Satisfying More Than Half of a System of Linear Equations Over GF(2): A Multivariate Approach[*]

R. Crowston[†]   M. Fellows[‡]   G. Gutin[†]   M. Jones[†]   E.J. Kim[§]

F. Rosamond[‡]   I.Z. Ruzsa[¶]   S. Thomassé[‖]   A. Yeo[**]

Abstract

In the parameterized problem MaxLin2-AA[k], we are given a system with variables $x_1, \ldots, x_n$ consisting of equations of the form $\prod_{i \in I} x_i = b$, where $x_i, b \in \{-1, 1\}$ and $I \subseteq [n]$, each equation has a positive integral weight, and we are to decide whether it is possible to simultaneously satisfy equations of total weight at least $W/2 + k$, where W is the total weight of all equations and k is the parameter (it is always possible for $k = 0$). We show that MaxLin2-AA[k] has a kernel with at most $O(k^2 \log k)$ variables and can be solved in time $2^{O(k \log k)}(nm)^{O(1)}$. This solves an open problem of Mahajan *et al.* (2006). The problem Max-r-Lin2-AA[k, r] is the same as MaxLin2-AA[k] with two differences: each equation has at most r variables and r is the second parameter. We prove that Max-r-Lin2-AA[k, r] has a kernel with at most $(2k - 1)r$ variables.

1 Introduction

1.1 MaxLin2-AA and Max-r-Lin2-AA. While MaxSat and its special case Max-r-Sat have been widely studied in the literature on algorithms and complexity for many years, MaxLin2 and its special case Max-r-Lin2 are less well known, but Håstad [29] succinctly summarized the importance of these

---

two problems by saying that they are "as basic as satisfiability." These problems provide important tools for the study of constraint satisfaction problems such as MaxSat and Max-r-Sat since constraint satisfaction problems can often be reduced to MaxLin2 or Max-r-Lin2, see, e.g., [1, 2, 13, 14, 29, 31]. Accordingly, in the last decade, MaxLin2 and Max-r-Lin2 have attracted significant attention in algorithmics.

The problem MaxLin2 can be stated as follows. We are given a system of m equations in variables $x_1, \ldots, x_n$, where each equation is $\prod_{i \in I_j} x_i = b_j$ and $x_i, b_j \in \{-1, 1\}$, $j = 1, \ldots, m$, and where each equation is assigned a positive integral weight w_j. We are required to find an assignment of values to the variables in order to maximize the total weight of the satisfied equations.

Let W be the sum of the weights of all equations in S and let $\text{sat}(S)$ be the maximum total weight of equations that can be satisfied simultaneously. To see that $W/2$ is a tight lower bound on $\text{sat}(S)$ choose assignments to the variables independently and uniformly at random. Then $W/2$ is the expected weight of satisfied equations (as the probability of each equation being satisfied is $1/2$) and thus $W/2$ is a lower bound; to see the tightness consider a system consisting of pairs of equations of the form $\prod_{i \in I} x_i = -1$, $\prod_{i \in I} x_i = 1$ of the same weight, for some non-empty sets $I \subseteq [n]$. This leads to the following decision problem:

MaxLin2-AA
Instance: A system S of equations $\prod_{i \in I_j} x_i = b_j$, where $x_i, b_j \in \{-1, 1\}$, $j = 1, \ldots, m$ and where each equation is assigned a positive integral weight w_j; and a nonnegative integer k.
Question: $\text{sat}(S) \geq W/2 + k$?

The maximization version of MaxLin2-AA (maximize k for which the answer is Yes), has been studied in the literature on approximation algorithms, cf. [29, 30]. These two papers also studied the following important special case of MaxLin2-AA:

Max-r-Lin2-AA
Instance: A system S of equations $\prod_{i \in I_j} x_i = b_j$, where $x_i, b_j \in \{-1, 1\}$, $|I_j| \leq r$, $j = 1, \ldots, m$; equation j is assigned a positive integral weight w_j, and a nonnegative integer k.
Question: $\text{sat}(S) \geq W/2 + k$?

Håstad [29] proved that, as a maximization problem, Max-r-Lin2-AA with any fixed $r \geq 3$ (and hence MaxLin2-AA) cannot be approximated within a constant factor c for any $c > 1$ unless P=NP (that is, the problem is not in APX unless P=NP). Håstad and Venkatesh [30] obtained some approximation algorithms for the two problems. In particular, they proved that for

2

Max-r-Lin2-AA there exists a constant $c > 1$ and a randomized polynomial-time algorithm that, with probability at least $3/4$, outputs an assignment with an approximation ratio of at most $c^r \sqrt{m}$.

The problem MaxLin2-AA was first studied in the context of parameterized complexity by Mahajan *et al.* [33] who naturally took k as the parameter[1]. We will denote this parameterized problem by MaxLin2-AA[k]. Despite some progress [13, 14, 27], the complexity of MaxLin2-AA[k] has remained prominently open in the research area of "parameterizing above guaranteed bounds" that has attracted much recent attention (cf. [1, 13, 14, 27, 31, 33]) and that still poses well-known and longstanding open problems (e.g., how difficult is it to determine if a planar graph has an independent set of size at least $(n/4) + k$?). One can parameterize Max-r-Lin2-AA by k for any fixed r (denoted by Max-r-Lin2-AA[k]) or by both k and r (denoted by Max-r-Lin2-AA[k, r])[2].

Define the *excess* for $x^0 = (x_1^0, \ldots, x_n^0) \in \{-1, 1\}^n$ over S to be

$$\varepsilon_S(x^0) = \sum_{j=1}^{m} c_j \prod_{i \in I_j} x_i^0, \quad \text{where } c_j = w_j b_j.$$

Note that $\varepsilon_S(x^0)$ is the total weight of equations satisfied by x^0 minus the total weight of equations falsified by x^0. The maximum possible value of $\varepsilon_S(x^0)$ is the *maximum excess* of S. Håstad and Venkatesh [30] initiated the study of the excess of a system of equations and further research on the topic was carried out by Crowston *et al.* [14] who concentrated on MaxLin2-AA. In this paper, we study the maximum excess for both MaxLin2-AA and Max-r-Lin2-AA. Note that the excess is a *pseudo-boolean function* [10], i.e., a function that maps $\{-1, 1\}^n$ to the set of reals.

1.2 Main Results and Structure of the Paper. Roughly speaking, a kernelization is a polynomial-time algorithm that transforms an instance $\mathcal{I}$ of the parameterized decision problem under consideration into an equivalent instance (called a kernel) $\mathcal{I}'$ of the same problem such that both the size of $\mathcal{I}'$ and the value of its parameter are bounded from above by a function in the parameter of $\mathcal{I}$ only.

Henceforth, $O(1)$ will denote an arbitrary absolute constant.

The main results of this paper are Theorems 3 and 4. In 2006 Mahajan *et al.* [33] introduced MaxLin2-AA[k] and asked what is its complexity. We

---

[1] We provide basic definitions on parameterized algorithms and complexity in Subsection 1.4 below.

[2] While in the preceding literature only MaxLin2-AA[k] was considered, we introduce and study Max-r-Lin2-AA[k, r] in the spirit of Multivariate Algorithmics as outlined by Fellows *et al.* [23] and Niedermeier [36].

answer this question in Theorem 3 by showing that MAXLIN2-AA[k] admits a kernel with at most $O(k^2 \log k)$ variables. In our kernel, it is only the number of variables that is bounded from above by a polynomial in k. In our kernel, the number of equations is only bounded by an exponential function of k. These two results imply that MAXLIN2-AA[k] admits a kernel and, hence, is fixed-parameter tractable (see Section 1.4).

The proof of Theorem 3 is based on two results: (a) If S is an irreducible system (i.e., a system that cannot be reduced using Rule 1 or 2 defined in Section 2) of MAXLIN2-AA[k] and $2k \le m \le 2^{n/(2k-1)} - 2$, then S is a YES-instance; (b) there is an algorithm for MAXLIN2-AA[k] of complexity $n^{2k}(nm)^{O(1)}$. To prove (a), we introduce a new notion of a sum-free subset of vectors over $\mathbb{F}_2$ and show the existence of such subsets using linear algebra. We also prove that MAXLIN2-AA[k] can be solved in time $2^{O(k \log k)}(nm)^{O(1)}$ (Corollary 1).

The other main result of this paper, Theorem 4, gives a sharp lower bound on the maximum excess for MAX-r-LIN2-AA as follows. Let S be an irreducible system and suppose that each equation contains at most r variables. Let $n \ge (k-1)r + 1$ and let $w_{\min}$ be the minimum weight of an equation of S. Then, in time $O(m^{O(1)})$, we can find an assignment x^0 to variables of S such that $\varepsilon_S(x^0) \ge k \cdot w_{\min}$. Essentially Theorem 4 follows from the existence of sum-free sets of vectors satisfying some simple conditions.

In Section 2, we give some reduction rules for MAX-r-LIN2-AA, describe an algorithm $\mathcal{H}$ introduced by Crowston *et al.* [14] and give some properties of the maximum excess, irreducible systems and Algorithm $\mathcal{H}$. In Section 3, we prove Theorem 3 and Corollary 1. A key tool in our proof of Theorem 4 is a lemma on sum-free subsets in a set of vectors from $\mathbb{F}_2^n$. The lemma and Theorem 4 are proved in Section 4. We prove several corollaries of Theorem 4 in Section 5. The corollaries are relevant to parameterized and approximation algorithms, as well as lower bounds for the maxima of pseudo-boolean functions and their applications in graph theory.

Our results on parameterized algorithms improve a number of previously known results including those of Kim and Williams [31]. In Section 6, we discuss some recent results and open problems.

1.3 Corollaries of Theorem 4. The following results have been obtained for MAX-r-LIN2-AA[k] when r is fixed and for MAX-r-LIN2-AA[k, r]. Gutin *et al.* [27] proved that MAX-r-LIN2-AA[k] is fixed-parameter tractable and, moreover, has a kernel with $n \le m = O(k^2)$. This kernel is, in fact, a kernel of MAX-r-LIN2-AA[k, r] with $n \le m = O(9^r k^2)$. This kernel for MAX-r-LIN2-AA[k] was improved by Crowston *et al.* [14], with respect to the number of variables, to $n = O(k \log k)$. For MAX-r-LIN2-AA[k], Kim and Williams [31] were the first to obtain a kernel with a linear number of variables, $n =$

$O(k)$, for fixed r. This kernel is, in fact, a kernel with the number of variables $n \leq r(r+1)k$ for MAX-r-LIN2-AA$[k, r]$. In this paper, we obtain a kernel with $n \leq (2k-1)r$ for MAX-r-LIN2-AA$[k, r]$. As an easy consequence of this result we show that the maximization problem MAX-r-LIN2-AA is in APX if restricted to $m = O(n)$ for all fixed r, where the weight of each equation is bounded by a constant. This is in sharp contrast with the fact mentioned above that for each $r \geq 3$, MAX-r-LIN2-AA with arbitrary weights is not in APX unless P=NP.

Fourier analysis of *pseudo-boolean functions*, i.e., functions $f : \{-1, 1\}^n \to \mathbb{R}$, has been used in many areas of computer science (cf. [1, 11, 14, 37]). In Fourier analysis, the Boolean domain is often assumed to be $\{-1, 1\}^n$ rather than more usual $\{0, 1\}^n$ and we will follow this assumption in our paper. Here we use the following well-known and easy to prove fact (see, e.g., [37]): each function $f : \{-1, 1\}^n \to \mathbb{R}$ can be uniquely written as

$$f(x) = \hat{f}(\emptyset) + \sum_{I \in \mathcal{F}} \hat{f}(I) \prod_{i \in I} x_i. \tag{1}$$

where $\mathcal{F} \subseteq \{I : \emptyset \neq I \subseteq [n]\}$, $[n] = \{1, 2, \ldots, n\}$, $x_i \in \{-1, 1\}$, and $\hat{f}(I)$ are non-zero reals. Formula (1) is the *Fourier expansion* of f and $\hat{f}(I)$ are the *Fourier coefficients* of f. The right hand side of (1) is a polynomial and the degree $\max\{|I| : I \in \mathcal{F}\}$ of this polynomial will be called the *degree* of f. Let A be a $(0, 1)$-matrix with n rows and $|\mathcal{F}|$ columns and with entries a_{ij} such that $a_{ij} = 1$ if and only if term j in (1) contains x_i.

In Section 5, we obtain the following lower bound on the maximum of a pseudo-boolean function f of degree r:

$$\max_x f(x) \geq \hat{f}(\emptyset) + \lfloor (\operatorname{rank}(A) + r - 1)/r \rfloor \cdot \min\{|\hat{f}(I)| : I \in \mathcal{F}\}, \tag{2}$$

where $\operatorname{rank}(A)$ is the rank of A over $\mathbb{F}_2$. (Note that since $\operatorname{rank}(A)$ does not depend on the order of the columns in A, we may order the terms in (1) arbitrarily.)

To demonstrate the combinatorial usefulness of (2), we apply it to obtain a short proof of the well-known lower bound of Edwards-Erdős on the maximum size of a bipartite subgraph in a graph (the MAX CUT problem). Erdős [21] conjectured and Edwards [20] proved that every connected graph with n vertices and m edges has a bipartite subgraph with at least $m/2 + (n-1)/4$ edges. For short graph-theoretical proofs, see, e.g., Bollobás and Scott [8] and Erdős *et al.* [22]. We consider the BALANCED SUBGRAPH problem [4] that generalizes MAX CUT and show that our proof of the Edwards-Erdős bound can be easily extended to BALANCED SUBGRAPH.

1.4 Parameterized Complexity and (Bi)kernelization. A *parameterized problem* is a subset $L \subseteq \Sigma^* \times \mathbb{N}$ over a finite alphabet Σ. L is *fixed-parameter*

tractable (FPT, for short) if membership of an instance (x, k) in $\Sigma^* \times \mathbb{N}$ can be decided in time $f(k)|x|^{O(1)}$, where f is a function of the parameter k only. If membership can be decided in time $O(|x|^{f(k)})$ then L belongs to the parameterized complexity class XP. It is known that FPT is a proper subset of XP [19]. Analogs of NP are provided by the classes of parameterized problems of the W[t] Hierarchy giving the tower: FPT $\subseteq$ W[1] $\subseteq$ W[2] $\subseteq \cdots \subseteq$ XP. For the definition of the classes W[t], see, e.g., [19, 24].

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. The notion of a bikernelization was introduced in [1], where it was observed that a parameterized problem L is fixed-parameter tractable if and only if it is decidable and admits a bikernelization to a parameterized problem L'. A *kernelization* of a parameterized problem L is simply a bikernelization from L to itself; the bikernel is the *kernel*, and $g(k)$ is the *size* of the kernel. Due to the importance of polynomial-time kernelization algorithms in applied multivariate algorithmics, low degree polynomial size kernels and bikernels are of considerable interest, and the subject has developed substantial theoretical depth, cf. [1, 5, 6, 7, 18, 24, 25, 26, 27].

The case of several parameters $k_1, \ldots, k_t$, as for MAX-r-LIN2-AA$[k, r]$, can be reduced to the one parameter case by setting $k = k_1 + \cdots + k_t$, see, e.g., [18]. We remark that other ways of handling multiple parameters (other than just adding them up) is an area of active investigation, e.g., see [23].

2 Maximum Excess, Irreducible Systems and Algorithm $\mathcal{H}$

Recall that an instance of MAXLIN2-AA consists of a system S of equations $\prod_{i \in I_j} x_i = b_j$, $j \in [m]$, where $\emptyset \neq I_j \subseteq [n]$, $b_j \in \{-1, 1\}$, $x_i \in \{-1, 1\}$. An equation $\prod_{i \in I_j} x_i = b_j$ has an integral positive weight w_j. Recall that the excess for $x^0 = (x_1^0, \ldots, x_n^0) \in \{-1, 1\}^n$ over S is $\varepsilon_S(x^0) = \sum_{j=1}^m c_j \prod_{i \in I_j} x_i^0$, where $c_j = w_j b_j$. The excess $\varepsilon_S(x^0)$ is the total weight of equations satisfied by x^0 minus the total weight of equations falsified by x^0. The maximum possible value of $\varepsilon_S(x^0)$ is the maximum excess of S.

Remark 1. *Observe that the answer to* MAXLIN2-AA *is* YES *if and only if the maximum excess is at least $2k$.*

Remark 2. *The excess $\varepsilon_S(x)$ is a pseudo-boolean function and its Fourier expression is $\varepsilon_S(x) = \sum_{j=1}^m c_j \prod_{i \in I_j} x_i$. Moreover, observe that every pseudo-boolean function $f(x) = \sum_{I \in \mathcal{F}} \hat{f}(I) \prod_{i \in I} x_i$ (where $\hat{f}(\emptyset) = 0$) is the excess*

over the system $\prod_{i \in I} x_i = b_I$, $I \in \mathcal{F}$, *where* $b_I = 1$ *if* $\hat{f}(I) > 0$ *and* $b_I = -1$ *if* $\hat{f}(I) < 0$, *with weights* $|\hat{f}(I)|$. *Thus, studying the maximum excess over a* MaxLin2-AA-*system (with real weights) is equivalent to studying the maximum of a pseudo-boolean function.*

Consider two reduction rules for MaxLin2 studied in [27]. Rule 1 was studied before in [30].

Reduction Rule 1. *If we have, for a subset I of $[n]$, an equation $\prod_{i \in I} x_i = b'_I$ with weight w'_I, and an equation $\prod_{i \in I} x_i = b''_I$ with weight w''_I, then we replace this pair by one of these equations with weight $w'_I + w''_I$ if $b'_I = b''_I$ and, otherwise, by the equation whose weight is bigger, setting its new weight to $|w'_I - w''_I|$. If the resulting weight is 0, we delete the equation from the system.*

Remark 3. *In what follows, we will use the fact that a maximum independent set in a set M of vectors from $\mathbb{F}_2^n$, can be found in polynomial time in n and $|M|$ [32].*

Henceforth, rank(A) will denote the rank of A over $\mathbb{F}_2$.

Reduction Rule 2. *Let A be the matrix over $\mathbb{F}_2$ corresponding to the set of equations in S, such that $a_{ji} = 1$ if $i \in I_j$ and 0, otherwise. Let $t = \text{rank}(A)$ and suppose columns $a^{i_1}, \ldots, a^{i_t}$ of A are linearly independent. Then delete all variables not in $\{x_{i_1}, \ldots, x_{i_t}\}$ from the equations of S.*

Observe that after applying Rule 2, the resulting matrix A has rank equal n.

Lemma 1. *[27] Let S' be obtained from S by Rule 1 or 2. Then the maximum excess of S' is equal to the maximum excess of S. Moreover, S' can be obtained from S in time polynomial in n and m.*

If we cannot change a weighted system S using Rules 1 and 2, we call it *irreducible*.

Lemma 2. *Let S' be a system obtained from S by first applying Rule 1 as long as possible and then Rule 2 as long as possible. Then S' is irreducible.*

Proof. Let S^* denote the system obtained from S by applying Rule 1 as long as possible. Without loss of generality, assume that $x_1 \notin \{x_{i_1}, \ldots, x_{i_t}\}$ (see the description of Rule 2) and thus Rule 2 removes x_1 from S^*. To prove the lemma it suffices to show that after the removal of x_1 no pair of equations has the same left hand side. Suppose that there is a pair of equations in S^* which has the same left hand side after the removal of x_1; let $\prod_{i \in I'} x_i = b'$ and $\prod_{i \in I''} x_i = b''$ be such equations and let $I' = I'' \cup \{1\}$. Then the entries

7

of the first column of A, a^1, corresponding to the pair of equations are 1 and 0, but in all the other columns of A the entries corresponding to the pair of equations are either 1,1 or 0,0. Thus, a^1 is independent from all the other columns of A, a contradiction. $\qquad\square$

Let S be an irreducible system of MaxLin2-AA. Consider the following algorithm. We assume that, in the beginning, no equation or variable in S is marked.

ALGORITHM $\mathcal{H}$

While the system S is nonempty do the following:

1. Choose an equation $\prod_{i \in I} x_i = b$ and mark a variable x_l such that $l \in I$.

2. Mark this equation and delete it from the system.

3. Replace every equation $\prod_{i \in I'} x_i = b'$ in the system containing x_l by $\prod_{i \in I \Delta I'} x_i = bb'$, where $I \Delta I'$ is the symmetric difference of I and I' (the weight of the equation is unchanged).

4. Apply Reduction Rule 1 to the system.

The *maximum $\mathcal{H}$-excess* of S is the maximum possible total weight of equations marked by $\mathcal{H}$ for S taken over all possible choices in Step 1 of $\mathcal{H}$. The following lemma indicates the potential power of $\mathcal{H}$.

Lemma 3. *Let S be an irreducible system. Then the maximum excess of S equals its maximum $\mathcal{H}$-excess. Furthermore, for any set of equations marked by Algorithm $\mathcal{H}$, in polynomial time, we can find an assignment of excess at least the total weight of marked equations.*

Proof. We first prove that the maximum excess of S is not smaller than its maximum $\mathcal{H}$-excess. By construction, for any assignment that satisfies all the marked equations, exactly half of the non-marked equations are satisfied. Therefore it suffices to find an assignment to the variables such that all marked equations are satisfied. Assign arbitrary values to the unmarked variables. Then assign values to the marked variables in the order opposite to which they were marked such that the corresponding marked equations are satisfied.

The above argument also proves the last statement of the lemma.

Now we prove that the maximum $\mathcal{H}$-excess of S is not smaller than its maximum excess. Let $x^0 = (x_1^0, \ldots, x_n^0)$ be an assignment that achieves the maximum excess, t. Observe that if at each iteration of $\mathcal{H}$ we mark an equation that is satisfied by x^0, then $\mathcal{H}$ will mark equations of total weight t. $\qquad\square$

3 MaxLin2-AA

The following two theorems provide a basis for proving Theorem 3, the main result of this section.

Theorem 1. *There exists an $O(n^{2k}(nm)^{O(1)})$-time algorithm for* MaxLin2-AA*[k] that returns an assignment of excess of at least $2k$ if one exists, and returns* NO *otherwise.*

Proof. Suppose we have an instance $\mathcal{L}$ of MaxLin2-AA$[k]$ that is reduced by Rules 1 and 2, and that the maximum excess of $\mathcal{L}$ is at least $2k$. Let A be the matrix introduced in Rule 2. Pick n equations $e_1, \ldots, e_n$ such that their rows in A are linearly independent. An assignment of excess at least $2k$ must either satisfy one of these equations, or falsify them all. If they are all falsified, then the system of equations $\bar{e}_1, \ldots, \bar{e}_n$, where each $\bar{e}_i$ is e_i with the changed right hand side, has a unique solution, an assignment of values to $x_1, \ldots, x_n$. If this assignment does not give excess at least $2k$ for $\mathcal{L}$, then any assignment that leads to excess at least $2k$ must satisfy at least one of $e_1, \ldots, e_n$. Thus, by Lemma 3, algorithm $\mathcal{H}$ can mark one of these equations and achieve an excess of at least $2k$.

This gives us the following depth-bounded search tree. At each node N of the tree, reduce the system by Rules 1 and 2, and let n' be the number of variables in the reduced system. Then find n' equations $e_1, \ldots, e_{n'}$ corresponding to linearly independent vectors. Find an assignment of values to $x_1, \ldots, x_{n'}$ that falsifies all of $e_1, \ldots, e_{n'}$. Check whether this assignment achieves excess of at least $2k - w^*$, where w^* is total weight of equations marked by $\mathcal{H}$ in all predecessors of N. If it does, then return the assignment and stop the algorithm. Otherwise, split into n' branches. In the i'th branch, run an iteration of $\mathcal{H}$ marking equation e_i. Then repeat this algorithm for each new node. Whenever the total weight of marked equations is at least $2k$, return the suitable assignment. Clearly, the algorithm will terminate without an assignment if the maximum excess of $\mathcal{L}$ is less than $2k$.

All the operations at each node take time $O((nm)^{O(1)})$, and there are less than n^{2k+1} nodes in the search tree. Therefore this algorithm takes time $O(n^{2k}(nm)^{O(1)})$. □

The following lemma is used to prove Theorem 2, but it might be also of independent interest. Let K and M be sets of vectors in $\mathbb{F}_2^n$ such that $K \subseteq M$. We say K is *M-sum-free* if no sum of two or more distinct vectors in K is equal to a vector in M. Observe that K is M-sum-free if and only if K is linearly independent and no sum of vectors in K is equal to a vector in $M \backslash K$.

Lemma 4. *Let M be a set in $\mathbb{F}_2^n$ such that M contains a basis of $\mathbb{F}_2^n$, the zero vector is in M and $|M| < 2^n$. If k is a positive integer and $k+1 \leq |M| \leq 2^{n/k}$*

then, in time polynomial in $|M|$ and n, we can find an M-sum-free set K of $k + 1$ vectors.

Proof. We first consider the case when $k = 1$. Since $|M| < 2^n$ and the zero vector is in M, there is a non-zero vector $v \notin M$. Since M contains a basis for $\mathbb{F}_2^n$, the vector v can be written as a sum of vectors in M. Consider such a sum with the minimum number of summands: $v = u_1 + \cdots + u_\ell$, $\ell \geq 2$. Since $u_1 + u_2 \notin M$, we may set $K = \{u_1, u_2\}$. We can find such a set K in polynomial time by looking at every pair in $M \times M$.

We now assume that $k > 1$. Since $k + 1 \leq |M| \leq 2^{n/k}$ and $k \geq 2$, we have $n \geq k + 1$.

We proceed with a greedy algorithm that tries to find K. Suppose we have a set $L = \{a_1, \ldots, a_l\}$ of vectors in M, $l \leq k$, such that no sum of two or more elements of L is in M. We can extend this set to a basis, so $a_1 = (1, 0, 0, \ldots, 0)$, $a_2 = (0, 1, 0, \ldots, 0)$ and so on. For every $a \in M \backslash L$ we check whether $M \backslash \{a_1, \ldots, a_l, a\}$ has an element that agrees with a in all co-ordinates $l + 1, \ldots, n$. If no such element exists, then we add a to the set L, as no element in M can be expressed as a sum of a and a subset of L.

If our greedy algorithm finds a set L of size at least $k + 1$, we are done and L is our set K. Otherwise, we have stopped at $l \leq k$. In this case, we do the next iteration as follows. Recall that L is part of a basis of M such that $a_1 = (1, 0, 0, \ldots, 0)$, $a_2 = (0, 1, 0, \ldots, 0), \ldots$. We create a new set M' in $\mathbb{F}_2^{n'}$, where $n' = n - l$. We do this[3] by removing the first l co-ordinates from M, and then identifying together any vectors that agree in the remaining n' co-ordinates. We are in effect identifying together any vectors that only differ by a sum of some elements in L. It follows that every element of M' was created by identifying together at least two elements of M, since otherwise we would have had an element in $M \backslash L$ that should have been added to L by our greedy algorithm. Therefore it follows that $|M'| \leq |M|/2 \leq 2^{n/k-1}$. From this inequality and the fact that $n' \geq n - k$, we get that $|M'| \leq 2^{n'/k}$. It also follows by the construction of M' that M' has a basis for $\mathbb{F}_2^{n'}$, and that the zero vector is in M'. (Thus, we have $|M'| \geq n' + 1$.) If $n' \geq k + 1$ we complete this iteration by running the algorithm on the set M' as in the first iteration. Otherwise ($n' \leq k$) and the algorithm stops.

Since each iteration of the algorithm decreases n', the algorithm terminates. Now we prove that at some iteration, the algorithm will actually find a set K of $k + 1$ vectors. To show this it suffices to prove that we will never reach the point when $n' \leq k$. Suppose this is not true and we obtained $n' \leq k$. Observe that $n' \geq 1$ (before that we had $n' \geq k + 1$ and we decreased n' by at most k) and $|M'| \geq n' + 1$. Since $|M'| \leq 2^{n'/k}$, we have $n' + 1 \leq 2^{n'/k}$, which

---

[3]For the reader familiar with vector space terminology: $\mathbb{F}_2^{n'}$ is $\mathbb{F}_2^n$ modulo span(L), the subspace of $\mathbb{F}_2^n$ spanned by L, and M' is the image of M in $\mathbb{F}_2^{n'}$.

is impossible due to $n' \le k$ unless $n' = 1$ and $k = 1$, a contradiction with the assumption that $k > 1$.

It is easy to check that the running time of the algorithm is polynomial in $|M|$ and n. $\qquad\square$

Theorem 2. *Let S be an irreducible system of* MAXLIN2-AA*[k] and let $k \ge 1$. If $2k \le m \le \min\{2^{n/(2k-1)} - 1, 2^n - 2\}$, then the maximum excess of S is at least $2k$. Moreover, we can find an assignment with excess of at least $2k$ in time $O(m^{O(1)})$.*

Proof. Consider a set M of vectors in $\mathbb{F}_2^n$ corresponding to equations in S as follows: for each equation $\prod_{i \in I} x_i = b$ in S, define a vector $v = (v_1, \dots, v_n) \in M$, where $v_i = 1$ if $i \in I$ and $v_i = 0$, otherwise. Add the zero vector to M.

As S is reduced by Rule 2 and $2k \le m \le \min\{2^{n/(2k-1)} - 1, 2^n - 2\}$, we have that M contains a basis for $\mathbb{F}_2^n$ and $2k \le |M| \le \min\{2^{n/(2k-1)}, 2^n - 1\}$. Therefore, using Lemma 4 we can find an M-sum-free set K of $2k$ vectors. Let $\{e_{j_1}, \dots, e_{j_{2k}}\}$ be the corresponding set of equations. Run Algorithm $\mathcal{H}$, choosing at Step 1 an equation of S from $\{e_{j_1}, \dots, e_{j_{2k}}\}$ each time, and let S' be the resulting system. Algorithm $\mathcal{H}$ will run for $2k$ iterations of the while loop as no equation from $\{e_{j_1}, \dots, e_{j_{2k}}\}$ will be deleted before it has been marked.

Indeed, suppose that this is not true. Then for some e_{j_l} and some other equation e in S, after applying Algorithm $\mathcal{H}$ for at most $l - 1$ iterations e_{j_l} and e contain the same variables. Thus, there are vectors $v_j \in K$ and $v \in M$ and a pair of nonintersecting subsets K' and K'' of $K \setminus \{v, v_j\}$ such that $v_j + \sum_{u \in K'} u = v + \sum_{u \in K''} u$. Thus, $v = v_j + \sum_{u \in K' \cup K''} u$, contradicting the definition of K.

Thus, by Lemma 3, we are done. $\qquad\square$

Theorem 3. *The problem* MAXLIN2-AA*[k] has a kernel with at most $O(k^2 \log k)$ variables.*

Proof. Let $\mathcal{L}$ be an instance of MAXLIN2-AA$[k]$ and let S be the system of $\mathcal{L}$ with m equations and n variables. We may assume that S is irreducible. Let the parameter k be an arbitrary positive integer.

If $m < 2k$ then $n < 2k = O(k^2 \log k)$ and we are done. Otherwise, if $2k \le m \le 2^{n/(2k-1)} - 2$ then, by Theorem 2 and Remark 1, the answer to $\mathcal{L}$ is YES and the corresponding assignment can be found in polynomial time. If $m \ge n^{2k} - 1$ then, by Theorem 1, we can solve $\mathcal{L}$ in polynomial time.

Finally we consider the case $2^{n/(2k-1)} - 2 \le m \le n^{2k} - 2$. Hence, $n^{2k} \ge 2^{n/(2k-1)}$. Therefore, $4k^2 \ge 2 + n/\log n \ge \sqrt{n}$ and $n \le (2k)^4$. Hence, $n \le 4k^2 \log n \le 4k^2 \log(16k^4) = O(k^2 \log k)$.

Since S is irreducible, $m < 2^n \le 2^{O(k^2 \log k)}$ and thus we have obtained the desired kernel. $\qquad\square$

Corollary 1. *The problem* MaxLin2-AA*[k] can be solved in time* $2^{O(k \log k)}(nm)^{O(1)}$.

Proof. Let $\mathcal{L}$ be an instance of MaxLin2-AA$[k]$. By Theorem 3, in time $O((nm)^{O(1)})$ either we solve $\mathcal{L}$ or we obtain a kernel with at most $O(k^2 \log k)$ variables. In the second case, we can solve the reduced system (kernel) by the algorithm of Theorem 1 in time $[O(k^2 \log k)]^{2k}[O(k^2 \log k)m]^{O(1)} = 2^{O(k \log k)}m^{O(1)}$. Thus, the total time is $2^{O(k \log k)}(nm)^{O(1)}$. $\qquad \square$

4   Max-r-Lin2-AA

The following theorem is the main result of this section.

Theorem 4. *Let S be an irreducible system and suppose that each equation contains at most r variables. Let $n \geq (k-1)r+1$ and let $w_{\min}$ be the minimum weight of an equation of S. Then, in time $O(m^{O(1)})$, we can find an assignment x^0 to variables of S such that $\varepsilon_S(x^0) \geq k \cdot w_{\min}$.*

We can easily prove Theorem 4 in the same way as we proved Theorem 2, but instead of Lemma 4, we use Lemma 5.

Lemma 5. *Let M be a set of vectors in $\mathbb{F}_2^n$ such that M contains a basis of $\mathbb{F}_2^n$. Suppose that each vector of M contains at most r non-zero coordinates. If $k \geq 1$ is an integer and $n \geq r(k-1)+1$, then in time $O(|M|^{O(1)})$, we can find a subset K of M of k vectors such that K is M-sum-free.*

Proof. Since the case of $k = 1$ is trivial, we may assume that $k \geq 2$. Let $\mathbf{1} = (1, \ldots, 1)$ be the vector in $\mathbb{F}_2^n$ in which every coordinate is 1. Note that $\mathbf{1} \notin M$. By our assumption M contains a basis B of $\mathbb{F}_2^n$ and we may write $\mathbf{1}$ as a sum of some vectors of B. This implies that $\mathbf{1}$ can be expressed as follows: $\mathbf{1} = v_1 + v_2 + \cdots + v_s$, where $\{v_1, \ldots, v_s\} \subseteq B$ and $v_1, \ldots, v_s$ are linearly independent, and we can find such an expression in polynomial time.

For each $v \in M \backslash \{v_1, \ldots, v_s\}$, consider the set $S_v = \{v, v_1, \ldots, v_s\}$. In polynomial time, we may check whether S_v is linearly independent. Consider two cases:

Case 1: S_v is linearly independent for each $v \in M \backslash \{v_1, \ldots, v_s\}$. Then $\{v_1, \ldots, v_s\}$ is M-sum-free (here we also use the fact that $\{v_1, \ldots, v_s\}$ is linearly independent). Since each v_i has at most r positive coordinates, we have $sr \geq n > r(k-1)$. Hence, $s > k-1$ implying that $s \geq k$. Thus, $\{v_1, \ldots, v_k\}$ is the required set K.

Case 2: S_v is linearly dependent for some $v \in M \backslash \{v_1, \ldots, v_s\}$. Then we can find (in polynomial time) $I \subseteq [s]$ such that $v = \sum_{i \in I} v_i$. Thus, we have a shorter expression for $\mathbf{1}$: $\mathbf{1} = v'_1 + v'_2 + \cdots + v'_{s'}$, where

12

$\{v'_1, \ldots, v'_{s'}\} = \{v\} \cup \{v_i : i \notin I\}$. Note that $\{v'_1, \ldots, v'_{s'}\}$ is linearly independent.

Since $s \leq n$ and Case 2 produces a shorter expression for $\mathbf{1}$, after at most n iterations of Case 2 we will arrive at Case 1. $\qquad\square$

Remark 4. *To see that the inequality $n \geq r(k-1) + 1$ in the theorem is best possible assume that $n = r(k-1)$ and consider a partition of $[n]$ into $k-1$ subsets $N_1, \ldots, N_{k-1}$, each of size r. Let S be the system consisting of subsystems S_p, $p \in [k-1]$, such that a subsystem S_p is comprised of equations $\prod_{i \in I} x_i = -1$ of weight 1 for every I such that $\emptyset \neq I \subseteq N_p$. Now assume without loss of generality that $N_p = [r]$. Observe that the assignment $(x_1, \ldots, x_r) = (1, \ldots, 1)$ falsifies all equations of S_p but by setting $x_j = -1$ for any $j \in [r]$ we satisfy the equation $x_j = -1$ and turn the remaining equations into pairs of the form $\prod_{i \in I} x_i = -1$ and $\prod_{i \in I} x_i = 1$. Thus, the maximum excess of S_p is 1 and the maximum excess of S is $k-1$.*

Remark 5. *It is easy to check that Theorem 4 holds when the weights of equations in S are real numbers, not necessarily integers, assuming unit cost real arithmetic (as in the model of Blum et al. [3]).*

5 Applications of Theorem 4

Theorem 5. *The problem* MAX-r-LIN2-AA$[k, r]$ *has a kernel with at most $(2k-1)r$ variables.*

Proof. Let T be the system of an instance of MAX-r-LIN2-AA$[k, r]$. After applying Rules 1 and 2 to T as long as possible, we obtain a new system S which is irreducible. Let n be the number of variables in S and observe that the number of variables in an equation in S is bounded by r (as in T). If $n \geq (2k-1)r + 1$, then, by Theorem 4 and Remark 1, S is a YES-instance of MAX-r-LIN2-AA$[k, r]$ and, hence, by Lemma 1, S and T are both YES-instances of MAX-r-LIN2-AA$[k, r]$. Otherwise $n \leq (2k-1)r$ and since S is irreducible, for the number m of equations in S, we have $m < 2^n \leq 2^{(2k-1)r}$. Thus, we have the required kernel. $\qquad\square$

Corollary 2. *The maximization problem* MAX-r-LIN2-AA *is in APX if restricted to $m = O(n)$ and the weight of each equation is bounded by a fixed constant.*

Proof. It follows from Theorem 4 and Remark 1 that the answer to MAX-r-LIN2-AA, as a decision problem, is YES as long as $2k \leq \lfloor (n + r - 1)/r \rfloor$. This implies approximation ratio at most $W/(2\lfloor (n + r - 1)/r \rfloor)$ which is bounded

by a constant provided $m = O(n)$ and the weight of each equation is bounded by a constant (then $W = O(n)$). □

The (parameterized) Boolean Max-r-Constraint Satisfaction Problem (MAX-r-CSP) generalizes MAXLIN2-AA$[k, r]$ as follows: We are given a set Φ of Boolean functions, each involving at most r variables, and a collection $\mathcal{F}$ of m Boolean functions, each $f \in \mathcal{F}$ being a member of Φ, and each acting on some subset of the n Boolean variables $x_1, x_2, \ldots, x_n$ (each $x_i \in \{-1, 1\}$). We are to decide whether there is a truth assignment to the n variables such that the total number of satisfied functions is at least $E + k$, where E is the average value of the number of satisfied functions. The parameters are k and r.

Using the bikernelization algorithm described in [1, 14] and our new kernel result, it easy to see that MAX-r-CSP with parameters k and r admits a bikernel with at most $(k2^{r+1} - 1)r$ variables. This result improves the corresponding result of Kim and Williams [31] ($n \leq kr(r + 1)2^r$).

The following result is essentially a corollary of Theorem 4 and Remark 5.

Theorem 6. *Let*
$$f(x) = \hat{f}(\emptyset) + \sum_{I \in \mathcal{F}} \hat{f}(I) \prod_{i \in I} x_i \tag{3}$$

be a pseudo-boolean function of degree r. Then

$$\max_x f(x) \geq \hat{f}(\emptyset) + \lfloor (\mathrm{rank}(A) + r - 1)/r \rfloor \cdot \min\{|\hat{f}(I)| : I \in \mathcal{F}\}, \tag{4}$$

where A is a $(0, 1)$-matrix with entries a_{ij} such that $a_{ij} = 1$ if and only if term j in (3) contains x_i and $\mathrm{rank}(A)$ is the rank of A over $\mathbb{F}_2$. One can find an assignment of values to x satisfying (4) in time $O((n|\mathcal{F}|)^{O(1)})$.

Proof. By Remark 2 the function $f(x) - \hat{f}(\emptyset) = \sum_{I \in \mathcal{F}} \hat{f}(I) \prod_{i \in I} x_i$ is the excess over the system $\prod_{i \in I} x_i = b_I$, $I \in \mathcal{F}$, where $b_I = +1$ if $\hat{f}(I) > 0$ and $b_I = -1$ if $\hat{f}(I) < 0$, with weights $|\hat{f}(I)|$. Clearly, Rule 1 will not change the system. Using Rule 2 we can replace the system by an equivalent one (by Lemma 1) with $\mathrm{rank}(A)$ variables. By Lemma 2, the new system is irreducible and we can now apply Theorem 4. By this theorem, Remark 2 and Remark 5, $\max_x f(x) \geq \hat{f}(\emptyset) + k^* \min\{|\hat{f}(I)| : I \in \mathcal{F}\}$, where k^* is the maximum value of k satisfying $\mathrm{rank}(A) \geq (k - 1)r + 1$. It remains to observe that $k^* = \lfloor (\mathrm{rank}(A) + r - 1)/r \rfloor$. □

We next give a new proof of the Edwards-Erdős bound, we need the following well-known and easy-to-prove lemma [9]. For a graph $G = (V, E)$, an incidence matrix is a $(0, 1)$-matrix with entries $m_{e,v}$, $e \in E$, $v \in V$ such that $m_{e,v} = 1$ if and only if v is incident to e.

Lemma 6. *The rank over $\mathbb{F}_2$ of an incidence matrix M of a connected graph equals $|V| - 1$.*

Theorem 7. *Let $G = (V, E)$ be a connected graph with n vertices and m edges. Then G contains a bipartite subgraph with at least $\frac{m}{2} + \frac{n-1}{4}$ edges. Such a subgraph can be found in polynomial time.*

Proof. Let $V = \{v_1, v_2, \ldots, v_n\}$ and let $c : V \to \{-1, 1\}$ be a 2-coloring of G. Observe that the maximum number of edges in a bipartite subgraph of G equals the maximum number of properly colored edges (i.e., edges whose end-vertices received different colors) over all 2-colorings of G. For an edge $e = v_i v_j \in E$ consider the following function $f_e(x) = \frac{1}{2}(1 - x_i x_j)$, where $x_i = c(v_i)$ and $x_j = c(v_j)$ and observe that $f_e(x) = 1$ if e is properly colored by c and $f_e(x) = 0$, otherwise. Thus, $f(x) = \sum_{e \in E} f_e(x)$ is the number of properly colored edges for c. We have $f(x) = \frac{m}{2} - \frac{1}{2} \sum_{e \in E} x_i x_j$. By Theorem 6, $\max_x f(x) \geq m/2 + \lfloor (\text{rank}(A) + 2 - 1)/2 \rfloor / 2$. Observe that matrix A in this bound is an incidence matrix of G and, thus, by Lemma 6 $\text{rank}(A) = n - 1$. Hence, $\max_x f(x) \geq \frac{m}{2} + \frac{1}{2} \lfloor \frac{n}{2} \rfloor \geq \frac{m}{2} + \frac{n-1}{4}$ as required. $\qquad \square$

This theorem can be extended to the BALANCED SUBGRAPH problem [4], where we are given a graph $G = (V, E)$ in which each edge is labeled either by $=$ or by $\neq$ and we are asked to find a 2-coloring of V such that the maximum number of edges is satisfied; an edge labeled by $=$ ($\neq$, resp.) is *satisfied* if and only if the colors of its end-vertices are the same (different, resp.).

Theorem 8. *Let $G = (V, E)$ be a connected graph with n vertices and m edges labeled by either $=$ or $\neq$. There is a 2-coloring of V that satisfies at least $\frac{m}{2} + \frac{n-1}{4}$ edges of G. Such a 2-coloring can be found in polynomial time.*

Proof. Let $V = \{v_1, v_2, \ldots, v_n\}$ and let $c : V \to \{-1, 1\}$ be a 2-coloring of G. Let $x_p = c(v_p)$, $p \in [n]$. For an edge $v_i v_j \in E$ we set $s_{ij} = 1$ if $v_i v_j$ is labeled by $\neq$ and $s_{ij} = -1$ if $v_i v_j$ is labeled by $=$. Then the function $\frac{1}{2} \sum_{v_i v_j \in E} (1 - s_{ij} x_i x_j)$ counts the number of edges satisfied by c. The rest of the proof is similar to that in the previous theorem. $\qquad \square$

6 Discussion and Open Problems

The kernels obtained in Theorems 3 and 5 are not of polynomial *size* as the number of equations in the kernels is not bounded by a polynomial in the parameter(s). The existence of polynomial-size kernels for MAXLIN2-AA[k] and MAX-r-LIN2-AA[k, r] remains an open question.

Perhaps the kernel obtained in Theorem 3 or the algorithm of Corollary 1 can be improved if we can find a structural characterization of irreducible

systems for which the maximum excess is less than $2k$ which would be of interest in itself.

Let F be a CNF formula with clauses $C_1, \ldots, C_m$ of sizes $r_1, \ldots, r_m$. Since the probability of C_i being satisfied by a random assignment is $1 - 2^{-r_i}$, the expected (average) number of satisfied clauses is $E = \sum_{i=1}^{m}(1 - 2^{-r_i})$. It is natural to consider the following parameterized problem MaxSat-AA[k]: decide whether there is a truth assignment that satisfies at least $E + k$ clauses. When there is a constant r such that $|C_i| \leq r$ for each $i = 1, \ldots, m$, MaxSat-AA[k] is denoted by Max-r-Sat-AA[k]. Mahajan *et al.* [33] asked what is the complexity of Max-r-Sat-AA[k] and Alon *et al.* [1] proved that it is fixed-parameter tractable [1]. Recently, Crowston *et al.* [16] determined the complexity of MaxSat-AA[k] by showing that MaxSat-AA[2] is NP-complete. Thus, MaxSat-AA[k] is not fixed-parameter tractable unless P=NP.

In a graph $G = (V, E)$, a bisection (X, Y) is a partition of V into sets X and Y such that $|X| \leq |Y| \leq |X| + 1$. The size of (X, Y) is the number of edges between X and Y. In Max Bisection, we are given a graph G with $n \geq 2$ vertices and m edges and asked to find a bisection of maximum size. It is not hard to see that $\lceil m/2 \rceil$ is a tight lower bound on the maximum size of a bisection of G. Gutin and Yeo [28] proved that Max Bisection parameterized above $\lceil m/2 \rceil$ has a kernel with $O(k^2)$ vertices. Mnich and Zenklusen [34] improved it to $O(k)$ vertices. Gutin and Yeo [28] also showed that $\lceil \frac{nm}{2(n-1)} \rceil$ is another tight lower bound on the maximum size of a bisection of G. Clearly, $\lceil \frac{nm}{2(n-1)} \rceil \geq \lceil m/2 \rceil$. Gutin and Yeo [28] left it as an open problem to determine the complexity of Max Bisection parameterized above $\lceil \frac{nm}{2(n-1)} \rceil$.

In Theorem 7, we gave another proof that every connected graph G with n vertices and m edges, has a bipartite subgraph with at least $\ell = \frac{m}{2} + \frac{n-1}{4}$ edges. Recently, Crowston *et al.* [17] solved an open question by showing that deciding whether G contains a bipartite subgraph with at least $\ell + k$ edges is fixed-parameter tractable and the problem admits a kernel with $O(k^5)$ vertices. Crowston *et al.* [15] improved this to $O(k^3)$ and showed that such a kernel exists for the more general problem of deciding whether a connected graph with edges labelled by $=$ and $\neq$, has a balanced subgraph with at least $\ell + k$ edges (for the bound, see Theorem 8). Can $O(k^3)$ be further improved?

Finally, the entire area of parameterizing above or below tight guaranteed bounds offers many challenging open problems in parameterized complexity.

References

[1] N. Alon, G. Gutin, E. J. Kim, S. Szeider, and A. Yeo, Solving MAX-r-SAT above a tight lower bound. Algorithmica 61(3): 638–655, 2011.

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

[3] E. Blum, F. Cucker, M. Shub, and S. Smale, Complexity and Real Computation. Springer, 1997.

[4] S. Böcker, F. Hüffner, A. Truss and M. Wahlström, A faster fixed-parameter approach to drawing binary tanglegrams. In IWPEC 2009, Lect. Notes Comput. Sci. 5917: 38–49, 2009.

[5] H. L. Bodlaender, R.G. Downey, M.R. Fellows, and D. Hermelin, On problems without polynomial kernels. J. Comput. Syst. Sci. 75(8): 423–434, 2009.

[6] H.L. Bodlaender, B.M.P. Jansen and S. Kratsch, Cross-Composition: A new technique for kernelization lower bounds. In STACS 2011, LIPICS 9:165–176, 2011.

[7] H. L. Bodlaender, S. Thomassé, and A. Yeo, Kernel bounds for disjoint cycles and disjoint paths. In ESA 2009, Lect. Notes Comput. Sci. 5757: 635–646, 2009.

[8] B. Bollobás and A. Scott, Better bounds for max cut. In Contemporary Combinatorics (B. Bollobás, ed.), Springer, 2002.

[9] J.A. Bondy and U.S.R. Murty, Graph Theory. Springer, 2008.

[10] E. Boros and P.L. Hammer, Pseudo-Boolean optimization. Discrete Applied Math. 123(1-3): 155–225, 2002.

[11] S.-S. Choi, K. Jung and J.H. Kim, Almost tight upper bound for finding Fourier coefficients of bounded pseudo-Boolean functions. J. Comput. Syst. Sci. 77(6): 1039–1053, 2011.

[12] R. Crowston, M. Fellows, G. Gutin, M. Jones, F. Rosamond, S. Thomassé and A. Yeo. Simultaneously Satisfying Linear Equations Over $\mathbb{F}_2$: MaxLin2 and Max-r-Lin2 Parameterized Above Average. In FSTTCS 2011, LIPICS 13, 229–240.

[13] R. Crowston, G. Gutin, and M. Jones, Note on Max Lin-2 above average. Inform. Proc. Lett. 110: 451–454, 2010.

[14] R. Crowston, G. Gutin, M. Jones, E. J. Kim, and I. Ruzsa. Systems of linear equations over $\mathbb{F}_2$ and problems parameterized above average. In SWAT 2010, Lect. Notes Comput. Sci. 6139 (2010), 164–175.

[15] R. Crowston, G. Gutin, M. Jones and G. Muciaccia. Maximum Balanced Subgraph Problem Parameterized Above Lower Bound. In COCOON 2013, Lect. Notes Comput. Sci. 7936 (2013), 434–445.

[16] R. Crowston, G. Gutin, M. Jones, V. Raman, S. Saurabh, and A. Yeo, Parameterized Complexity of MaxSat Above Average. Theor. Comput. Sci., DOI: 10.1016/j.tcs.2013.01.005.

[17] R. Crowston, M. Jones and M. Mnich, Max-Cut Parameterized above the Edwards-Erdős Bound. In ICALP 2012, Lect. Notes Comput. Sci. 7391: 242–253, 2012.

[18] M. Dom, D. Lokshtanov and S. Saurabh, Incrompressibility though Colors and IDs, In ICALP 2009, Part I, Lect. Notes Comput. Sci. 5555: 378–389, 2009.

[19] R. G. Downey and M. R. Fellows. Parameterized Complexity, Springer, 1999.

[20] C.S. Edwards, An improved lower bound for the number of edges in a largest bipartite subgraph. In Recent Advances in Graph Theory, Proc. 2nd Czecholslovac Symp., Academia, Prague, 1995, 167–181.

[21] P. Erdős, On some extremal problems in graph theory. Israel J. Math. 3: 113–116, 1965.

[22] P. Erdős, A. Gyárfás and Y. Kohayakawa, The size of the largest bipartite subgraphs. Discrete Math. 117: 267–271, 1997.

[23] M. R. Fellows, B. M. P. Jansen and F. A. Rosamond. Towards Full Multivariate Algorithmics: Parameter Ecology and the Deconstruction of Computational Complexity, *European J. Combinatorics* 34 (2013), 541–566.

[24] J. Flum and M. Grohe, Parameterized Complexity Theory, Springer, 2006.

[25] F.V. Fomin, D. Lokshtanov, N. Misra, G. Philip and S. Saurabh, Hitting forbidden minors: Approximation and kernelization. In STACS 2011, LIPICS 9:189–200, 2011.

[26] G. Gutin, L. van Iersel, M. Mnich, and A. Yeo, All ternary permutation constraint satisfaction problems parameterized above average have kernels with quadratic number of vertices. J. Comput. Syst. Sci. 78: 151–163, 2012.

[27] G. Gutin, E. J. Kim, S. Szeider, and A. Yeo. A probabilistic approach to problems parameterized above or below tight bounds. J. Comput. Sys. Sci. 77: 422–429, 2011.

[28] G. Gutin and A. Yeo, Note on Maximal Bisection above Tight Lower Bound. Inform. Proc. Lett. 110: 966–969, 2010.

[29] J. Håstad, Some optimal inapproximability results. J. ACM 48: 798–859, 2001.

[30] J. Håstad and S. Venkatesh, On the advantage over a random assignment. Random Structures & Algorithms 25(2):117–149, 2004.

[31] E.J. Kim and R. Williams, Improved Parameterized Algorithms for above Average Constraint Satisfaction. In IPEC 2011, Lect. Notes Comput. Sci. 7112: 118–131, 2012.

[32] B. Korte and J. Vygen, Combinatorial Optimization: theory and algorithms, 3rd Edition, Springer, 2006.

[33] M. Mahajan, V. Raman, and S. Sikdar. Parameterizing above or below guaranteed values. J. Computer System Sciences, 75(2):137–153, 2009. Preliminary version appeared in IWPEC 2006, *Lect. Notes Comput. Sci.* 4169:38–49, 2006.

[34] M. Mnich and R. Zenklusen, Bisections above Tight Lower Bounds. In WG'2012, Lect. Notes Comput. Sci. 7551:184–193, 2012.

[35] R. Niedermeier. Invitation to Fixed-Parameter Algorithms. Oxford University Press, 2006.

[36] R. Niedermeier, Reflections on Multivariate Algorithmics and Problem Parameterization. In STACS 2010, LIPICS 5:17–32, 2010.

[37] R. O'Donnell, Some topics in analysis of Boolean functions. Technical report, ECCC Report TR08-055, 2008. Paper for invited talk at STOC'08, www.eccc.uni-trier.de/eccc-reports/2008/TR08-055/ .