Solving MAX-r-SAT above a Tight Lower Bound*

Noga Alon[1], Gregory Gutin[2], Eun Jung Kim[2], Stefan Szeider[3], and Anders Yeo[2]

[1] Schools of Mathematics and Computer Science
Tel Aviv University
Tel Aviv 69978, Israel
nogaa@post.tau.ac.il

[2] Department of Computer Science
Royal Holloway, University of London
Egham, Surrey TW20 0EX, UK
{gutin|eunjung|anders}@cs.rhul.ac.uk

[3] Institute of Information Systems, Vienna University of Technology
A-1040 Vienna, Austria
stefan@szeider.net

January 27, 2010

Abstract

We present an exact algorithm that decides, for every fixed $r \geq 2$ in time $O(m) + 2^{O(k^2)}$ whether a given multiset of m clauses of size r admits a truth assignment that satisfies at least $((2^r - 1)m + k)/2^r$ clauses. Thus MAX-r-SAT is fixed-parameter tractable when parameterized by the number of satisfied clauses above the tight lower bound $(1 - 2^{-r})m$. This solves an open problem of Mahajan, Raman and Sikdar (J. Comput. System Sci., 75, 2009).

Our algorithm is based on a polynomial-time data reduction procedure that reduces a problem instance to an equivalent algebraically represented problem with $O(k^2)$ variables. This is done by representing the instance as an appropriate polynomial, and by applying a probabilistic argument combined with some simple tools from Harmonic analysis to show that if the polynomial cannot be reduced to one of size $O(k^2)$, then there is a truth assignment satisfying the required number of clauses.

We introduce a new notion of bikernelization from a parameterized problem to another one and apply it to prove that the above-mentioned parameterized MAX-r-SAT admits a polynomial-size kernel.

Combining another probabilistic argument with tools from graph matching theory and signed graphs, we show that if an instance of MAX-2-SAT with m clauses has at least $3k$ variables after application of certain polynomial time reduction rules to it, then there is a truth assignment that satisfies at least $(3m + k)/4$ clauses.

We also outline how the fixed-parameter tractability and polynomial-size kernel results on MAX-r-SAT can be extended to more general families of Boolean Constraint Satisfaction Problems.

*A preliminary version of this paper is to appear in the proceedings of ACM-SIAM Symposium on Discrete Algorithms (SODA 2010).

1 Introduction

The Maximum r-Satisfiability Problem (MAX-r-SAT) is a classic optimization problem with a wide range of real-world applications. The task is to find a truth assignment to a multiset of clauses, each with exactly r literals, that satisfies as many clauses as possible, or in the decision version of the problem, to satisfy at least t clauses where t is given with the input. Even MAX-2-SAT is NP-hard [10] and hard to approximate [15], in strong contrast with 2-SAT which is solvable in linear time [3].

It is always possible to satisfy a $1-2^{-r}$ fraction of a given multiset of clauses with exactly r literals each; a truth assignment that meets this lower bound can be found in polynomial time by Johnson's algorithm [19]. This lower bound is *tight* in the sense that it is optimal for an infinite sequence of instances. In this paper we show that for every fixed r we can decide in time $O(m)+2^{O(k^2)}$ whether a given multiset of m clauses admits a truth assignment that satisfies at least $((2^r-1)m+k)/2^r$ clauses. Thus, MAX-r-SAT is fixed-parameter tractable when parameterized by the number of satisfied clauses above the tight lower bound; this answers a question posed by Mahajan, Raman and Sikdar [21].

Our algorithm described in Section 4 is based on a polynomial-time data reduction procedure that reduces a problem instance to an equivalent algebraically represented problem with $O(k^2)$ variables. This is done by representing the instance as an appropriate polynomial, and by applying a probabilistic argument combined with some simple tools from Harmonic analysis to show that if the polynomial cannot be reduced to one of size $O(k^2)$, then there is a truth assignment satisfying the required number of clauses. The basic approach is based on the ideas of [1], and a similar one which, however, does not apply any algebraic reductions, was used in [11, 12] to show the existence of quadratic kernels for other problems parameterized above tight lower bounds.

We also show that the above-mentioned parameterized MAX-r-SAT admits a polynomial-size kernel. This can be deduced from our fixed-parameter result and a general lemma proved in Section 3, and can also be proved by a more efficient, direct argument. The lemma, which is interesting in its own right, links a new concept that we call bikernelization with the well-known concept of kernelization. We believe that bikernelization, in general, and the lemma, in particular, will have further applications.

In Section 5, combining another probabilistic argument with tools from graph matching theory and signed graphs, we show that if an instance $\mathcal{I}$ of MAX-2-SAT on m clauses has at least $3k$ variables after application of certain polynomial time reduction rules to it, then there is a truth assignment for $\mathcal{I}$ that satisfies at least $(3m+k)/4$ clauses. Thus, MAX-2-SAT admits a problem kernel with at most $3k-1$ variables.

Section 6 is devoted to discussions. In particular, we outline how the fixed-parameter tractability and polynomial-size kernel results on MAX-r-SAT can be extended to more general families of Boolean Constraint Satisfaction Problems.

Related Work Parameterizations *above a guaranteed value* were first considered by Mahajan and Raman [20] for the problems MAX-SAT and MAX-CUT. They devised an algorithm for MAX-SAT with running time $O^*(1.618^k+\sum_{i=1}^{m}|C_i|)$ that finds, for a

multiset $\{C_1, \ldots, C_m\}$ of m clauses, a truth assignment satisfying at least $\lceil m/2 \rceil + k$ clauses, or decides that no such truth assignment exists ($|C_i|$ denotes the number of literals in C_i). In a recent paper [21] Mahajan, Raman and Sikdar argued that a natural (and challenging) parameter for a maximization problem is the number of clauses satisfied above a *tight* lower bound, which is $(1-2^{-r})m$ for MAX-SAT if each clause contains exactly r different variables. Only a few non-trivial results are known for problems parameterized above a tight lower bound [12, 13, 14, 17, 20].

Mahajan et al. [21] state several problems parameterized above a tight lower bound whose parameterized complexity is open. One of the problems is the (exact) MAX-r-SAT problem (an instance consists of m clauses, each containing exactly r different literals) parameterized by the number of satisfied clauses above the tight lower bound $(1-2^{-r})m$. Our main result answers this question.

2 Preliminaries

We assume an infinite supply of propositional *variables*. A *literal* is a variable x or its negation $\overline{x}$. A *clause* is a finite set of literals not containing a complementary pair x and $\overline{x}$. A clause is of *size* r if it contains exactly r literals. For simplicity of presentation, we will denote a clause by a sequence of its literals. For example, the clause $\{\overline{x}, y\}$ will be denoted $\overline{x}y$ or equivalently $y\overline{x}$. A *CNF formula* F is a finite multiset of clauses (a clause may appear in the multiset several times). A variable x *occurs* in a clause if the clause contains x or $\overline{x}$, and x occurs in a CNF formula F if it occurs in some clause of F. Let $\mathrm{var}(C)$ and $\mathrm{var}(F)$ denote the sets of variables occurring in C and F, respectively. A CNF formula is an r-CNF formula if $|C| = r$ for all $C \in F$. Thus we require that each clause of a r-CNF formula contains *exactly* r different literals (some authors use for that the term *exact* r-CNF). A *truth assignment* is a mapping $\tau : V \to \{-1, 1\}$ defined on some set V of variables. We write 2^V to denote the set of all truth assignments on V. A truth assignment τ *satisfies* a clause C if there is some variable $x \in C$ with $\tau(x) = 1$ or a negated variable $\overline{x} \in C$ with $\tau(x) = -1$. We write $\mathrm{sat}(\tau, F)$ for the number of clauses of F that are satisfied by τ, and we write

$$\mathrm{sat}(F) = \max_{\tau \in 2^{\mathrm{var}(F)}} \mathrm{sat}(\tau, F).$$

A *parameterized problem* is a subset $L \subseteq \Sigma^* \times \mathbb{N}$ over a finite alphabet Σ. L is *fixed-parameter tractable* if the membership of an instance (x, k) in $\Sigma^* \times \mathbb{N}$ can be decided in time $|x|^{O(1)} \cdot f(k)$ where f is a computable function of the parameter [8, 9, 22]. Given a parameterized problem L, a *kernelization of* L is a polynomial-time algorithm that maps an instance (x, k) to an instance (x', k') (the *kernel*) such that (i) $(x, k) \in L$ if and only if $(x', k') \in L$, (ii) $k' \leq f(k)$, and (iii) $|x'| \leq g(k)$ for some functions f and g. The function $g(k)$ is called the *size* of the kernel. A parameterized problem is fixed-parameter tractable if and only if it is decidable and admits a kernelization [9].

We shall consider the following parameterized version of MAX-r-SAT.

MAX-r-SAT ABOVE TIGHT LOWER BOUND (or MAX-r-SAT$_{\text{TLB}}$ for short)

Instance: A pair (F, k) where F is a multiset of m clauses of size r and k is a nonnegative integer.

Parameter: The integer k.

Question: Is $\text{sat}(F) \geq ((2^r - 1)m + k)/2^r$?

We note that Mahajan et al. [21] use a slightly different formulation of the problem, asking for an assignment that satisfies at least $(1-2^{-r})m+k$ clauses; since r is fixed, this change does not affect the complexity of the problem.

We will also refer to the following special case of another problem introduced by Mahajan et al. [21].

MAX r-LIN2 ABOVE TIGHT LOWER BOUND (or MAX-r-LIN2$_{\text{TLB}}$ for short)

Instance: A system of m linear equations $e_1, \ldots, e_m$ in n variables over $\mathbb{F}_2$, where no equation has more than r variables, and each equation e_j has a positive integral weight w_j, and a nonnegative integer k.

Parameter: The integer k.

Question: Is there an assignment of values to the n variables such that the total weight of the satisfied equations is at least $(W + k)/2$, where $W = w_1 + \cdots + w_m$?

Note that trivially $W/2$ is indeed a tight lower bound for the above problem, as the expected number of satisfied equations in a random assignment is $W/2$, and if the equations come in identical pairs with contradicting free terms, no assignment satisfies more equations. It was proved in [12] that MAX-r-LIN2$_{\text{TLB}}$ admits a kernel with $O(k^2)$ equations and variables.

3 Bikernelization

In this section we introduce a new notion of a bikernelization and study its basic properties. A bikernelization from L to L' is of interest especially when L' is a well-studied problem.

Given a pair L, L' of parameterized problems, a *bikernelization from L to L'* is a polynomial-time algorithm that maps an instance (x, k) to an instance (x', k') (the *bikernel*) such that (i) $(x, k) \in L$ if and only if $(x', k') \in L'$, (ii) $k' \leq f(k)$, and (iii) $|x'| \leq g(k)$ for some functions f and g. The function $g(k)$ is called the *size* of the bikernel. Observe that a *kernelization* of a parameterized problem L is simply a bikernelization from L to itself, i.e., a bikerenelization generalizes a kernelization.

Recall that a parameterized problem is fixed-parameter tractable if and only if it is decidable and admits a kernelization. This result can be extended as follows: A parameterized problem L is fixed-parameter tractable if and only if it is decidable and admits a bikernelization from itself to a parameterized problem L'. Indeed, if L is fixed-parameter tractable, then L is decidable and admits a bikernelization to itself. If L is decidable and admits a bikernelization from itself to a parameterized problem

L', then (x,k) can be decided by first mapping it to (x',k') in polynomial time and then deciding (x',k') in time depending only on k', and thus only on k.

We are especially interested in cases when kernels are of polynomial size. The next lemma is similar to Theorem 3 in [5].

Lemma 1. *Let L, L' be a pair of parameterized problems such that L' is in NP and L is NP-complete. If there is a bikernelization from L to L' producing a bikernel of polynomial size, then L has a polynomial-size kernel.*

Proof. Consider a bikernelization from L to L' that maps an instance $(x,k) \in L$ to an instance $(x',k') \in L'$ with $k' \leq f(k)$. Since L' is in NP and L is NP-complete, there exists a polynomial time reduction from L' to L. Thus, we can find in polynomial time an instance (x'',k'') of L which is decision-equivalent with (x',k'), and in turn with (x,k). Observe that $|x''| \leq |x'|^{O(1)} \leq k^{O(1)}$ and $k'' \leq (k')^{O(1)} + (|x'|)^{O(1)} \leq f(k)^{O(1)} + k^{O(1)}$. Thus, (x'',k'') is a kernel of L of polynomial size. □

4 MAX-r-SAT

4.1 An Algebraic Representation

Let F be an r-CNF formula with clauses $C_1, \ldots, C_m$ in the variables $x_1, x_2, \ldots, x_n$.

For F, consider

$$X = \sum_{C \in F} [1 - \prod_{x_i \in \mathrm{var}(C)} (1 + \varepsilon_i x_i)],$$

where $\varepsilon_i \in \{-1, 1\}$ and $\varepsilon_i = -1$ if and only if x_i is in C.

Lemma 2. *For a truth assignment τ, we have $X = 2^r(\mathrm{sat}(\tau, F) - (1 - 2^{-r})m)$.*

Proof. Observe that $\prod_{x_i \in \mathrm{var}(C)}(1 + \varepsilon_i x_i)$ equals 2^r if C is falsified and 0, otherwise. Thus, $X = m - 2^r(m - \mathrm{sat}(\tau, F))$ implying the claimed formula. □

After algebraic simplification $X = X(x_1, x_2, \ldots, x_n)$ can be written as $X = \sum_{I \in \mathcal{S}} X_I$, where $X_I = c_I \prod_{i \in I} x_i$, each c_I is a nonzero integer and $\mathcal{S}$ is a family of nonempty subsets of $\{1, \ldots, n\}$ each with at most r elements.

The question we address is that of deciding whether or not there are values $x_i \in \{-1, 1\}$ so that $X = X(x_1, x_2, \ldots, x_n) \geq k$. The idea is to use a probabilistic argument and show that if the above polynomial has many nonzero coefficients, that is, if $|\mathcal{S}|$ is large, this is necessarily the case, whereas if it is small, the question can be solved by checking all possibilities of the relevant variables.

4.2 The Properties of X

In what follows, we assume that each variable x_i takes its values randomly and independently in $\{-1, 1\}$ and thus X is a random variable. Our approach is similar to the one in [1]. For completeness, we reproduce part of the argument (modifying it a bit and slightly improving the constant for the case considered here). We need the simple lemma.

Lemma 3 (see, e.g., [1], Lemma 3.1)**.** *For every real random variable X with finite and positive forth moment,*

$$\mathbb{E}(|X|) \geq \frac{\mathbb{E}(X^2)^{3/2}}{\mathbb{E}(X^4)^{1/2}}.$$

The above lemma implies the following (see [1], Lemma 3.2, part (ii) for a similar result).

Corollary 1. *Let X be a real random variable and suppose that its first, second and forth moments satisfy $\mathbb{E}(X) = 0$, $\mathbb{E}(X^2) = \sigma^2 > 0$ and $\mathbb{E}(X^4) \leq b\sigma^4$. Then*

$$\mathbb{P}(X \geq \frac{\sigma}{2\sqrt{b}}) > 0.$$

Proof. By Lemma 3, $\mathbb{E}(|X|) \geq \frac{\sigma}{\sqrt{b}}$. Since $\mathbb{E}(X) = 0$ it follows that

$$\mathbb{P}(X > 0)\mathbb{E}(X|X > 0) \geq \frac{\sigma}{2\sqrt{b}}. \tag{1}$$

Therefore, X must be at least $\sigma/(2\sqrt{b})$ with positive probability. □

We also use the following lemma of Bourgain [7].

Lemma 4 ([7])**.** *Let $f = f(x_1, \ldots, x_n)$ be a polynomial of degree r in n variables $x_1, \ldots, x_n$ with domain $\{-1, 1\}$. Define a random variable X by choosing a vector $(\varepsilon_1, \ldots, \varepsilon_n) \in \{-1, 1\}^n$ uniformly at random and setting $X = f(\varepsilon_1, \ldots, \varepsilon_n)$. Then, $\mathbb{E}(X^4) \leq 2^{6r}(\mathbb{E}(X^2))^2$.*

Returning to the random variable $X = X(x_1, x_2, \ldots, x_n)$ defined in the previous subsection, we prove the following.

Lemma 5. *Let $X = \sum_{I \in \mathcal{S}} X_I$, where $X_I = c_I \prod_{i \in I} x_i$ is as in the previous subsection, and assume it is not identically zero. Then $\mathbb{E}(X) = 0$, $\mathbb{E}(X^2) = \sum_{I \in \mathcal{S}} c_I^2 \geq |\mathcal{S}| > 0$ and $\mathbb{E}(X^4) \leq 2^{6r}\mathbb{E}(X^2)^2$.*

Proof. Since the x_i's are mutually independent, $\mathbb{E}(X) = 0$. Note that for $I, J \in \mathcal{S}$, $I \neq J$, we have $\mathbb{E}(X_I X_J) = c_I c_J \mathbb{E}(\prod_{i \in I \Delta J} x_i) = 0$, where $I \Delta J$ is the symmetric difference of I and J. Thus, $\mathbb{E}(X^2) = \sum_{I \in \mathcal{S}} c_I^2$. By Lemma 4, $\mathbb{E}(X^4) \leq 2^{6r}\mathbb{E}(X^2)^2$. □

4.3 The Main Result for General r

Theorem 1. *The problem* Max-r-Sat$_{\text{TLB}}$ *is fixed-parameter tractable and can be solved in time $O(m) + 2^{O(k^2)}$. Moreover, there exist (i) a polynomial-size bikernel from* Max-r-Sat$_{\text{TLB}}$ *to* Max-r-Lin2$_{\text{TLB}}$*, and (ii) a polynomial-size kernel of* Max-r-Sat$_{\text{TLB}}$*. In fact, there are such a bikernel and a kernel of size $O(k^2)$.*

Proof. By Lemma 2 our problem is equivalent to that of deciding whether or not there is a truth assignment to the variables $x_1, x_2, \ldots, x_n$, so that

$$X(x_1, \ldots, x_n) \geq k. \tag{2}$$

Note that in particular this implies that if X is the zero polynomial, then any truth assignment satisfies exactly a $(1-2^{-r})$ fraction of the original clauses. By Corollary 1 and Lemma 5, $\mathbb{P}(X \geq \frac{\sqrt{\mathbb{E}(X^2)}}{2\sqrt{b}}) > 0$, where $b = 2^{6r}$ and $\mathbb{E}(X^2) = \sum_{I \in \mathcal{S}} c_I^2 \geq |\mathcal{S}|$; the last inequality follows from the fact that each $|c_I|$ is a positive integer. Therefore $\mathbb{P}(X \geq \frac{\sqrt{|\mathcal{S}|}}{2 \cdot 8^r}) > 0$. Now, if $k \leq \frac{\sqrt{|\mathcal{S}|}}{2 \cdot 8^r}$ then there are $x_i \in \{-1, 1\}$ such that (2) holds, and there is an assignment for which the answer to MAX-r-SAT$_{\text{TLB}}$ is YES. Otherwise, $|\mathcal{S}| = O(k^2)$, and in fact even $\sum_{I \in \mathcal{S}} |c_I| \leq \sum_{I \in \mathcal{S}} c_I^2 = O(k^2)$, that is, the total number of terms of the simplified polynomial, even when counted with multiplicities, is at most $O(k^2)$.

For any fixed r, the representation of a problem instance of m clauses as a polynomial, and the simplification of this polynomial, can be performed in time $O(m)$. If the number of nonzero terms of this polynomial is larger than $4 \cdot 8^{2r} k^2$, then the answer to the problem is YES. Otherwise, the polynomial has at most $O(k^2)$ terms and depends on at most $O(k^2)$ variables, and its maximum can be found in time $2^{O(k^2)}$.

This completes the proof of the first part of the theorem. We next establish the second part. Given the simplified polynomial X as above, define a problem in MAX-r-LIN2$_{\text{TLB}}$ with the variables $z_1, z_2, \ldots, z_n$ as follows. For each nonzero term $c_I \prod_{i \in I} x_i$ consider the linear equation $\sum_{i \in I} z_i = b$, where $b = 0$ if c_I is positive, and $b = 1$ if c_I is negative, and either associate this equation with the weight $w_I = |c_I|$, or duplicate it $|c_I|$ times. It is easy to check that this system of equations has an assignment z_i satisfying at least $[\sum_{I \in \mathcal{S}} w_I + k]/2$ of the equations if and only if there are $x_i \in \{-1, 1\}$ so that $X(x_1, x_2, \ldots, x_n) \geq k$. This is shown by the transformation $x_i = (-1)^{z_i}$. See also [16] and [12] for a similar discussion. Since, as explained above, we may assume that $\sum_{I \in \mathcal{S}} |c_I| = O(k^2)$ (as otherwise we know that the answer to our problem is YES), this provides the required bikernel of size $O(k^2)$ to MAX-r-LIN2$_{\text{TLB}}$.

It remains to prove the existence of a polynomial size kernel for the original problem. One way to do that is to apply Lemma 1. Indeed, MAX-r-LIN2$_{\text{TLB}}$ is in NP, and MAX-r-SAT$_{\text{TLB}}$ is NP-complete, implying the desired result.

It is also possible to give a direct proof, which shows that the problem admits a kernel of size at most $O(k^2)$. To do so, we replace each linear equation of at most r variables by a set of 2^{r-1} clauses, so that if the variables z_i satisfy the equation, the same Boolean variables $x_i = z_i$ satisfy all these clauses, and if the variables z_i do not satisfy the equation, then the variables x_i above satisfy only $2^{r-1} - 1$ of the clauses. This is done as follows. Consider, first, a linear equation with exactly r variables. After renumbering the variables, if needed, a typical equation is of the form $z_1 + z_2 + \cdots + z_r = b$, where the sum is over $\mathbb{F}_2$ and $b \in \{0, 1\}$. There are exactly 2^{r-1} Boolean assignments $\delta = (\delta_1, \delta_2, \ldots, \delta_r)$ for the variables z_i that do **not** satisfy the equation. For each such assignment δ let C_δ be the clause consisting of r literals, where the literal number i is x_i if $\delta_i = 0$ and is $\overline{x_i}$ if $\delta_i = 1$. Note that if the variables $z_1, z_2, \ldots, z_r$ satisfy the above equation, then $(z_1, z_2, \ldots, z_r)$ is not one of the vectors δ considered, and hence each of the clauses C_δ constructed contains at least one satisfied literal when $x_i = z_i$. Therefore, in this case all clauses are satisfied. A similar argument shows that if the variables z_i do not satisfy the equation, there will be exactly one non-satisfied clause, namely the one corresponding to the vector $\delta = (z_1, z_2, \ldots, z_r)$. The construction can be extended to equations with less than r

variables. Indeed, the only property used in the transformation above is that there are exactly 2^{r-1} Boolean assignments for the variables $z_1, z_2, \ldots, z_r$ that do not satisfy the equation. If the equation has only $(1 \leq)\ s < r$ variables, add to these variables an arbitrary set of $r - s$ of the other variables, and consider the set of all Boolean assignments to this augmented set of variables that do not satisfy the equation. Here, too, there are exactly 2^{r-1} such assignments and we can thus repeat the construction above in this case as well.

The above procedure transforms a set of W linear equations over $\mathbb{F}_2$ into a multiset of $2^{r-1}W$ clauses. Moreover, if some truth assignment does not satisfy exactly ℓ equations, then the same assignment does not satisfy the same number, ℓ, of clauses. In particular, there is an assignment satisfying all equations but $(W-k)/2$ of them, if and only if there is an assignment satisfying all clauses but $(W-k)/2$ of them. This means that among the $m = 2^{r-1}W$ clauses, the number of satisfied ones is $m-(W-k)/2 = [(2^r-1)m+2^{r-1}k]/2^r$. This reduces an instance of Max-r-Lin2$_{\text{TLB}}$ with W equations and parameter k to an instance of Max-r-Sat$_{\text{TLB}}$ with $2^{r-1}W$ clauses and parameter $2^{r-1}k$. Since r is a constant, this provides the required kernel of size $O(k^2)$, completing the proof. □

5 Max-2-Sat

In this section we describe an alternative, more combinatorial, approach to the problem for $r = 2$. Although this approach is somewhat more complicated than the one discussed in the previous section, it provides an additional insight to this special case of the problem, and allows us to obtain a linear kernel for Max-2-Sat$_{\text{TLB}}$.

We start with a simple reduction rule that applies to any value of r.

5.1 The Semicomplete Reduction

We say that a pair of distinct clauses Y and Z has a *conflict* if there is a literal $p \in Y$ such that $\overline{p} \in Z$. We say that an r-CNF formula F is *semicomplete* if the number of clauses is $m = 2^r$ and every pair of distinct clauses of F has a conflict. A semicomplete r-CNF formula is *complete* if each clause is over the same set of variables. There are r-CNF formulas that are semicomplete but not complete; consider for example $\{xy,\ x\overline{y},\ \overline{x}z,\ \overline{xz}\}$. We have the following:

Lemma 6. *Every truth assignment to a semicomplete r-CNF formula satisfies exactly $2^r - 1$ clauses.*

Proof. Let S be a semicomplete r-CNF formula. To prove that no truth assignment satisfies all clauses of S we use the following simple counting argument from [18]. Observe that every clause is not satisfied by exactly 2^{n-r} truth assignments. However, each of these assignments satisfies each other clause (due to the conflicts). So, we have exactly $2^r \cdot 2^{n-r}$ truth assignments not satisfying F. But $2^r \cdot 2^{n-r} = 2^n$, the total number of truth assignments.

Now let τ be a truth assignment of S. By the above, τ does not satisfy a clause C of F. However, τ satisfies any other clause of S as any other clause has a conflict with C. □

Consider the following data reduction procedure.

Given an r-CNF formula F that contains a semicomplete subset $F' \subseteq F$, delete F' from F and consider $F \setminus F'$ instead. Let F^S denote the formula obtained from F by applying this deletion process as long as possible. We say that F^S is obtained from F by *semicomplete reduction*.

We state the following two simple observations as a lemma.

Lemma 7. *Let F be an r-CNF formula.*

1. *F^S can be obtained from F in polynomial time.*
2. $\mathrm{sat}(F) - \mathrm{sat}(F^S) = (1 - 2^{-r})(|F| - |F^S|)$.

5.2 Kernelization

Let F be a 2-CNF formula. A variable $x \in \mathrm{var}(F)$ is *insignificant* if for each literal y the numbers of occurrences of the two clauses xy and $\overline{x}y$ in F are the same. A variable $x \in \mathrm{var}(F)$ is *significant* if it is not insignificant. A literal is significant or insignificant if its underlying variable is significant or insignificant, respectively.

Theorem 2. *Let F be a 2-CNF formula with $F = F^S$ (i.e., F contains no semicomplete subsets) and let $k \geq 0$ be an integer. If F has more than $3k - 2$ significant variables, then* $\mathrm{sat}(F) \geq (3|F| + k)/4$.

The remainder of this section is devoted to the proof of Theorem 2 and its corollaries. Let F be a 2-CNF formula with m clauses and n variables and let k be an integer. We assume that F contains no semicomplete subsets, i.e., $F = F^S$.

For a literal x let $c(x)$ denote the number of clauses in F containing x. Given a pair of literals x and y, $x \neq \overline{y}$, let $c(xy)$ be the number of occurrences of clause xy in F.

Given a clause $C \in F$ and a variable $x \in \mathrm{var}(F)$, let $\delta_C(x)$ be an indicator variable whose value is set as $\delta_C(x) = 1$ if $x \in C$, $\delta_C(x) = -1$ if $\overline{x} \in C$, and $\delta_C(x) = 0$ otherwise.

Lemma 8. *For each subset $R = \{x_1, \ldots, x_q\} \subseteq \mathrm{var}(F)$ we have* $\mathrm{sat}(F) \geq (3m + k_R)/4$ *for*

$$k_R = \sum_{1 \leq i \leq q} (c(x_i) - c(\overline{x_i})) + \sum_{1 \leq i < j \leq q} \big(c(x_i\overline{x_j}) + c(\overline{x_i}x_j) - c(x_ix_j) - c(\overline{x_ix_j})\big).$$

Proof. Take a random truth assignment $\tau \in 2^{\mathrm{var}(F)}$ such that $\tau(x_i) = 1$ for all $i \in \{1, \ldots, q\}$ and $\mathbb{P}(\tau(x) = 1) = 0.5$ for all $x \in \mathrm{var}(F) \setminus R$. A simple case analysis yields that the probability that a clause $C \in F$ is satisfied by τ is given by

$$\mathbb{P}(\tau \text{ satisfies } C) = 1 - \frac{1}{4} \prod_{1 \leq i \leq q} (1 - \delta_C(x_i)).$$

Observe that for any clause C and any three distinct variables x, y, z we have $\delta_C(x)\delta_C(y)\delta_C(z) = 0$ as $\mathrm{var}(C)$ contains exactly two variables. Hence we can deter-

mine the expected number of clauses satisfied by τ as follows.

$$\begin{aligned}
\mathbb{E}(\mathrm{sat}(\tau, F)) &= \sum_{C \in F} \mathbb{P}[\ \tau \text{ satisfies } C\ ] \\
&= \sum_{C \in F} \{1 - \frac{1}{4} \prod_{1 \le i \le q} (1 - \delta_C(x_i))\} \\
&= \frac{3}{4}m + \frac{1}{4} \sum_{C \in F} \{ \sum_{1 \le i \le q} \delta_C(x_i) - \sum_{1 \le i < j \le q} \delta_C(x_i)\delta_C(x_j)\} \\
&= \frac{3}{4}m + \frac{1}{4} \{ \sum_{1 \le i \le q} \sum_{C \in F} \delta_C(x_i) - \sum_{1 \le i < j \le q} \sum_{C \in F} \delta_C(x_i)\delta_C(x_j)\} \\
&= \frac{3}{4}m + \frac{1}{4}k_R. \qquad \square
\end{aligned}$$

We construct an *auxiliary graph* $G = (V, E)$ from F by letting $V = \mathrm{var}(F)$ and $xy \in E$ if and only if there exists a clause $C \in F$ with $\mathrm{var}(C) = \{x, y\}$ (equivalently, $c(x\overline{y}) + c(\overline{x}y) + c(xy) + c(\overline{xy}) \ge 1$).

We assign a *weight* to each vertex x and edge xy of $G = (V, E)$:

$$\begin{aligned}
w(x) &:= \sum_{C \in F} \delta_C(x) = c(x) - c(\overline{x}), \\
w(xy) &:= -\sum_{C \in F} \delta_C(x)\delta_C(y) = c(x\overline{y}) + c(\overline{x}y) - c(xy) - c(\overline{xy}).
\end{aligned}$$

For subsets $U \subseteq V$ and $H \subseteq E$, let $w(U) = \sum_{x \in U} w(x)$ and $w(H) = \sum_{xy \in H} w(xy)$. The *weight* $w(Q)$ of a subgraph $Q = (U, H)$ is $w(U) + w(H)$. Let G^0 be the graph obtained from G by removing all edges of weight zero.

Lemma 9. *A variable $x \in \mathrm{var}(F)$ is insignificant if and only if x is an isolated vertex in G^0 and $w(x) = 0$.*

Proof. Suppose $x \in \mathrm{var}(F)$ is insignificant. Choose an edge $xy \in E$ (this is possible since by construction G has no isolated vertices). Since x is insignificant, $c(x\overline{y}) = c(\overline{xy})$ and $c(xy) = c(\overline{x}y)$ and thus $w(xy) = 0$. Therefore the edge xy does not appear in G^0 and x is isolated in G^0. Observe that we have $c(x) = c(\overline{x})$, which implies $w(x) = 0$.

Suppose $x \in \mathrm{var}(F)$ is an isolated vertex of G^0 and $w(x) = 0$. Since G has no isolated vertices, we have $w(xy) = 0$ for all $xy \in E$. In order to derive a contradiction, let us suppose x is a significant variable of F. Consequently there is (i) either a clause $xy \in F$ such that $c(xy) > c(\overline{x}y)$, or (ii) there is a clause $\overline{x}y \in F$ such that $c(\overline{x}y) > c(xy)$. We consider case (i) only, case (ii) can be treated analogously. With $w(xy) = 0$, we have $c(x\overline{y}) > c(\overline{xy})$, and thus $x\overline{y} \in F$.

Now the condition $w(x) = c(x) - c(\overline{x}) = 0$ implies the existence of an edge $xz \in E$ with $z \ne y$ such that for some $z' \in \{z, \overline{z}\}$ we have $\overline{x}z' \in F$ and $c(\overline{x}z') > c(xz')$. Without loss of generality, assume that $z' = z$. Since $w(xz) = 0$, we have $\overline{xz} \in F$. However, the four clauses xy, $x\overline{y}$, $\overline{x}z$, $\overline{xz}$ in F form a semicomplete 2-CNF formula, which contradicts our assumption that $F = F^S$. Hence x is indeed an insignificant variable. $\square$

For a set $X \subseteq \mathrm{var}(F)$ we let F_X denote the 2-CNF formula obtained from F by replacing x with $\overline{x}$ and $\overline{x}$ with x for each $x \in X$. We say that F_X is obtained from F by *switching* X.

The following lemma follows immediately from the definitions of switch and weights.

Lemma 10. *The auxiliary graph G_X corresponding to F_X can be obtained from $G = (V, E)$ by reversing the signs of the weights of all vertices in X and all edges between X and $V \setminus X$. Moreover,* $\mathrm{sat}(F) = \mathrm{sat}(F_X)$.

To distinguish between weights in G and G_X, we use $w_X(.)$ for weights of G_X. Similarly, we use $c_X(.)$ for F_X.

It is sometimes convenient to stress that the set X we are switching induces a subgraph. We can *switch an induced graph* Q by switching all the vertices of Q. Observe that by switching an induced graph Q, we reverse the signs of weights on all vertices of Q and all edges incident with exactly one vertex of Q, but the sign of each edge within Q remains unchanged. This property will play a major role to show that a certain structure meets the condition of the following lemma.

Lemma 11. *If there exist a set $X \subset V(G^0)$ and an induced subgraph $Q = (U, H)$ of G^0 with $w_X(Q) \geq k$, then* $\mathrm{sat}(F) \geq (3m + k)/4$.

Proof. We consider $U = \{x_1, \ldots, x_q\}$ as a subset of $\mathrm{var}(F_X)$. By Lemmas 8 and 10, $\mathrm{sat}(F) = \mathrm{sat}(F_X) \geq (3m + k_U)/4$, where

$$\begin{aligned} k_U &= \sum_{i=1}^{q}(c_X(x_i) - c_X(\overline{x_i})) + \sum_{1\leq i<j\leq q}(c_X(x_i\overline{x_j}) + c_X(\overline{x_i}x_j) - c_X(x_ix_j) - c_X(\overline{x_i}\overline{x_j})) \\ &= \sum_{i=1}^{q} w_X(x_i) + \sum_{1\leq i<j\leq q} w_X(x_ix_j) \quad = \quad w_X(Q) \quad \geq \quad k. \end{aligned}$$

□

To apply Lemma 11 in the proof of Theorem 2, we will focus on a special case of induced subgraphs of G^0. For a set $U \subseteq V(G^0)$, let $G^0[U]$ denote the subgraph of G^0 induced by U. We call $G^0[U]$ an *induced star* with *center* x if x is a vertex of G^0, I is an independent set in the subgraph of G^0 induced by the neighbors of x and $U = \{x\} \cup I$. We are interested in the induced star due to the following property.

Lemma 12. *Let x be the center of an induced star $Q = G^0[U]$ and let $I = U \setminus \{x\}$. Then there is a set $X \subseteq U$ such that $w_X(Q) \geq |I|$.*

Proof. Let H be the set of edges of Q. We may assume that $w(xy) > 0$ for each $y \in I$ since otherwise we can switch y. By a *random switch* of Q, we mean a switch of Q with probability 0.5. Take a random switch R of Q. Then we have $\mathbb{E}(w_R(z)) = 0$ for all $z \in U$. Note that the sign of each edge in H remains positive. Hence we have $\mathbb{E}(w_R(Q)) = w(H) \geq |I|$ and thus there exists a set $X \subseteq U$ for which $w_X(Q) \geq |I|$. □

If we are given more than one induced star, a sequence of random switches gives us a similar result.

Lemma 13. *Let $Q_1 = (U_1, H_1), \ldots, Q_m = (U_m, H_m)$ be a collection of vertex-disjoint induced stars of G^0 with centers $x_1, \ldots, x_m$, let $U = \bigcup_{i=1}^{m} U_i$, and let $Q = G^0[U]$. Then there is a set $X \subseteq U$ such that $w_X(Q) \geq \sum_{i=1}^{m} |I_i|$, where $I_i = U_i \setminus \{x_i\}$, $i = 1, \ldots, m$.*

Proof. As in the proof of Lemma 12, we may assume that all the edges of H_i have positive weights. Let H be the set of edges of Q. By a *random switch* of Q, we mean a sequence of switches of $Q_1, \ldots, Q_m$ each with probability 0.5. Take a random switch R of Q. Then we have $\mathbb{E}(w_R(x)) = 0$ for all $x \in U$. Moreover, for the subgraph Q of G^0_R, it holds that $\mathbb{E}(w_R(xy)) = 0$ for all $xy \in H \setminus \bigcup_{i=1}^{m} H_i$ since each choice of $w_R(xy) \geq 0$ and $w_R(xy) \leq 0$ is equally likely. By linearity of expectation and Lemma 12, we have $\mathbb{E}(w_R(Q)) = w(\bigcup_{i=1}^{m} H_i) \geq \sum_{i=1}^{m} |I_i|$ and thus there exists a set $X \subseteq U$ for which $w_X(Q) \geq \sum_{i=1}^{m} |I_i|$. □

We are now in the position to complete the proof of Theorem 2.

Suppose that (F, k) is a no-instance, i.e., $\mathrm{sat}(F) < (3m + k)/4$. Notice that a matching can be viewed as a collection of induced stars of G^0 for which $|I_i| = 1$. It follows by Lemmas 11 and 13 that G^0 has no matching of size k. The Tutte-Berge formula [4, 6] states that the size of a maximum matching in G^0 equals

$$\min_{S \subseteq V(G^0)} \frac{1}{2}\{|V(G^0)| + |S| - oc(G^0 - S)\}$$

where $oc(G^0 - S)$ is the number of odd components (connected components with an odd number of vertices) in $G^0 - S$. Hence there is a set $S \subseteq V(G^0)$ such that $|V(G^0)| + |S| - oc(G^0 - S) < 2k$. It follows that

$$|V(G^0)| \leq oc(G^0 - S) - |S| + 2k - 1. \tag{3}$$

We will now classify odd components in $G^0 - S$. One obvious type of odd components is an isolated vertex in G^0 of weight zero, which corresponds to an insignificant variable by Lemma 9. All the other odd components can be categorized into one of the following two types:

1. Let $Q_1, \ldots, Q_L$ be the odd components of $G^0 - S$ such that for all $1 \leq i \leq L$ we have $|Q_i| = 1$ and Q_i is a significant variable.
2. Let $Q'_1, \ldots, Q'_{L'}$ be the odd components of $G^0 - S$ such that for all $1 \leq i \leq L'$ we have $|Q'_i| > 1$.

We construct a collection of induced stars as follows. From each of $Q'_1, \ldots, Q'_{L'}$ we choose an edge, which is an induced star with $|I| = 1$. Let us consider $Q_1, \ldots, Q_L$. Each vertex Q_i is adjacent to at least one vertex of S. Thus, we can partition $Q_1, \ldots, Q_L$ into $|S|$ sets, some of them possibly empty, such that each partite set forms an independent set in which every vertex is adjacent to the corresponding vertex x_i of S. Each partite set, together with x_i, forms an induced star. Now observe that we have a collection of induced stars and the total number of edges equals $L + L'$. If $L + L' \geq k$, Lemma 13 implies that for some set X of vertices from the odd components $w_X(Q) \geq k$, which is impossible by Lemma 11. Hence $L + L' \leq k - 1$.

Therefore, $oc(G^0 - S) - n' = L + L' \leq k - 1$, where n' is the number of insignificant variables. By (3), we have $|V(G^0)| - n' \leq k - 1 - |S| + 2k - 1 \leq 3k - 2$. It remains to observe that $|V(G^0)| - n'$ equals the number of significant variables of F. This completes the proof of Theorem 2.

Corollary 2. *The problem* Max-2-Sat$_{\text{TLB}}$ *admits a (polynomial time) reduction to a problem kernel with at most* $3k - 1$ *variables.*

Proof. Consider an instance (F, k) of the problem. First we apply the semicomplete reduction and obtain (in polynomial time) an instance (F', k) with $F' = F^S$. We determine (again in polynomial time) the set S' of significant variables of F'. If $|S'| > 3k - 2$ then (F', k) is a yes-instance by Theorem 2, and consequently (F, k) is a yes-instance by Lemma 7. Assume now that $|S'| \leq 3k - 2$.

Let z be a new variable not occurring in F. Since $F' = F^S$, no clause contains two insignificant variables and, thus, each insignificant variable can be replaced by z without changing the solution to (F', k). Let us denote the modified F' by F''; F'' has at most $3k - 1$ variables.

Let p be the number of clauses in F''. Observe that we can find a truth assignment satisfying the maximum number of clauses of F'' in time $O(p8^k)$. Thus, if $p > 8^k$, we can find the optimal truth assignment in the polynomial time $O(p^2) = O(m^2)$. Thus, we may assume that F'' has at most 8^k clauses. Therefore, F'' is a kernel of the MAX-2-SAT$_{\text{TLB}}$ problem. □

6 Concluding Remarks

• Our algorithm for the parameterized MAX-r-SAT problem can be easily modified to provide, efficiently, for any given instance of m clauses to which there is a truth assignment satisfying at least $k/2^r$ clauses above the average, an assignment for the variables with this property. Indeed, the proof of Theorem 1 only requires that the variables x_i are $4r$-wise independent, and there are known constructions of polynomial size sample spaces supporting such random variables (see, e.g., [2], Chapter 16). Thus, if in the polynomial X, $\sqrt{|\mathcal{S}|}/(2 \cdot 8^r) > k$, then one can find an assignment satisfying at least as many clauses as needed by going over all points in such a sample space, and if not, one can solve the problem by an exhaustive search.

• The fixed-parameter tractability result on MAX-r-SAT can be easily extended to any family of Boolean r-Constraint Satisfaction Problems. Here is an outline of the argument.

Let r be a fixed positive integer, let Φ be a set of Boolean functions, each involving at most r variables, and let $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$ be a collection of Boolean functions, each being a member of Φ, and each acting on some subset of the n Boolean variables $x_1, x_2, \ldots, x_n$. The Boolean Max-r-Constraint Satisfaction Problem (corresponding to Φ), which we denote by the MAX-r-CSP problem, for short, when Φ is clear from the context, is the problem of finding a truth assignment to the variables so as to maximize the total number of functions satisfied. Note that this includes, as a special case, the MAX-r-SAT problem considered in the previous section, as well as many related problems. As most interesting problems of this type are NP-hard, we consider their parameterized version, where the parameter is, as before, the number of functions satisfied minus the expected value of this number. Note, in passing, that the above expected value is a tight lower bound for the problem, whenever the family Φ is closed under replacing each variable by its complement, since if we apply any Boolean function to all 2^r choices of literals whose underlying variables are any fixed set of r variables, then any truth assignment to the variables satisfies exactly the same number of these 2^r functions.

For each Boolean function f of r Boolean variables

$$x_{i_1}, x_{i_2}, \ldots, x_{i_r},$$

define a random variable X_f as follows. As in the discussion of the MAX-r-SAT problem, suppose each variable x_{i_j} attains values in $\{-1, 1\}$. Let $V \subset \{-1, 1\}^r$ denote the set of all satisfying assignments of f. Then

$$X_f(x_1, x_2, \ldots, x_n) = \sum_{v=(v_1,\ldots,v_r)\in V} [\prod_{j=1}^{r}(1 + x_{i_j} v_j) - 1].$$

This is a random variable defined over the space $\{-1, 1\}^n$ and its value at $x = (x_1, x_2, \ldots, x_n)$ is $2^r - |V|$ if x satisfies f, and is $-|V|$ otherwise. Thus, the expectation of X_f is zero. Define now $X = \sum_{f\in\mathcal{F}} X_f$. Then the value of X at $x = (x_1, x_2, \ldots, x_n)$ is precisely $2^r(s - a)$, where s is the number of the functions satisfied by the truth assignment x, and a is the average value of the number of satisfied functions. Our objective is to decide if X attains a value of at least k. As this is a polynomial of degree at most r with integer coefficients and expectation zero, we can repeat the arguments of Section 4 and prove that, for every fixed r, the problem is fixed-parameter tractable. Moreover, our previous arguments show that the problem admits a polynomial-size bikernel reducing it to an instance of MAX-r-LIN2$_{\text{TLB}}$ of size $O(k^2)$, and if the specific r-CSP problem considered is NP-complete, then there is a polynomial size kernel. This is the case for most interesting choices of the family Φ.

Acknowledgments Research of Alon was partially supported by an ERC Advanced grant. Research of Gutin, Kim and Yeo was supported in part by an EPSRC grant.

References

[1] N. Alon, G. Gutin and M. Krivelevich, Algorithms with large domination ratio, J. Algorithms 50 (2004), 118–131.

[2] N. Alon and J.H. Spencer, *The Probabilistic Method*, 3rd edition, Wiley, New York, 2008.

[3] B. Aspvall, M. F. Plass, and R. E. Tarjan. A linear-time algorithm for testing the truth of certain quantified Boolean formulas. *Information Processing Letters*, 8(3):121–123, 1979.

[4] C. Berge. Sur le couplage maximum d'un graphe. *C. R. Acad. Sci. Paris*, 247:258–259, 1958.

[5] H.L. Bodlaender, S. Thomassé and A. Yeo, Kernel Bounds for Disjoint Cycles and Disjoint Paths. *Proc. ESA 2009*, Lect. Notes Comput. Sci. Volume 5757:635–646, 2009.

[6] J. A. Bondy and U. S. R. Murty. *Graph Theory*, volume 244 of *Graduate Texts in Mathematics*. Springer Verlag, New York, 2008.

[7] J. Bourgain, Walsh subspaces of L^p-product spaces. Seminar on Functional Analysis, 1979–1980 (in French), 1980, Exp. No. 4A, 9.

[8] R. G. Downey and M. R. Fellows. *Parameterized Complexity.* Monographs in Computer Science. Springer Verlag, 1999.

[9] J. Flum and M. Grohe. *Parameterized Complexity Theory*, volume XIV of *Texts in Theoretical Computer Science. An EATCS Series.* Springer Verlag, 2006.

[10] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified NP-complete graph problems. *Theoret. Comput. Sci.*, 1(3):237–267, 1976.

[11] G. Gutin, E. J. Kim, M. Mnich, and A. Yeo. Ordinal Embedding Relaxations Parameterized Above Tight Lower Bound. *CoRR* technical report (abs/0907.5427), http://arxiv.org/abs/0907.5427.

[12] G. Gutin, E. J. Kim, S. Szeider, and A. Yeo. A probabilistic approach to problems parameterized above tight lower bound. *CoRR* technical report (abs/0906.1356), http://arxiv.org/abs/0906.1356. (Accepted at IWPEC'09, The Fourth International Workshop on Parameterized and Exact Computation, IT University of Copenhagen, Denmark, September 10–11, 2009.)

[13] G. Gutin, A. Rafiey, S. Szeider, and A. Yeo. The linear arrangement problem parameterized above guaranteed value. *Theory Comput. Syst.*, 41:521–538, 2007.

[14] G. Gutin, S. Szeider, and A. Yeo. Fixed-parameter complexity of minimum profile problems. *Algorithmica*, 52(2):133–152, 2008.

[15] J. Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001.

[16] J. Håstad and S. Venkatesh. On the advantage over a random assignment. In *Proceedings of the Thirty-Fourth Annual ACM Symposium on Theory of Computing*, pages 43–52, New York, 2002. ACM. Full version appeared in *Random Structures Algorithms* 25(2) (2004), pp. 117–149.

[17] P. Heggernes, C. Paul, J. A. Telle, and Y. Villanger. Interval completion with few edges. In *STOC'07—Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, pages 374–381. ACM, 2007. Full version appeared in *SIAM J. Comput.* 38(5), 2008/09.

[18] K. Iwama. CNF-satisfiability test by counting and polynomial average time. *SIAM J. Comput.*, 18(2):385–391, 1989.

[19] D. S. Johnson. Approximation algorithms for combinatorial problems. In *Fifth Annual ACM Symposium on Theory of Computing (Austin, Tex., 1973)*, pages 38–49. Assoc. Comput. Mach., New York, 1973.

[20] M. Mahajan and V. Raman. Parameterizing above guaranteed values: MaxSat and MaxCut. *J. Algorithms*, 31(2):335–354, 1999.

[21] M. Mahajan, V. Raman, and S. Sikdar. Parameterizing above or below guaranteed values. *J. of Computer and System Sciences*, 75(2):137–153, 2009.

[22] R. Niedermeier. *Invitation to Fixed-Parameter Algorithms.* Oxford Lecture Series in Mathematics and its Applications. Oxford University Press, 2006.