Complexity and Approximability of Parameterized MAX-CSPs

Holger Dell[1], Eun Jung Kim[2], Michael Lampis[3], Valia Mitsou[4], and Tobias Mömke[5]

1   Saarland University and Cluster of Excellence, Saarbrücken, Germany
   Simons Institute for the Theory of Computing, Berkeley, US
   `hdell@mmci.uni-saarland.de`
2   Université Paris Dauphine, France
   Simons Institute for the Theory of Computing, Berkeley, US
   `eunjungkim78@gmail.com`
3   Université Paris Dauphine, France
   `michail.lampis@dauphine.fr`
4   SZTAKI, Hungarian Academy of Sciences, Budapest, Hungary[*]
   `vmitsou@gradcenter.cuny.edu`
5   Saarland University, Saarbrücken, Germany[†]
   `moemke@cs.uni-saarland.de`

Abstract

We study the optimization version of constraint satisfaction problems (Max-CSPs) in the framework of parameterized complexity; the goal is to compute the maximum fraction of constraints that can be satisfied simultaneously. In standard CSPs, we want to decide whether this fraction equals one. The parameters we investigate are structural measures, such as the treewidth or the clique-width of the variable–constraint incidence graph of the CSP instance.

We consider Max-CSPs with the constraint types AND, OR, PARITY, and MAJORITY, and with various parameters k. We attempt to fully classify them into the following three cases:

1. The exact optimum can be computed in FPT-time.
2. It is W[1]-hard to compute the exact optimum, but there is a randomized FPT approximation scheme (FPT-AS), which computes a $(1 - \epsilon)$-approximation in time $f(k, \epsilon) \cdot \text{poly}(n)$.
3. There is no FPT-AS unless FPT=W[1].

For the corresponding standard CSPs, we establish FPT vs. W[1]-hardness results.

1   Introduction

Constraint Satisfaction Problems (CSPs) play a central role in almost all branches of theoretical computer science. Starting from CNF-SAT, the prototypical NP-complete problem, the computational complexity of CSPs has been widely studied from various points of view.

10th International Symposium on Parameterized and Exact Computation (IPEC 2015).
Editors: Thore Husfeldt and Iyad Kanj; pp. 294–306
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

In this paper we focus on two aspects of CSP complexity which, though extremely well-investigated, have mostly been considered separately so far in the literature: parameterized complexity and approximability. We study four standard predicates and contribute some of the first results indicating that the point of view of approximability considerably enriches the parameterized complexity landscape of CSPs.

1.1 Parameterized CSPs

The vast majority of interesting CSPs are NP-hard [21, 12]. This has motivated the study of such problems from a parameterized complexity point of view, and indeed this topic has attracted considerable attention in the literature [9, 26, 6, 18, 8, 25]. We refer the reader to [20] where an extensive classification of CSP problems for a large range of parameters is given. In this paper we focus on *structurally* parameterized CSPs, that is, we consider CSPs where the parameter is some measure of the structure of the input instance. The central idea behind this approach is to represent the structure of the CSP using a (hyper-)graph and leverage the powerful tools commonly applied to parameterized graph problems (such as tree decompositions) to solve the CSP.

The typical goal of this line of research is to find the most general parameterization of a CSP that still remains fixed-parameter tractable (FPT). To give a concrete example for a very well-known CSP, CNF-SAT is FPT when parameterized by the treewidth of its incidence graph[1] [23] but it is W-hard for more general parameters such as clique-width [16], or even the more restricted modular treewidth [17]. General (boolean) CSP on the other hand, where the description of each constraint is part of the input is known to be a harder problem: it is already W[1]-hard parameterized by the incidence treewidth, but FPT parameterized by the treewidth of the primal graph [24]. Thus, parameterized investigations aim to locate the boundary where a CSP jumps from being FPT to being W-hard. It is of course a natural question how we can deal with the W-hard cases of a CSP once they are identified.

1.2 Approximation

CSPs also play a central role in the theory of (polynomial-time) approximation algorithms [27, 13, 2]. In this context we typically consider a CSP as an optimization problem (MAX-CSP) where the goal is to find an assignment to the variables that satisfies as many of the constraints as possible. Unfortunately, essentially all non-trivial CSPs are hard to approximate (APX-hard) from this point of view [4, 12], even those where deciding if an assignment can satisfy all constraints is in P (e.g. 2CNF-SAT or HORN SAT). Thus, research in this area typically focuses on discovering exactly the best approximation ratio that can be achieved in polynomial time. Amazingly, for many natural CSPs this happens to be exactly the ratio achieved by a completely random assignment [11]. This motivates the question of whether we can find natural cases where non-trivial efficient approximations are possible.

1.3 Results

In this paper we consider four different types of CSPs where the constraints are respectively OR, AND, PARITY and MAJORITY functions. Our approach follows, for the most part, the standard parameterized complexity script: we consider the input instance's incidence graph and try to determine the complexity of the CSP when parameterized by various graph

---

[1] See the next section for a definition of incidence graphs

widths. The new ingredient in our approach is that, in addition to trying to determine which parameters make a CSP FPT or W-hard, we also ask if the optimization versions of W-hard cases can be well-approximated. We believe that this is a question of special interest since, as it turns out, there are CSPs for which W-hardness can be (almost) circumvented using approximation, and others which are inapproximable.

More specifically, our results are as follows: for OR constraints, which corresponds to the standard CNF-SAT (MAX-CNF-SAT) problem, we present a new hardness proof establishing that deciding a formula's satisfiability is W-hard even if parameterized by the incidence graph's neighborhood diversity. Neighborhood diversity is a parameter much more restricted than modular treewidth (already a restriction of clique-width) [14], for which the strongest previously known W[1]-hardness result was known [17]. We complement this negative result with a strong positive approximation result: there exists a randomized FPT Approximation Scheme (FPT-AS)[2] for MAX-CNF-SAT parameterized by clique-width, that is, an algorithm which for all $\epsilon > 0$ runs in time $f(k, \epsilon)n^{O(1)}$ and returns an assignment satisfying $(1 - \epsilon)\text{OPT}$ clauses. Thus, even though we establish that solving CNF-SAT exactly is W-hard even for extremely restricted dense graph parameters, MAX-CNF-SAT is well-approximable even in the quite general case of clique-width. To the best of our knowledge, this is the first approximation result of this type for a W-hard MAX-CSP problem.

Recalling that MAX-CNF-SAT is FPT parameterized by the treewidth of the incidence graph, we consider other problems for which the jump from treewidth to clique-width could have interesting complexity consequences. We show that MAX-DNF-SAT and MAX-PARITY, which are FPT parameterized by treewidth, exhibit two wildly different behaviors. On the one hand, the problem of maximizing the largest possible number of satisfied PARITY constraints remains FPT even for dense parameters such as clique-width. On the other hand, by modifying our reduction for CNF-SAT, we are able to show not only that maximizing the number of satisfied AND constraints is W[1]-hard parameterized by neighborhood diversity, but also that this problem cannot even admit an FPT-AS (like MAX-CNF-SAT), unless W[1]=FPT. We recall that PARITY and AND constraints are similar in other aspects: for example, for both we can decide in polynomial time if an assignment satisfying all constraints exists.

Finally, we consider CSPs with MAJORITY constraints, that is, constraints which are satisfied if at least half their literals are true. We give a reduction establishing that this is an interesting case of a natural constraint type for which deciding satisfiability is already W[1]-hard parameterized by treewidth (we actually show W[1]-hardness for the more restricted case of incidence feedback vertex set) and by neighborhood diversity. We complement this negative result with two algorithmic results: first, we show that the corresponding MAX-CSP is FPT parameterized by incidence vertex cover. Then, we use this algorithm as a sub-routine to obtain an FPT-AS for incidence feedback vertex set. Both of these algorithmic results also apply to the more general case of THRESHOLD constraints. We leave it as an interesting open problem to examine if the approximation algorithm for feedback vertex set can be extended to treewidth.

2 Preliminaries

A Boolean CSP ψ is defined as a set $\{C_1, \ldots, C_m\}$ of m constraints over a set $X(\psi) = \{x_1, \ldots, x_n\}$ of n variables and their negations. Each constraint C_i is regarded as a function

---

[2] We follow the standard definition of FPT-AS given in [15].

$$\text{tw}^* \longleftarrow \text{fvs}^*$$
$$\text{cw}^* \longleftarrow \text{nd}^* \longleftarrow \text{vc}^*$$
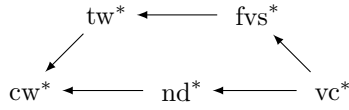
Figure 1 The structural parameters we study and their relationships. For example, the arrow between cw^* and tw^* means that if the treewidth is bounded, then the clique-width is bounded as well — more precisely, there is a monotone computable function $f : \mathbf{N} \to \mathbf{N}$ so that $\text{cw}^* \leq f(\text{tw}^*)$. On the other hand, tw^* and nd^* as well as fvs^* and nd^* cannot be bounded by each other in general.

MAX-CNF-SAT

MAX-MAJORITY-CSP

MAX-DNF-SAT

MAX-PARITY-CSP
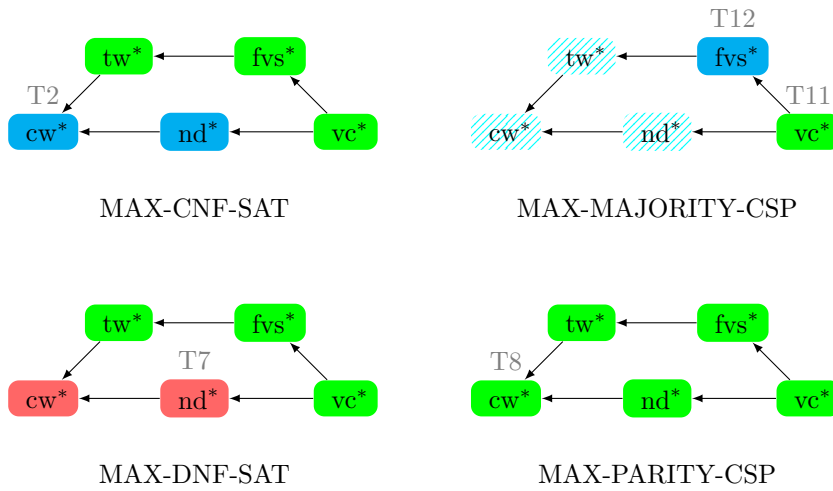
Figure 2 The parameterized complexity status of MAX-CSP problems. The *gray* labels above the boxes indicate the theorem in which we establish the result; previously known results are displayed without reference. *Red* means that the problem is W[1]-hard to compute exactly, and there is no FPT-AS unless FPT = W[1]. *Blue* means that the problem is W[1]-hard to compute exactly, and there is an FPT-AS. *Green* means that the problem is FPT to compute exactly. The *blue/white* stripes mean that it's W[1]-hard to compute exactly, and it's open whether there is an FPT-AS.

CNF-SAT
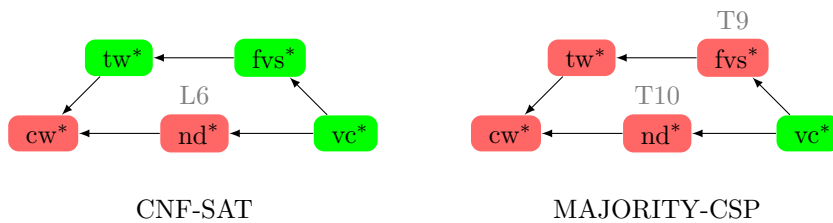
MAJORITY-CSP

Figure 3 The parameterized complexity status of CNF-SAT and MAJORITY-CSP. Recall that DNF-SAT and PARITY-CSP are polynomial-time computable. *Red* means that the problem is W[1]-hard and *green* means that the problem is FPT.

of literals (positive or negative appearances of variables) mapped to the set $\{0, 1\}$, where literals can take the values 0 or 1. Furthermore, we define $|C_j|$ to denote the arity of constraint C_j (the number of literals that occur in C) and $|\psi| = m$ the number of constraints in ψ. For simplicity, we also write $l_i \in C_j$ for a literal l_i and a constraint C_j if l_i appears in C_j.

We will be dealing with Boolean constraint satisfaction problems for four well-studied Boolean functions: OR constraints, AND constraints, PARITY (or XOR) constraints and MAJORITY constraints. We say that an assignment $t : X \to \{0, 1\}$ satisfies a constraint C of

- OR, if $\exists l_i \in C$, $t(l_i) = 1$;
- AND, if $\forall l_i \in C$, $t(l_i) = 1$;
- PARITY, if it satisfies some equation $\Sigma_{l_i \in C} t(l_i) = b$ (for $b \in \{0, 1\}$) modulo 2;
- MAJORITY, if at least $\lceil |C|/2 \rceil$ literals in C are set to 1. More generally, we may consider THRESHOLD constraints, where a certain threshold number of literals must be set to true to satisfy the constraint.

Let $\mathrm{occ}(\psi) = \sum_{C \in \psi} |C|$ be the total number of variable occurrences in ψ, that is, the total size of the formula. For a variable x, we write ψ_x for the set of all constraints $C \in \psi$ where x occurs either positive or negative; for the functions we consider without loss of generality, no clause contains both literals. Thus, the total number of occurrences of a variable x is equal to $|\psi_x|$.

We are dealing also with MAX-CSPs, where given a set of constraints ψ, we are interested in finding an assignment to the variables that maximizes the number of satisfied constraints. The natural decision version of this problem is, given a target k, decide whether there exists an assignment that satisfies at least k constraints. Clearly, the problem where we want to decide whether we can satisfy all the constraints is a special case of the above decision problem since we can set $k = m$, but in some cases we consider this simpler decision version, particularly when we want to show hardness.

In the case of OR constraints, the CSP and MAX-CSP problems correspond to the more widely known CNF-SAT and MAX-CNF-SAT problems. In this case we call the constraints *clauses*. When the constraint function is AND, the MAX-CSP problem is called MAX-DNF-SAT. In that case, the constraints are called *terms*. The problem MAX-PARITY is also known as MAX-LIN-2 (satisfy a maximum number of given linear equations modulo 2).

For a CSP ψ, the incidence graph G_ψ^* is defined as the bipartite graph where we construct one vertex v_i for each (unsigned) variable x_i and one vertex u_j for each constraint C_j and connect v_i with u_j if $x_i \in C_j$.

We are interested in parameterizations of the incidence graph $\mathrm{p}(G_\psi^*)$ (or simply p^* if G_ψ^* is clear from the context), where p is a structural parameter of G_ψ^*. We are mostly interested in the two most widely studied graph parameters, treewidth tw^* and clique-width cw^*. We refer the reader to standard parameterized complexity textbooks for the definitions, as well as the definitions of standard parameterized complexity terminology used in this paper [5].

3 CNF-SAT and MAX-CNF-SAT

In this section, we consider one of the most fundamental problems in computer science: the satisfiability problem for CNF formulas, which can be viewed as a constraint satisfaction problem where the only allowed constraints are clauses, that is, ORs of literals. An optimal solution for MAX-CNF-SAT can be computed in FPT when parameterized by the treewidth tw^* of the incidence graph [1], and hence CNF-SAT can be solved in the same time. When parameterized by the clique-width cw^* of the incidence graph, all known exact

algorithms for CNF-SAT and MAX-CNF-SAT run in XP time [22, 19]. Moreover, we don't expect these problems to be in FPT since they are both W[1]-hard parameterized by cw* [17].

In Section 3.1, we construct an approximation scheme for MAX-CNF-SAT that runs in FPT time. Intuitively, our algorithm works as follows: given a formula ϕ with 'small' incidence clique-width, we first examine the formula to see if it contains many or few 'large' clauses. If the formula contains relatively few large clauses, then we simply disregard them. We then know that the incidence graph does not contain 'large' bi-cliques, so by a theorem of Gurski and Wanke [10] the remaining formula has small treewidth and we can solve the problem. If on the other hand the original formula contains a large number of large clauses, then we observe that we can rely on a random assignment to satisfy almost everything.

In Section 3.2, we explore a class of CSP instances that is smaller than the class of bounded incidence clique-width instances; our goal is to understand which incidence graph parameter is responsible for the transition from FPT to W[1]. To this end, we have to look for a graph parameter that is bounded by a function of cw* (where it's hard) but can leave the tw* unbounded (where it's FPT). In fact, [17] shows that the problem is W[1]-hard parameterized by the modular treewidth mtw* of the incidence graph, which lies between cw* and tw*.[3] We study the incidence *neighborhood diversity* nd*, which is incomparable to tw*; however, mtw* is bounded when nd* is. We prove that CNF-SAT remains W[1]-hard parameterized by nd*.

▶ **Definition 1.** A graph $G(V, E)$ has neighborhood diversity $\mathrm{nd}(G) = k$ if we can partition V into k sets $V_1, \ldots, V_k$ such that, for all $v \in V$ and all $i \in \{1, \ldots, k\}$, either v is adjacent to every vertex in V_i or it is adjacent to none of them.

In other words, $\mathrm{nd}(G) = k$ if V can be partitioned into k modules that are either cliques or independent sets.

Formulas whose incidence graph has neighborhood diversity at most k seem very restrictive: there are only at most k variable- and clause-types, where all variables of the same type belong to the same clauses and all clauses of the same type involve the same variables. This class of formulas is a subset of formulas with mtw* $\leq k$ because contracting all modules leaves a graph of *order* at most k, which trivially has treewidth at most k.

3.1 Approximation Algorithm parameterized by clique-width

▶ **Theorem 2.** *There is a randomized algorithm that, given a CNF formula ψ with n variables, m clauses, and incidence clique-width cw, runs in time $f(\epsilon, \mathrm{cw}) \cdot \mathrm{poly}(n + m)$, and outputs a truth assignment that satisfies at least $(1 - \epsilon) \cdot \mathrm{OPT}$ clauses in expectation.*

We formulate the following basic lemma about probability distributions.

▶ **Lemma 3.** *For all $\epsilon, L > 0$ there is a $c = c(\epsilon, L) > 0$ such that all $c' \geq c$ and all sequences $p_1, \ldots, p_{c'} \geq 0$ with $\sum_{i=1}^{c'} p_i \leq 1$ have an index $d \leq c/L$ with the property*

$$p_{[d, L \cdot d]} \doteq \sum_{j=d}^{L \cdot d} p_j < \epsilon.$$

---

[3] A graph of *bounded modular treewidth* is a graph of bounded treewidth after merging modules into a single vertex, where a module is a set of vertices with same neighborhood outside of the set. In fact, the reduction in [17] constructs a formula whose incidence graph has small feedback vertex set after contracting modules.

Proof. Let $\epsilon, L > 0$. We set $c = c(\epsilon, L)$ below. Assume for contradiction that $p_{[d, L \cdot d]} \geq \epsilon$ holds for all $d \in [1, c/L]$. If there are $1/\epsilon + 1$ disjoint intervals $[a_1, L \cdot a_1], \ldots, [a_{1/\epsilon+1}, L \cdot a_{1/\epsilon+1}] \subseteq [1, c]$, then we arrive at a contradiction with the fact that the p_i's are non-negative and sum to at most one. Clearly there exists a constant $c = c(\epsilon, L)$ such that $1/\epsilon + 1$ disjoint intervals of the form $[a, La]$ fit into $[1, c]$. This proves the claim. ◀

For an arbitrary given $\epsilon > 0$, we fix $L = \epsilon^{-4}$. We use Lemma 3 as follows: For a CNF formula ψ, we define p_i as the fraction of clauses of size i, that is,

$$p_i \doteq \frac{\left| \left\{ C \in \psi \mid |C| = i \right\} \right|}{|\psi|}.$$

Then Lemma 3 gives us a number $d \leq c(\epsilon)$ such that the total fraction of clauses whose size is between d and $\epsilon^{-4}d$ is bounded by ϵ. It is now natural to partition all clauses into short, medium, and long clauses. More precisely, we define $\psi = \psi^{<d} \,\dot\cup\, \psi^{[d,D]} \,\dot\cup\, \psi^{>D}$ for $D = \epsilon^{-4}d$ as follows:

$$\psi^{<d} \doteq \left\{ C \in \psi \mid |C| < d \right\},$$
$$\psi^{[d,D]} \doteq \left\{ C \in \psi \mid d \leq |C| \leq D \right\}, \text{and}$$
$$\psi^{>D} \doteq \left\{ C \in \psi \mid |C| > D \right\}.$$

An immediate corollary to Lemma 3 is thus that we can choose $d \leq c(\epsilon)$ in such a way that $|\psi^{[d,D]}| \leq \epsilon|\psi|$.

▶ **Corollary 4.** *For all $\epsilon > 0$ there is some $c = c(\epsilon) > 0$ such that all CNF formulas ψ have some $d = d(\epsilon) \in [1, c]$ with $|\psi^{[d, \epsilon^{-4}d]}| \leq \epsilon \cdot |\psi|$.*

If $\psi^{[d,D]} = \emptyset$ holds for $D = \epsilon^{-4}d$ and $d \in [1, c(\epsilon)]$, we say that ψ is ϵ-*well separated*. We call ψ ϵ'-*balanced* if, in addition, we have $|\psi^{<d}| \geq \epsilon'm$ and $|\psi^{>D}| \geq \epsilon'm$.

▶ **Lemma 5.** *Let ψ be an ϵ-well separated formula (and thus $V = V(\psi^{<d}) \cup V(\psi^{>D})$).*
    Then, for each subset $\hat\psi \subseteq \psi^{>D}$ with $|\hat\psi| > \epsilon^2 m$, there is a variable y such that $|\psi_y^{<d}| \leq \epsilon^2|\hat\psi_y|$.

That is, for every set $\hat\psi$ that contains a significant fraction of long clauses, there is a variable that occurs $|\hat\psi_y|$ times in $\hat\psi$, but only at most an ϵ^2-fraction of that in the short clauses.

Proof. Let $\hat\psi \subseteq \psi^{>D}$ with $|\hat\psi| > \epsilon^2 m$. Note that the total number of literal occurrences in $\hat\psi$ is $\operatorname{occ}(\hat\psi) > D \cdot \epsilon^2 \cdot m = \epsilon^{-2}dm$. In contrast, $\operatorname{occ}(\psi^{<d}) < dm$. Now suppose that there was no variable y with the claimed properties, that is, suppose that every variable y satisfies $|\psi_y^{<d}| > \epsilon^2|\hat\psi_y|$. Then the total number of variable occurrences in $\psi^{<d}$ can be bounded from below as follows:

$$\operatorname{occ}(\psi^{<d}) = \sum_y |\psi_y^{<d}| > \sum_y \epsilon^2|\hat\psi_y| = \epsilon^2 \operatorname{occ}(\hat\psi) > d \cdot m.$$

This yields a contradiction and thus proves the claim. ◀

Proof of Theorem 2. The algorithm A works as follows. Let $\epsilon' = \epsilon^2$, and we assume w.l.o.g. that $\epsilon < 1/8$. Given a CNF formula ϕ, we compute an ϵ'-well separated formula ψ by dropping all clauses in $\phi^{[d,D]}$. Corollary 4 guarantees that the fraction of deleted clauses is

bounded by ϵ'. If ψ is not $\epsilon/2$-balanced, we discard the smaller side (with fewer clauses) and only handle the larger one: If $\psi^{<d}$ is the larger side, we compute an optimal assignment for $\psi^{<d}$ in FPT time, by using the result of Gurski and Wanke [10]. This way the total number of unsatisfied clauses is at most $\epsilon m/2$, and together with the unsatisfied clauses due to applying Corollary 4, the total number of unsatisfied clauses is smaller than ϵm. Since OPT $> m/2$, we get the approximation guarantee.

If $\psi^{>D}$ is the larger side, we use a random assignment. This way, at most $\epsilon m/2$ clauses from $\psi^{<d}$ are violated, and in expectation at most a 2^{-D} fraction of clauses from $\psi^{>D}$ are violated. Since 2^{-D} is smaller than $\epsilon/4$, we conclude that – together with unsatisfied clauses due to applying Corollary 4 – at least $(1-\epsilon)m$ clauses are satisfied in expectation.

This finishes the analysis of unbalanced formulas, and in the remaining proof we may assume that ψ is $\epsilon/2$-balanced. To handle this case, we determine a set of variables Y such that

- there are at most $\epsilon m/4$ short clauses with variables from Y and
- there are at most $\epsilon^2 m$ long clauses that contain $\leq 1/\epsilon$ variables from Y.

Before we construct Y, let us verify that the properties of Y imply the correctness of the theorem. Our algorithm computes a satisfying assignment of the short clauses without variables from Y, again using the result of Gurski and Wanke [10]. Afterwards it assigns values uniformly at random to the variables in Y.

There are at most $\epsilon' m = \epsilon^2 m$ unsatisfied clauses due to applying Corollary 4, $\epsilon m/4$ short clauses clauses that we did not consider when satisfying clauses from $\psi^{<d}$, and $\epsilon^2 m$ clauses from $\psi^{>D}$ that we did not consider in the random assignment. Additionally, in expectation there are less than $2^{-1/\epsilon} m$ clauses left unsatisfied from the remaining $|\psi^{>D}| - \epsilon^2 m$ clauses from $\psi^{>D}$. Since we assumed that $\epsilon < 1/8$, the theorem follows.

To construct the set Y, we iteratively apply Lemma 5 with the parameter $\epsilon/4$. Initially, we set $\hat{\psi} = \psi^{>D}$. In each iteration, we identify a variable y according to the lemma and add the variable to Y. In the subsequent iterations, we mark y to be inactive and handle it as if it was not contained in any clause. Whenever we identify a clause C that has at least $1/\epsilon$ inactive variables (i.e., variables from Y), we remove C from $\hat{\psi}$. We continue this process until $|\hat{\psi}| \leq \epsilon^2$. Note that applying Lemma 5 for $\epsilon/4$ but having an ϵ'-well separated formula ensures that at all times, all clauses in $\hat{\psi}$ have sufficiently many literals to apply Lemma 5. Therefore the process terminates and the generated set Y has the aimed-for properties since $|Y| \leq m/\epsilon$. ◀

3.2 Hardness parameterized by neighborhood diversity

A *constraint* on r variables is a relation $R \subseteq \{0,1\}^r$. We define the unary constraints $U_0 = \{0\}$ and $U_1 = \{1\}$, which corresponds to clauses $(\neg x)$ and (x), respectively. We define the equality $=$ and disequality $\neq$ constraints on two groups of Boolean variables $x = x_1 \ldots x_n$ and $y = y_1 \ldots y_n$ in infix notation in the usual way: For an assignment a to the x- and y-variables, we say that $a \models x = y$ if and only if, for all $i \in [n]$, we have $a(x_i) = a(y_i)$, that is, the assignment sets x_i to the same value as y_i; as usual, $x \neq y$ is interpreted as the negation of $x = y$.

▶ **Lemma 6.** *CNF-SAT parameterized by* nd^* *is* W[1]-*hard, where* nd^* *is the neighborhood diversity of the input's incidence graph.*

Proof. We devise an FPT-reduction from k-Multicolored Clique to CNF-SAT. Given a k-partite graph G, whose parts $V_1, \ldots, V_k$ all have the same size n, we construct k groups of

variables $x_1, \ldots, x_k$, which together are supposed to represent a k-clique in G, should one exist. Each group x_i consists of $\log n$ Boolean variables and represents the supposed clique's vertex in the part V_i. Without loss of generality, we assume that $\log n$ is an integer.

Starting from the empty CNF formula, we construct a formula ϕ on the x-variables as follows. First choose, for each $i \in [k]$, an arbitrary bijection $\mathrm{bin}_i : V_i \to \{0,1\}^{\log n}$ that maps any vertex $u \in V_i$ to its binary representation $\mathrm{bin}(u)$; for convenience, we drop the index i. For each $i, j \in [k]$ with $i < j$, and for each non-edge $(u,v) \notin E(V_i, V_j)$ between V_i and V_j, we add the following constraint $C_{i,j,u,v}$ to ϕ:

$$x_i x_j \neq \mathrm{bin}(u)\mathrm{bin}(v) \,.$$

Clearly, this constraint excludes exactly one of the $2^{2\log n}$ possible assignments to $x_i x_j$, and so it can be written as an OR of literals of the x-variables. In the end, ϕ is a CNF formula with $|E(G)|$ clauses.

For the completeness of the reduction, let $v_i \in V_i$ for all $i \in [k]$ be such that $v_1, \ldots, v_k$ is a clique in G. For all $i \in [k]$, set $x_i = \mathrm{bin}(v_i)$. This assignment satisfies all constraints: for all $(u,v) \notin E(V_i, V_j)$, we have that $\mathrm{bin}(v_i)\mathrm{bin}(v_j) \neq \mathrm{bin}(u)\mathrm{bin}(v)$ because (v_i, v_j) is an edge and (u,v) is not, and bin is a bijection.

For the soundness of the reduction, let $a_1, \ldots, a_k \in \{0,1\}^{k\log n}$ be a satisfying assignment of ϕ. For each $i \in [k]$, let v_i be the unique vertex in V_i for which $\mathrm{bin}(v_i) = a_i$. Let $i, j \in [k]$ with $i < j$. Since the assignment satisfies all constraints of ϕ, it must be the case that (v_i, v_j) is an edge in G. For if it was a non-edge, its corresponding constraint in ϕ would have excluded the assignment $a_i a_j$ for $x_i x_j$. Hence $v_1, \ldots, v_k$ is a clique in G.

It remains to argue that the neighborhood diversity of the incidence graph of ϕ is at most $k + \binom{k}{2}$. The modules of the incidence graph are the variable group x_h for each $h \in [k]$ and the clause group $\{C_{i,j,u,v}\}$ for each $i, j \in [k]$ with $i < j$. Indeed, the incidence graph between x_h and $C_{i,j,*,*}$ is a bipartite clique if $h \in \{i,j\}$, and otherwise it is an independent set.

We constructed an FPT-reduction from the W[1]-complete problem Multicolored Clique to CNF-SAT parameterized by nd^*, which finishes the proof of the theorem. ◄

4   From Treewidth to Clique-width

In the previous section, we have seen that MAX-CNF-SAT is fixed-parameter tractable when parameterized by tw^*, which is a sparse graph parameter, and it is W[1]-hard to compute exactly and has an FPT-AS when parameterized by nd^*, which is a dense graph parameter. In this section we observe that the transition from sparse to dense parameters has different effects on the complexity of MAX-CSP, depending on which types of constraints are allowed.

By modifying our reduction for CNF-SAT we show that MAX-DNF-SAT, the problem of maximizing the number of satisfied AND constraints is W[1]-hard parameterized nd^*. Furthermore, because the maximum number of constraints that could be satisfied in our reduction is also bounded by some function of the parameter, we show that the problem does not have an FPT-AS unless FPT=W[1]. Thus, while MAX-DNF-SAT is FPT parameterized by tw^*, it does not even appear to have an FPT approximation scheme when parameterized by nd^*.

▶ **Theorem 7.** *Suppose that there exists an* FPT-AS *which, given $\epsilon > 0$ and an instance ϕ of* MAX-DNF-SAT, *computes a $(1-\epsilon)$-approximate solution and runs in time $f(\mathrm{nd}^*, \epsilon)\cdot\mathrm{poly}(n)$, where nd^* is the neighborhood diversity of the incidence graph of ϕ. Then* FPT = W[1].

The proof is similar that of Theorem 6 and deferred to the full version.

When parameterized by tw*, MAX-CNF-SAT and MAX-DNF-SAT are both FPT, and when parameterized by a dense graph parameter, such as nd*, the former problem is hard but approximable while the latter problem is hard even to approximate. We next consider natural constraint types where the corresponding CSPs stay FPT both for sparse as well as dense incidence graph parameters. MAX-PARITY wants to find an assignment that satisfies the maximum number of linear equations modulo two. While deciding whether there is an assignment that satisfies all equations is in P (by Gauss elimination), the maximization version is a typical APX-hard problem [11]. Here we show that computing the optimal solution of MAX-PARITY is FPT, regardless of whether the parameter is the treewidth or the clique-width of the incidence graph. Our intuition for why MAX-PARITY appears to be so much easier than CNF-SAT is that negations are (almost) irrelevant, and so the incidence graph seems to capture most of the structure relevant to the complexity of the CSP instance.

▶ **Theorem 8.** *Given an instance ϕ for* MAX-PARITY*, we can find an optimal solution in time $f(\mathrm{cw}^*)|\phi|^{O(1)}$, where cw^* is the clique-width of the incidence graph of ϕ.*

The proof relies on the meta-theorem of [3] and appears in the full version.

5 Majority and Threshold CSPs

In this section we describe our results for CSPs where each constraint is a MAJORITY constraint. Here a constraint MAJORITY($\{\ell_1, \ldots, \ell_d\}$) stipulates that at least $d/2$ of the literals ℓ_i are set to true. We denote the resulting CSP problem with MAJORITY, and the resulting MAX-CSPwith MAX-MAJORITY.

5.1 Hardness of exact algorithms

We parameterize MAJORITY by the size fvs* of the smallest feedback vertex set, or by the neighborhood diversity nd* of the instance's incidence graph. These parameterized problems turn out to be hard, even for the special case of MAJORITY constraints. Thus, neither dense nor sparse incidence graph parameters appear to put the problem in FPT.

▶ **Theorem 9.** MAJORITY *parameterized by the incidence feedback vertex set number* fvs* *is* W[1]*-hard.*

▶ **Theorem 10.** MAJORITY *parameterized by the incidence neighborhood diversity* nd* *is* W[1]*-hard.*

While the former theorem is proved by a technical reduction from Multicolored Clique, the latter is established using a straightforward reduction from Lemma 6. Proofs appear in the full version.

5.2 Exact Algorithm parameterized by vertex cover

Motivated by the negative result of Theorem 9 we now investigate the complexity of MAJORITY for more restricted parameters. The first parameter we consider is the vertex cover of the incidence graph. This is a natural, though quite restrictive, parameter which is often considered for problems which are W-hard for treewidth.

▶ **Theorem 11.** MAX-THRESHOLD *parameterized by the incidence vertex cover* vc* *is* FPT*.*

The proof appears in the full version, and it works by reducing the problem to integer linear programming parameterized by the number of variables, which is FPT.

5.3 Approximation Algorithm parameterized by feedback vertex set

The results of Theorem 9 naturally pose the following question: can we evade the W-hardness of MAJORITY by designing an FPT-AS for the problem? In this section, though we do not resolve this question, we give some first positive indication that this may be possible. We consider THRESHOLD parameterized by the incidence graph's feedback vertex set (that is, the number of vertices that need to be deleted to make the graph acyclic). This is a natural, well-studied parameter that generalizes vertex cover but is a restriction of treewidth. It is also connected to the concept of back-door sets to acyclicity, which is well-studied in the parameterized CSP literature [16, 7].

Observe that approximating this CSP is non-trivial, since MAX-MAJORITY with constraints of arity two already generalizes MAX-2SAT, and is hence APX-hard. On the other hand, MAX-MAJORITY can easily be 2-approximated by considering any assignment and its negation. Hence, the natural goal here is an approximation ratio of $(1 - \epsilon)$. Using Corollary 11 as a sub-routine we achieve this with an FPT-AS.

▶ **Theorem 12.** *There exists an* FPT-AS *which, given* $\epsilon > 0$ *and an instance* ϕ *of* MAX-THRESHOLD*, computes a* $(1 - \epsilon)$*-approximate solution and runs in time* $f(\text{fvs}^*, \epsilon) \cdot \text{poly}(n)$*, where* fvs^* *is the size of the smallest feedback vertex set of the incidence graph of* ϕ*.*

───── **References** ─────

1   Michael Alekhnovich and Alexander A. Razborov. Satisfiability, branch-width and Tseitin tautologies. *Computational Complexity*, 20(4):649–678, 2011.

2   Per Austrin and Subhash Khot. A characterization of approximation resistance for even k-partite csps. In Robert D. Kleinberg, editor, *Innovations in Theoretical Computer Science, ITCS'13, Berkeley, CA, USA, January 9-12, 2013*, pages 187–196. ACM, 2013.

3   Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. Linear time solvable optimization problems on graphs of bounded clique-width. *Theory Comput. Syst.*, 33(2):125–150, 2000.

4   Nadia Creignou. A dichotomy theorem for maximum generalized satisfiability problems. *Journal of Computer and System Sciences*, 51(3):511–522, 1995.

5   Rodney G. Downey and Michael R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.

6   Serge Gaspers and Stefan Szeider. Kernels for global constraints. In Toby Walsh, editor, *IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011*, pages 540–545. IJCAI/AAAI, 2011.

7   Serge Gaspers and Stefan Szeider. Backdoors to acyclic SAT. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *Automata, Languages, and Programming – 39th International Colloquium, ICALP 2012, Warwick, UK, July 9–13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 363–374. Springer, 2012.

8   Serge Gaspers and Stefan Szeider. Guarantees and limits of preprocessing in constraint satisfaction and reasoning. *Artif. Intell.*, 216:1–19, 2014.

9   Martin Grohe. The structure of tractable constraint satisfaction problems. In Rastislav Kralovic and Pawel Urzyczyn, editors, *MFCS 2006, Stará Lesná, Slovakia, August 28-September 1, 2006, Proceedings*, volume 4162 of *Lecture Notes in Computer Science*, pages 58–72. Springer, 2006.

10 Frank Gurski and Egon Wanke. The tree-width of clique-width bounded graphs without $K_{n,n}$. In Ulrik Brandes and Dorothea Wagner, editors, *Graph-Theoretic Concepts in*

Computer Science, 26th International Workshop, WG 2000, Konstanz, Germany, June 15-17, 2000, Proceedings, volume 1928 of *Lecture Notes in Computer Science*, pages 196–205. Springer, 2000.

11 Johan Håstad. Some optimal inapproximability results. *J. ACM*, 48(4):798–859, 2001.

12 Sanjeev Khanna, Madhu Sudan, and David P. Williamson. A complete classification of the approximability of maximization problems derived from boolean constraint satisfaction. In Frank Thomson Leighton and Peter W. Shor, editors, *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 11–20. ACM, 1997.

13 Subhash Khot and Rishi Saket. Approximating csps using LP relaxation. In Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann, editors, *Automata, Languages, and Programming – 42nd International Colloquium, ICALP 2015, Kyoto, Japan, July 6–10, 2015, Proceedings, Part I*, volume 9134 of *Lecture Notes in Computer Science*, pages 822–833. Springer, 2015.

14 Michael Lampis. Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica*, 64(1):19–37, 2012.

15 Dániel Marx. Parameterized complexity and approximation algorithms. *Comput. J.*, 51(1):60–78, 2008.

16 Sebastian Ordyniak, Daniël Paulusma, and Stefan Szeider. Satisfiability of acyclic and almost acyclic CNF formulas. *Theor. Comput. Sci.*, 481:85–99, 2013.

17 Daniël Paulusma, Friedrich Slivovsky, and Stefan Szeider. Model counting for CNF formulas of bounded modular treewidth. In Natacha Portier and Thomas Wilke, editors, *STACS 2013, February 27 – March 2, 2013, Kiel, Germany*, volume 20 of *LIPIcs*, pages 55–66. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2013.

18 Reinhard Pichler, Stefan Rümmele, Stefan Szeider, and Stefan Woltran. Tractable answer-set programming with weight constraints: bounded treewidth is not enough. *TPLP*, 14(2):141–164, 2014.

19 Sigve Hortemo Sæther, Jan Arne Telle, and Martin Vatshelle. Solving maxsat and #sat on structured CNF formulas. In Carsten Sinz and Uwe Egly, editors, *SAT 2014 – Vienna, Austria, July 14–17, 2014. Proceedings*, volume 8561 of *Lecture Notes in Computer Science*, pages 16–31. Springer, 2014.

20 Marko Samer and Stefan Szeider. Constraint satisfaction with bounded treewidth revisited. *J. Comput. Syst. Sci.*, 76(2):103–114, 2010.

21 Thomas J. Schaefer. The complexity of satisfiability problems. In Richard J. Lipton, Walter A. Burkhard, Walter J. Savitch, Emily P. Friedman, and Alfred V. Aho, editors, *Proceedings of the 10th Annual ACM Symposium on Theory of Computing, May 1-3, 1978, San Diego, California, USA*, pages 216–226. ACM, 1978.

22 Friedrich Slivovsky and Stefan Szeider. Model counting for formulas of bounded clique-width. In Leizhen Cai, Siu-Wing Cheng, and Tak Wah Lam, editors, *ISAAC 2013, Hong Kong, China, Proceedings*, volume 8283 of *Lecture Notes in Computer Science*, pages 677–687. Springer, 2013.

23 Stefan Szeider. On fixed-parameter tractable parameterizations of SAT. In Enrico Giunchiglia and Armando Tacchella, editors, *Theory and Applications of Satisfiability Testing, 6th International Conference, SAT 2003. Santa Margherita Ligure, Italy, May 5-8, 2003 Selected Revised Papers*, volume 2919 of *Lecture Notes in Computer Science*, pages 188–202. Springer, 2003.

24 Stefan Szeider. Not so easy problems for tree decomposable graphs. *CoRR*, abs/1107.1177, 2011.

25 Stefan Szeider. The parameterized complexity of constraint satisfaction and reasoning. In Hans Tompits, Salvador Abreu, Johannes Oetsch, Jörg Pührer, Dietmar Seipel, Masanobu

Umeda, and Armin Wolf, editors, *INAP 2011, and WLP 2011, Vienna, Austria, September 28-30, 2011, Revised Selected Papers*, volume 7773 of *Lecture Notes in Computer Science*, pages 27–37. Springer, 2011.

26 Stefan Szeider. The parameterized complexity of k-flip local search for SAT and MAX SAT. *Discrete Optimization*, 8(1):139–145, 2011.

27 Luca Trevisan. Inapproximability of combinatorial optimization problems. *Electronic Colloquium on Computational Complexity (ECCC)*, 2004.