bits to obtain an index $j \in [M]$ that identifies $\bigvee_{i=1}^{m_j} T_i^j$ being processed.
- The following information read is identical to the inversion process of $\bigvee_{i=1}^{m_j} T_i^j$ as in the standard switching lemma.
- Once we have the long branch λ^j of T^j we need to recover the chosen centers v at distance at most 1 from the newly exposed chosen centers. We read at most $\log \Delta$ bits for each such v. Since there are at most linear in s many such nodes we can afford this—the cost is absorbed by the constants c_1 and c_2.
- Finally we read the difference in values of variables queried in the decision tree T^j and the same variables in the common decision tree $\mathcal{T}$. Note that analogous to cases 1–3 there will be some information pieces which will be recovered in future stages.

The inversion process of each round runs parallel to the inversion for the standard switching lemma. We recover the information pieces used in the single formula process and use the knowledge of the difference to turn these into information pieces for the common decision tree.

Of the additional extra information needed (that is, the index $j \in [m]$ for each round, the identity of the additional chosen centers, and the differences in values) only the index j cannot be absorbed by the constants A, c_1, and c_2. This extra information causes the factor $M^{s/\ell}$ in the bound of Lemma 4.4. This completes the discussion of the proof of Lemma 4.4.

8. Conclusion. Of course our bounds are not exactly tight so there is always room for improvement. We could hope to get truly exponential lower bounds for a bounded depth Frege proof, i.e., essentially bounds 2^n, where n is the number of variables. Since any formula given by a small CNF has a resolution proof this is the best we could hope for. As our formulas have $O(n^2)$ variables we are off by a square.

If one is to stay with the Tseitin contradiction one would need to change the graph, and the first alternative that comes to mind is an expander graph. We have not really studied this question, but as our current proof relies heavily on properties of the grid, significant modifications are probably needed.

This brings up the question for which probability distributions of restrictions it is possible to prove a (multi) switching lemma. Experience shows that this is possible surprisingly often. It seems, however, that it needs to be done on a case by case basis. It is probably too much to ask for a general characterization, but maybe it could be possible to prove switching lemmas that cover several of the known cases.

Appendix A. Omitted proofs.

A.1. Section 2.

LEMMA 2.1. *Consider the Tseitin formula* $\mathrm{Tseitin}(G_n, \alpha)$ *defined over the* $n \times n$ *grid. If* $\sum_v \alpha_v$ *is even, then* $\mathrm{Tseitin}(G, \alpha)$ *is satisfiable and has* 2^{r_n} *solutions for a positive integer* r_n *that only depends on* n *and not on* α*.*

Proof. Let us first argue that the formula $\mathrm{Tseitin}(G_n, \alpha)$ is satisfiable if $\sum_v \alpha_v$ is even. Toward contradiction suppose this is not the case, and consider an assignment β that satisfies the maximum number of linear constraints. Note that the number of such violated linear constraints is even since the sum of the constraints

$$\sum_v \sum_{e \ni v} x_e = 2 \sum_e x_e \tag{18}$$

is always even. Hence if for a node v it holds that $\sum_{e \ni v} \beta_e \neq \alpha_v$, then there is another node $u \neq v$ such that $\sum_{e \ni u} \beta_e \neq \alpha_u$. Suppose there are two such nodes u and v.

Since the graph G_n is connected we may consider a path P connecting u to v. By negating the assignment β along the path P we obtain the assignment β' that satisfies the constraints at u and v. Furthermore, the parity of the edges incident to other nodes remains the same since we negated an even number of variables incident to every other node. This contradicts the assumption that β is an assignment that satisfies the maximum number of linear constraints.

For the other part of the lemma we only need to recall that the number of satisfying assignments to a satisfiable system of linear equations only depends on the dimension of the system and not on the right-hand side. This establishes the claim. ∎

LEMMA 2.3. *If* α *is a locally consistent assignment satisfying* $|\mathrm{supp}(\alpha)| \leq n/2$*, then for any variable* x_e *there is a locally consistent assignment* $\alpha' \supseteq \alpha$ *with* x_e *in its domain.*

Proof. Denote the support of α by $U = \mathrm{supp}(\alpha)$, and let $V \supseteq U$ be the set of nodes that consists of U and the endpoints of e. Since α is locally consistent we may consider an extension $\beta \supseteq \alpha$ that satisfies all the constraints on the nodes in $\mathrm{closure}(U)$.

Let $\gamma \supseteq \beta$ be an extension of β to all variables incident to the nodes in $\mathrm{closure}(V)$ that satisfies the maximum number of constraints on the nodes in $\mathrm{closure}(V)$. Suppose γ violates the linear constraint of some node $v \in \mathrm{closure}(V)$.

Note that $v \in \mathrm{closure}(U)^c$ since $\beta \subseteq \gamma$ satisfies all the constraints on $\mathrm{closure}(U)$. Since the component $\mathrm{closure}(U)^c$ is connected we may consider a path P that starts at v, ends in the giant component $\mathrm{closure}(V)^c \subseteq \mathrm{closure}(U)^c$, and does not pass through any node in $\mathrm{closure}(U)$. Negate the assignment γ along the edges of P to

obtain γ'. Note that the assignment γ' satisfies the constraint on v, extends β, and causes no new violations on closure$(V) \setminus$ closure(U) since it negates an even number of variables incident to all nodes in closure$(V) \setminus \{v\}$. This is in contradiction to the initial assumption that γ satisfies the maximum number of constraints on the nodes in closure(V).

We conclude that there is an extension $\gamma \supseteq \alpha$ that is locally consistent and contains the variable x_e it its domain. The statement follows. ∎

LEMMA 2.13. *Let $n, t \in \mathbb{N}$ such that $t \leq n/16$, and suppose that we have a Frege proof of a formula A from the Tseitin formula* Tseitin(G_n) *defined over the $n \times n$ grid. If this proof has a t-evaluation, then each line in the derivation is mapped to a* 1*-tree. In particular $A \neq \bot$, that is, contradiction cannot be derived.*

Proof. We proceed by induction over the number of derivation steps. Consider the formula F derived on line ν of the proof. Toward contradiction suppose that the t-evaluation φ^ν of line ν does not assign F to a 1-tree. That is, the decision tree $\varphi^\nu(F)$ contains a branch τ that ends in a 0-leaf. Since axioms are mapped to 1-trees (property 2 of Definition 2.11) the formula F has to be derived by a rule in the Frege system as listed in subsection 2.1. The idea is to consider each rule separately and show that the 0-branch τ causes a 0-branch in one of the t-evaluations of a line used to derive F. This contradicts the inductive hypothesis.

The assumption that all decision trees are of depth less than $n/16$ ensures by virtue of Corollary 2.7 that it is always possible to find a locally consistent branch in any decision tree restricted by τ. We do a case distinction on the rule used to derive F.

Exculuded middle. We have $F = p \vee \neg p$. By Definition 2.11, property 3, the decision tree $T_p = \phi^\nu(p)$ is equal to $T_{\neg p} = \varphi^\nu(\neg p)$ except that the labels of the leaves are negated. Since $\varphi^\nu(F)$ represents $T_p \vee T_{\neg p}$ by property 4 of Definition 2.11, the two restricted decision trees $T_p\lceil_\tau$ and $T_{\neg p}\lceil_\tau$ are both 0-trees. This cannot be since the two restricted trees are the same except that the labels at the leaves are negated.

Expansion rule. We have $F = q \vee p$. Let $T_p = \varphi^\nu(p)$. By property 4 of Definition 2.11 the decision tree $\varphi^\nu(F)$ represents $\varphi^\nu(q) \vee T_p$. This implies in particular that the decision tree $T_p\lceil_\tau$ is a 0-tree.

Denote by $\nu' < \nu$ the line used to derive F. Note that p is the formula on line ν' and that by the inductive hypothesis the decision tree $T'_p = \varphi^{\nu'}(p)$ is a 1-tree. Hence $T'_p\lceil_\tau$ is also a 1-tree. By Lemma 2.9 this contradicts the assumed functional equivalence of the t-evaluations φ^ν and $\varphi^{\nu'}$.

Contraction rule. We have $F = p$. Consider the formula $p \vee p$ on line $\nu' < \nu$ used to derive F. Let $T'_p = \varphi^{\nu'}(p)$. Since φ^ν and $\varphi^{\nu'}$ are functionally equivalent, it holds that $T'_p\lceil_\tau$ is a 0-tree. Hence $\varphi^{\nu'}(p \vee p)\lceil_\tau$ is a 0-tree by property 4 of Definition 2.11. This contradicts the inductive hypothesis which asserts that each line before ν is mapped to a 1-tree.

Association rule. We have $F = (p \vee q) \vee r$. Consider the formula $F' = p \vee (q \vee r)$ on line $\nu' < \nu$ used to derive F. Since F and F' are isomorphic by Definition 2.12 the two decision trees $\varphi^\nu(F)$ and $T_{F'} = \varphi^{\nu'}(F')$ are functionally equivalent. This implies that $T_{F'}\lceil_\tau$ is a 0-tree. This is in direct contradiction to the inductive hypothesis.

Cut rule. We have $F = (q \vee r)$. Let $T_q = \varphi^\nu(q)\lceil_\tau$ and $T_r = \varphi^\nu(r)\lceil_\tau$. Since by property 4 of Definition 2.11 the decision tree $\varphi^\nu(F)$ represents $T_q \vee T_r$ it holds that $T_q\lceil_\tau$ and $T_r\lceil_\tau$ are both 0-trees.

Suppose $p \vee q$ was derived on line $\nu' < \nu$ and $\neg p \vee r$ was derived on line $\nu'' < \nu$. Since the t-evaluation of line ν is functionally equivalent with both the t-evaluations of lines ν' and ν'', the decision trees $\varphi^{\nu'}(q)\lceil_\tau$ and $\varphi^{\nu''}(r)\lceil_\tau$ are 0-trees by Lemma 2.9.

If any branch τ' in $\varphi^{\nu'}(p)\lceil_\tau$ ends in a leaf labeled 0, then the restricted decision tree $\varphi^{\nu'}(p \vee q)\lceil_{\tau\cup\tau'}$ must be a 0-tree by property 4 of Definition 2.11 and using Corollary 2.7.

But $\varphi^{\nu'}(p \vee q)\lceil_{\tau\cup\tau'}$ cannot be a 0-tree by the inductive assumption. Hence $\varphi^{\nu'}(p)\lceil_\tau$ is a 1-tree. By repeating the above argument on line ν'' with formulas $\neg p$ and r we obtain that also $\varphi^{\nu''}(\neg p)\lceil_\tau$ is a 1-tree. This is in direct contradiction to the assumed functional equivalence of the t-evaluations on lines ν' and ν''.

This completes the case distinction. The statement follows. ∎

A.2. Section 5.

LEMMA 5.3 (see [8, Lemma 4.3]). *For large enough integer $a \in \mathbb{N}$ the following holds. If each sub-square has $(1 \pm 0.01)a$ alive centers, then there is a graphical pairing π_0.*

Proof. Consider the graph H defined on the set of nonchosen centers with an edge between any two such centers if they are in adjacent sub-squares. We want to show that there exists a graphical pairing π_0 in H. In other words, we want to partition the set of nodes of H such that each sub-graph induced by such a partition is either a single edge or a star of size 4.

Let $m = \lceil 0.26a \rceil$. The graphical pairing π_0 will have either m or $m+1$ edges between any two adjacent sub-squares. Since every node in H will have odd degree in π_0 the parity of the number of edges leaving a fixed sub-square is determined. When determining whether π_0 has m or $m+1$ edges in between two adjacent sub-squares we need to take into account the parity of the number of edges leaving each sub-square.

Consider the following (satisfiable) Tseitin formula. For every pair of adjacent sub-squares s_1, s_2 we introduce a variable $y_{\{s_1,s_2\}}$ and introduce the constraint that the four variables incident to a single sub-square have the same parity as the number of edges leaving it. Note that since the number of nonchosen centers is even (k as well as the number of chosen centers is odd) this is indeed a satisfiable Tseitin formula by Lemma 2.1. Take an assignment β satisfying said formula, and determine that there are $m + \beta_{\{s_1,s_2\}}$ many edges in π_0 between the adjacent sub-squares s_1 and s_2.

Consider any sub-square s, and let b denote the number of nonchosen centers in it. We determined that there are $4m + \sum_{e \ni s} y_e$ edges leaving the sub-square s. This determines the number of degree-3 centers in s to

$$c = \frac{4m + \sum_{e \ni s} y_e - b}{2} \quad . \tag{19}$$

Since the parity of $\sum_{e \ni s} y_e$ is equal to the parity of b, c is integer and because $b \in (1 \pm 0.01)a - 1$ it is also positive and bounded by $c \leq 0.025a + 5$.

Choose c nonchosen centers in the sub-square s to have degree 3 in the graphical pairing π_0, and connect them to nonchosen centers in adjacent sub-squares of designated degree 1. The remaining nonchosen centers can be paired up in such a manner that the number of edges between any two sub-squares is respected. This establishes the lemma. ∎

Appendix B. Switching lemma algorithms.

Algorithm B.1 A Stage of the Construction of the Extended Canonical Decision Tree $\mathcal{T}$ at τ.

Require: the sets S and I

procedure ExtendCanonicalDecisionTree($\mathcal{T}, \tau, \sigma, T_1, \ldots, T_m$)
  if no 1-branch in $T_1, \ldots, T_m$ is locally consistent with τ and σ **then**
    $\tau \leftarrow$ label 0
    return
  $\psi \leftarrow$ first 1-branch in $T_1, \ldots, T_m$ locally consistent with τ and σ
  $J \leftarrow$ possible forcing information for ψ ▷ Exists by Lemma 6.3
  $I_\pi \leftarrow$ all information of the connected components in π of centers $v \in \mathrm{supp}(J) \setminus C_\sigma$ ▷ By Definition 6.2, property 4, it holds that $I_\pi \subseteq J$
  $S_J \leftarrow$ chosen centers at distance at most 1 from $\mathrm{supp}(J) \cap C_\sigma$
  extend $\mathcal{T}$ at τ by querying all variables incident to S_J
  for all $\tau' \leftarrow$ locally consistent extension of τ in $\mathcal{T}$ **do**
    $I_{\tau' \setminus \tau} \leftarrow$ information pieces from queries along $\tau' \setminus \tau$
    $S(\tau', \sigma) \leftarrow S(\tau, \sigma) \cup S_J \cup \mathrm{supp}(J)$
    $I(\tau', \sigma) \leftarrow I(\tau, \sigma) \cup I_\pi \cup I_{\tau' \setminus \tau}$
    if $I(\tau', \sigma)$ traverses ψ **then**
      $\tau' \leftarrow$ label 1

Algorithm B.2 Recover the partial restriction ρ from ρ^* given some extra information X.

procedure Reconstruct($\rho^*, T_1, \ldots, T_m, s, t, X$)
  $j \leftarrow 1$
  $\rho_1^* \leftarrow \rho^*$
  $S_0^*, I_0^{*-}, E \leftarrow \emptyset$
  while $|S_{j-1}^*| \leq s/4$ **do**
    $\psi \leftarrow$ next 1-branch in $T_1, \ldots, T_m$ traversed by ρ_j^*
    while ψ and I_{j-1}^{*-} *not* in conflict with E **do**
      discover $\leftarrow$ next bit from X ▷ associated center to discover on ψ?
      if discover **then**
        $i \leftarrow$ next $\log(t)$ bits from X
        $v \leftarrow$ associated center of ith variable on ψ
        sign $\leftarrow$ next 9 bits from X
        $E \leftarrow E \cup \{(v, \text{sign})\}$
      else ▷ We found the forceable branch of stage j
        $\psi_j \leftarrow \psi$
        $\rho_j^*, I_j^{*-}, S_j^* \leftarrow$ RecoverForcingInformation($E, I_{j-1}^{*-}, \psi_j, \rho_{j-1}^*, S_{j-1}^*, X$)
        $E \leftarrow E$ with used signatures removed and new ones added
        $j \leftarrow j + 1$
        break
  $\rho \leftarrow \rho_j^*$ with the assignment flipped along edges in I_{j-1}^{*-}
  return ρ

Algorithm B.3 Recover the objects from a single stage given the forceable branch ψ.

procedure RecoverForcingInformation$(E, I^{*-}, \psi, \rho^*, S^*, X)$
  $E_\psi \leftarrow$ set of $(v, \text{sign}) \in E$ where v is the associated center of a variable on ψ
  $J \leftarrow \emptyset$
  for all $(v, \text{sign}_v) \in E_\psi$ **do**
    $c, d_1, \ldots, d_4, e_1, \ldots, e_4 \leftarrow \text{sign}_v$ ▷ split the signature into single bits
    for $i = 1, \ldots, 4$ **do**
      if d_i **then** ▷ there is a variable on ψ in direction i from v
        $R_i \leftarrow$ sub-square adjacent to v in direction i
        if e_i **then**
          $u_i \leftarrow$ GetPossiblyDeadCenter(R_i, E, c, X)
          $J \leftarrow J \cup \big\{\{v, u_i\}\big\}$
        else
          $J \leftarrow J \cup \{(v, i, \perp)\}$
  for all $(v, \text{sign}_v) \in E_\psi$ **do** ▷ it remains to recover connected components of π
    $c, d_1, \ldots, d_4, e_1, \ldots, e_4 \leftarrow \text{sign}_v$
    if $\neg c$ **then** ▷ v is not a chosen center
      cc $\leftarrow$ next $\log(20)$ bits from X
        ▷ encodes what kind of component v is in π
      $C \leftarrow$ centers from cc that are connected by an edge in J from v
      for all sub-squares R in which cc has a center **do**
        ▷ recover remaining centers
        if no center in C from R **then**
          $C \leftarrow C \cup$ GetPossiblyDeadCenter(R, E, c, X)
      for all edges (c_1, c_2) in cc **do** ▷ ensure that all edges are present
        $w_1, w_2 \leftarrow$ centers in C corresponding to c_1 and c_2
        $J \leftarrow J \cup \big\{\{w_1, w_2\}\big\}$
      for all nonedges $(c, \delta, \perp)$ in cc **do**
        ▷ ensure that all nonedges are present
        $w \leftarrow$ center in C corresponding to c
        $J \leftarrow J \cup \{(w, \delta, \perp)\}$
  $K \leftarrow$ RecoverK(I^{*-}, J)
  $\rho^* \leftarrow \rho^*$ with assignment flipped along edges in K
  $S \leftarrow$ RecoverExposed(I^{*-}, J, E, X)
  $S^+ \leftarrow$ RecoverNonExposed(E, S, X)
  $I \leftarrow$ read from X structure of I on the centers $S \cup S^+$
  $I^{*-} \leftarrow I^{*-} \cup I \cup$ information from J incident to nodes in $\text{supp}(I^{*-})$
  return $(\rho^*, I^{*-}, S^* \cup S)$

Algorithm B.4 Recover K from J.

procedure RecoverK(I^{*-}, J)
  $K \leftarrow J$
  for $u \in \mathrm{supp}(J)$ **do**
    if u has odd number of edges incident in J **then**
      $\delta \leftarrow$ direction in which u has no information in $J \cup I^{*-}$
      $K \leftarrow K \cup (u, \delta, \perp)$
  return K

Algorithm B.5 Recover the exposed centers at distance ≤ 1 from the chosen center in $\mathrm{supp}(J)$.

procedure RecoverExposed(I^{*-}, J, E, X)
  $S \leftarrow \mathrm{supp}(J)$
  for $u \in \mathrm{supp}(J) \cap C_\sigma$ **do**
    for δ direction **do**
      $R \leftarrow$ sub-square in direction δ of u
      if R has no chosen center in $\mathrm{supp}(I^{*-} \cup J)$ **then**
        $S \leftarrow S \cup$ GetPossiblyDeadCenter($R, E, 1, X$)
  return S

Algorithm B.6 Recover the nonexposed centers incident to the exposed chosen centers in S.

procedure RecoverNonExposed(E, S, X)
  $S^+ \leftarrow \emptyset$
  for $u \in S \cap C_\sigma$ **do**
    for δ direction **do**
      $R \leftarrow$ sub-square in direction δ of u
      if R has no chosen center in S **then**
        recover $\leftarrow$ bit from X
        if recover **then**
          $S^+ \leftarrow S^+ \cup$ GetPossiblyDeadCenter($R, E, 1, X$)
  return S^+

Algorithm B.7 Get endpoint in sub-square R potentially with the help of signatures from E.

procedure GetPossiblyDeadCenter($R, E,$ chosen, X)
  known $\leftarrow$ next bit from X ▷ is the center in R already in E or still alive?
  if known **then**
    if chosen **then**
      $u \leftarrow$ center with lowest numbered row in R that is in E or alive
    else
      $s \leftarrow$ number of signatures in E plus alive centers in R
      $i \leftarrow$ next $\log(s)$ bits from X
      $u \leftarrow i$th center in R that is in E or alive
  else
    $i \leftarrow$ next $\log(\Delta)$ bits from X
    $u \leftarrow i$th center in R
  return u

Acknowledgments. We are indebted to Susanna F. de Rezende for suggesting the change in the restriction that allows us to obtain better parameters. Furthermore, we are grateful to Mrinal Ghosh, Björn Martinsson, and Aleksa Stanković for helpful discussions on the topic of this paper.

Last but not least we would like to thank the anonymous referees for the time they took to read our manuscript. The reviews we received were incredibly detailed, contained a lot of useful suggestions, and helped us improve the presentation of this paper considerably.

REFERENCES

[1] M. Ajtai, *The complexity of the Pigeonhole Principle*, Combinatorica, 14 (1994), pp. 417–433, https://doi.org/10.1007/BF01302964; preliminary version in FOCS '88.

[2] E. Ben-Sasson, *Hard examples for the bounded depth Frege proof system*, Comput. Complexity, 11 (2002), pp. 109–136, https://doi.org/10.1007/s00037-002-0172-5.

[3] S. A. Cook and R. A. Reckhow, *The relative efficiency of propositional proof systems*, J. Symbolic Logic, 44 (1979), pp. 36–50, https://doi.org/10.2307/2273702.

[4] M. Furst, J. Saxe, and M. Sipser, *Parity, circuits and the polynomial-time hierarchy*, Math. Systems Theory, 17 (1984), pp. 13–27, https://doi.org/10.1007/BF01744431.

[5] A. Haken, *The intractability of resolution*, Theoret. Comput. Sci., 39 (1985), pp. 297–308, https://doi.org/10.1016/0304-3975(85)90144-6.

[6] J. Håstad, *Almost optimal lower bounds for small depth circuits*, in Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing, STOC '86, ACM, New York, 1986, pp. 6–20, https://doi.org/10.1145/12130.12132.

[7] J. Håstad, *On the correlation of parity and small-depth circuits*, SIAM J. Comput., 43 (2014), pp. 1699–1708, https://doi.org/10.1137/120897432.

[8] J. Håstad, *On small-depth Frege proofs for Tseitin for grids*, J. ACM, 68 (2021), 1, https://doi.org/10.1145/3425606.

[9] J. Håstad and K. Risse, *On bounded depth proofs for Tseitin formulas on the grid; revisited*, in 2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS), 2022, pp. 1138–1149.

[10] R. Impagliazzo, W. Matthews, and R. Paturi, *A satisfiability algorithm for* AC^0, in Proceedings of the Twenty-Third Annual ACM-SIAM Symposium on Discrete Algorithms, ACM, New York, SIAM, Philadelphia, 2012, pp. 961–972, https://doi.org/10.1137/1.9781611973099.77.

[11] J. Krajíček, P. Pudlák, and A. Woods, *An exponential lower bound to the size of bounded depth Frege proofs of the Pigeonhole Principle*, Random Structures Algorithms, 7 (1995), pp. 15–39, https://doi.org/10.1002/rsa.3240070103.

[12] T. Pitassi, P. Beame, and R. Impagliazzo, *Exponential lower bounds for the pigeonhole principle*, Comput. Complexity, 3 (1993), pp. 97–140, https://doi.org/10.1007/BF01200117; preliminary version in *STOC* '92.

[13] T. Pitassi, P. Ramakrishnan, and L. Tan, *Tradeoffs for small-depth Frege proofs*, in 2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society, Los Alamitos, CA, 2022, pp. 445–456, https://doi.org/10.1109/FOCS52979.2021.00052.

[14] T. Pitassi, B. Rossman, R. A. Servedio, and L.-Y. Tan, *Poly-logarithmic Frege depth lower bounds via an expander switching lemma*, in Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing, STOC '16, Association for Computing Machinery, 2016, pp. 644–657, https://doi.org/10.1145/2897518.

[15] A. Razborov, *Bounded-depth formulae over the basis {AND, XOR} and some combinatorial problems,* (in Russian), in Problems of Cybernetics. Complexity Theory and Applied Mathematical Logic, VINITI, Moscow, 1988, pp. 149–166.

[16] A. A. Razborov, *Bounded arithmetic and lower bounds in Boolean complexity*, in Feasible Mathematics, II (Ithaca, NY, 1992), P. Clote and J. Remmel, eds., Birkhäuser Boston, Boston, 1995, pp. 344–386.

[17] B. Rossman, R. A. Servedio, and L. Tan, *An average-case depth hierarchy theorem for Boolean circuits*, in IEEE 56th Annual Symposium on Foundations of Computer Science, FOCS 2015, Berkeley, CA, 2015, pp. 1030–1048.

[18] M. Sipser, *Borel sets and circuit complexity*, in Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing, STOC '83, ACM, 1983, pp. 61–69, https://doi.org/10.1145/800061.808733.

[19] R. Smolensky, *Algebraic methods in the theory of lower bounds for Boolean circuit complexity*, in Proceedings of the Nineteenth Annual ACM Symposium on Theory of Computing, STOC '87, ACM, 1987, pp. 77–82, https://doi.org/10.1145/28395.28404.

[20] G. S. Tseitin, *On the complexity of derivation in the propositional calculus*, in Studies in Constructive Mathematics and Mathematical Logic, Part II, A. O. Slisenko, ed., Springer, Berlin, Heidelberg, 1968, https://doi.org/10.1007/978-3-642-81955-1_28.

[21] A. Urquhart and X. Fu, *Simplified lower bounds for propositional proofs*, Notre Dame J. Formal Logic, 37 (1996), pp. 523–544, https://doi.org/10.1305/ndjfl/1040046140.

[22] A. C.-C. Yao, *Separating the polynomial-time hierarchy by oracles*, in 26th Annual Symposium on Foundations of Computer Science (sfcs 1985), IEEE, 1985, pp. 1–10, https://doi.org/10.1109/SFCS.1985.49.