Constraint satisfaction parameterized by solution size[*]

Andrei A. Bulatov[†]   Dániel Marx[‡]

September 2, 2018

Abstract

In the constraint satisfaction problem (CSP) corresponding to a constraint language (i.e., a set of relations) Γ, the goal is to find an assignment of values to variables so that a given set of constraints specified by relations from Γ is satisfied. The complexity of this problem has received substantial amount of attention in the past decade. In this paper, we study the fixed-parameter tractability of constraint satisfaction problems parameterized by the size of the solution in the following sense: one of the possible values, say 0, is "free," and the number of variables allowed to take other, "expensive," values is restricted. A *size constraint* requires that exactly k variables take nonzero values. We also study a more refined version of this restriction: a *global cardinality constraint* prescribes how many variables have to be assigned each particular value. We study the parameterized complexity of these types of CSPs where the parameter is the required number k of nonzero variables. As special cases, we can obtain natural and well-studied parameterized problems such as INDEPENDENT SET, VERTEX COVER, d-HITTING SET, BICLIQUE, etc.

In the case of constraint languages closed under substitution of constants, we give a complete characterization of the fixed-parameter tractable cases of CSPs with size constraints, and we show that all the remaining problems are W[1]-hard. For CSPs with cardinality constraints, we obtain a similar classification, but for some of the problems we are only able to show that they are BICLIQUE-hard. The exact parameterized complexity of the BICLIQUE problem is a notorious open problem, although it is believed to be W[1]-hard.

Contents

1 Introduction

In a constraint satisfaction problem (CSP) we are given a set of variables, and the goal is to find an assignment of the variables subject to specified constraints. A constraint is usually expressed as a requirement that combinations of values of a certain (usually small) set of variables belong to a certain relation. In the theoretical study of CSPs, one of the key research directions has been the complexity of the CSP when there are restrictions on the type of allowed relations [17, 7, 6, 5, 1]. This research direction has been started by the seminal Schaefer's Dichotomy Theorem [28], which showed that every Boolean CSP (i.e., CSP with 0-1 variables) restricted in this way is either solvable in polynomial time or is NP-complete. An outstanding open question is the so called *Dichotomy conjecture* of Feder and Vardi [12, 23], which suggests that the dichotomy remains true for CSPs over any fixed finite domain. The significance of a dichotomy result is that it is very likely to provide a comprehensive understanding of the algorithmic nature of the problem. Indeed, in order to obtain the tractability part of such a conjecture, one needs to identify and understand all the algorithmic ideas relevant for the problem.

Parameterized complexity [11, 14] investigates problem complexity in finer details than classical complexity. Instead of expressing the running time of an algorithm as a function of the input size n only, the running time is expressed as a function of n and a well-defined parameter k of the input instance (such as the size of the solution k we are looking for). For many problems and parameters, there is a polynomial-time algorithm for every fixed value of k, i.e., the problem can be solved in time $n^{f(k)}$. In this case, it makes sense to ask whether there is an algorithm with running time $f(k) \cdot n^{O(1)}$, that is, whether the combinatorial explosion can be limited to the parameter k only. Problems having algorithms with running time of this form are called *fixed-parameter tractable (FPT)*; it turns out that many well-known NP-hard problems, such as k-Vertex Cover, k-Path, and k-Disjoint Triangles are FPT. On the other hand, the theory of W[1]-hardness suggests that certain problems (e.g., k-Clique, k-Dominating Set) are unlikely to be FPT.

Investigating the fixed-parameter tractability is relevant only for those problems that can be solved in polynomial time for every fixed value of k. For example, the canonical complete problems of the W-hierarchy are (formula or circuit) satisfiability problems where the solution is required to contain exactly k ones; clearly, such a problem is solvable in time $n^{O(k)}$, which is polynomial for every fixed value of k. This leads us naturally to the study of Boolean CSP, where the goal is to find a solution with exactly k ones.

The first attempt to study the parameterized complexity of Boolean CSP was made in [24]. If we consider 0 as a "cheap" value available in abundance, while 1 is "costly" and of limited supply then the natural parameter is the number of 1's in a solution. Boolean CSP asking for a solution that assigns exactly k ones is known as the k-Ones problem (see, e.g. [10, 19]). Clearly, the problem is polynomial-time solvable for every fixed k, as we can search through all assignments with exactly k ones, but it is not at all obvious whether it is FPT. For example, it is possible to express k-Vertex Cover (which is FPT) and k-Independent Set (which is W[1]-hard) as a Boolean CSP. Therefore, characterizing the parameterized complexity of k-Ones requires understanding a class of problems that includes, among many other things, the most basic parameterized graph problems. It turned out that the parameterized complexity of the k-Ones problem depends on a combinatorial property called *weak separability* in [24]. Assuming that the constraints are restricted to a finite set Γ of Boolean relations, if every relation in Γ is weakly separable, then the problem is FPT; if Γ contains even one relation violating weak separability, then the problem is W[1]-hard. Another natural problem of this flavor is deciding whether there exists a solution with at most k ones; however, this problem is always FPT, as it follows from Lemma 2.3 below even in a more general case.

There have been further parameterized complexity studies of Boolean CSP [21, 29, 22, 20], but CSP's with larger domains (i.e., where the variables are not Boolean) were not studied. In most cases, we expect that results for larger domains are much more complex than for the Boolean case, and usually require significant new ideas (compare e.g., Schaefer's Theorem [28] with the 3-element version [5]). The goal of the present paper is to generalize the results of [24] to non-Boolean domains. First, we have to define what the proper generalization of k-Ones is if the variables are not Boolean. One natural generalization assumes that there is a distinguished "cheap" value 0 and requires that in a solution there are exactly k nonzero variables. We will call this version of the CSP a *constraint satisfaction problem with size constraints* and denote it by OCSP. In another generalization of k-Ones, we have a cardinality requirement not only on the total number of nonzero variables, but a separate cardinality requirement for each nonzero value restricting the number of times this value is used: A mapping $\pi : D \setminus \{0\} \to \mathbb{N}$ is given in the input, and it is required that for each nonzero value d, exactly $\pi(d)$ variables are assigned value d. In the CSP and AI literature, requirements of this form are called *global cardinality constraints* [2, 3, 25, 9, 8, 15, 27] and have been intensively studied. We will call this problem the *constraint satisfaction problem with cardinality constraints* and denote it by CCSP. In both versions, the parameter is the number of nonzero values required, that is, k for OCSP, and $\sum_{d \in D \setminus \{0\}} \pi(d)$ for CCSP. We investigate both versions; as we shall see, there are interesting and unexpected differences between the complexity of the two variants.

A natural minor generalization of CSPs is to allow the use of constants in the input, that is, certain variables in the input can be fixed to constant values, or equivalently, the constant unary relation $\{d\}$ is allowed for every element d of the domain. Yet another equivalent way of formulating this generalization is requiring that Γ is closed under substitution of constants. It is known that to classify the complexity of the general CSP with respect to polynomial-time solvability, it suffices to classify constraint languages closed under substitution of constants [7]. This motivates our assumption that the constant relations are available for CSPs with cardinality and size constraints. While there is no result similar to that from [7] for the versions of CSPs we study here (and thus this assumption somewhat diminishes the generality of our results), this setting is still quite general and at the same time more robust. Lots of technicalities can be avoided with this formulation. For example, being able to substitute constants ensures that the decision and search problems are equivalent: by repeatedly substituting constants and solving the resulting decision problems, we can actually find a solution.

Is weak separability the right tractability criterion in the non-Boolean case? It is not difficult to observe that the algorithm of [24] using weak separability generalizes for non-Boolean problems. (In fact, we give a much simpler algorithm in this paper.) However, it is not true that only weakly separable relations are tractable. It turns out that there are certain degeneracies and symmetries that allow us to solve the problem even for some relations that are not weakly separable. To understand these degenerate situations, we introduce the notion of multivalued morphisms, which is a generalization of homomorphisms. While the use of algebraic techniques and homomorphisms is a standard approach for understanding the complexity of CSPs [7], this notion is new and seems to be essential for understanding the problem in our setting.

Results. For CSP with size constraints, we prove a dichotomy result:

Theorem 1.1. *For every finite Γ closed under substitution of constants,* OCSP(Γ) *is either* FPT *or* W[1]-*hard.*

The precise tractability criterion (which is quite technical) is stated in Section 4. The algorithmic part of Theorem 1.1 consists of preprocessing to eliminate degeneracies and trivial cases,

followed by the use of weak separability. In the hardness part, we take a relation R having a counterexample to weak separability, and try to use it to show that a known W[1]-hard problem can be simulated with this relation. In the Boolean case [24], this can be done in a fairly simple way: by identifying coordinates and substituting 0's, we can assume that the relation R is binary, and we need to prove hardness only for two concrete binary relations. For larger domains, however, this approach does not work. We cannot reduce the counterexample to a binary relation by identifying coordinates, thus it seems that a complex case analysis would be needed. Fortunately, our hardness proof is more uniform than that. We introduce gadgets that control the values that can appear on the variables. There are certain degenerate cases when these gadgets do not work. However, these degenerate cases can be conveniently described using multivalued morphisms, and these cases turn out to be exactly the cases that we can use in the algorithmic part of the proof.

In the case of cardinality constraints, we face an interesting problem. Consider the binary relation R containing only tuples $(0,0)$, $(1,0)$, and $(0,2)$. Given a CSP instance with constraints of this form, finding a solution where the number of 1's is exactly k and the number of 2's is exactly k is essentially equivalent to finding an independent set of a bipartite graph with k vertices in both classes, or equivalently, a complete bipartite graph (biclique) with $k + k$ vertices. However, the parameterized complexity of the k-Biclique problem is a longstanding open question (it is conjectured to be W[1]-hard [16]). Thus at this point, it is not possible to give a dichotomy result similar to Theorem 1.1 in the case of cardinality constraints, unless assuming that Biclique is hard.

Theorem 1.2. *For every finite Γ closed under substitution of constants,* CCSP(Γ) *is either* FPT *or* Biclique-*hard.*

Note that in many cases, we actually prove W[1]-hardness in Theorem 1.2 and we state Biclique-hardness only in very specific situations. Similarly to Theorem 1.1, the algorithmic part of Theorem 1.2 exploits weak separability after a preprocessing phase, but in this case there is a final phase that extends the solution using certain degeneracies. The hardness proof is similar to Theorem 1.1, however, it requires at times different combinatorial arguments, slightly different types of degeneracies, and reductions from different hard parameterized problems.

The parameterized complexity of Biclique has been a significant open problem for a number of years and in fact it is the most natural and most easily stated problem on graphs whose parameterized complexity status is open. The parameterized complexity of similar basic problems (such as k-Clique, k-Path, k-Disjoint Triangles, etc.) is well understood, but it seems that very different techniques are required to tackle Biclique. Our results give a further incentive for the study of Biclique: its hardness would explain hardness in all the cases that are not classified as FPT or W[1]-hard in this paper. In other words, in a very general problem family, the hardness of Biclique is the only question that is not yet understood, highlighting the importance of this problem. We believe this observation to be a significant byproduct of our study.

Organization. The paper is organized as follows. In Section 2 we introduce constraint satisfaction problems with cardinality constraints and argue that only 0-valid constraint languages need to be studied, so from that point on, we consider only cc0-languages. Weak separability and various degeneracies of constraint languages are introduced and studied in Section 3. The main classification results are proved in Section 4 (for size constraints) and Section 5 (for cardinality constraints). In both sections, we first present the algorithmic side of the classification, then prove hardness in various cases. After going through certain fairly degenerate situations, the most generic (and from the point of view of proof techniques, most interesting) proofs are found at the end of these sections. Section 6 contains examples that demonstrate some of the concepts introduced in the paper. We will continuously refer to these examples throughout the paper.

2   Preliminaries

Let D be a set. We assume that D contains a distinguished element 0. Let D^n denote the set of all n-tuples of elements from D. An n-ary *relation* on D is a subset of D^n, and a *constraint language* Γ is a set of relations on D. In this paper, constraint languages are assumed to be finite. We denote by $\mathrm{dom}(\Gamma)$ the set of all values that appear in tuples of the relations in Γ. In proofs throughout the paper, we use languages derived from the original language Γ, so this set does not have to be equal to D. Given a set D and a constraint language Γ, an instance of the *constraint satisfaction problem* (CSP) is a pair $I = (V, \mathcal{C})$, where V is a set of *variables*, and $\mathcal{C}$ is a set of *constraints*. Each constraint is a pair $\langle \mathbf{s}, R \rangle$, where R is a (say, n-ary) relation from Γ, and $\mathbf{s}$ is an n-tuple of variables (repetitions of variables are allowed). A *satisfying assignment* of I is a mapping $\tau : V \to D$ such that for every $\langle \mathbf{s}, R \rangle \in \mathcal{C}$ with $\mathbf{s} = (s_1 \ldots, s_n)$ the image $\tau(\mathbf{s}) = (\tau(s_1), \ldots, \tau(s_n))$ belongs to R. The question in the CSP is whether there is a satisfying assignment for a given instance. The CSP over constraint language Γ is denoted by $\mathrm{CSP}(\Gamma)$.

The *size* of an assignment is the number of variables receiving nonzero values. A *size constraint* is a requirement to a satisfying assignment to have a prescribed size. The variant of $\mathrm{CSP}(\Gamma)$ that allows size constraints will be denoted by $\mathrm{OCSP}(\Gamma)$. A *cardinality constraint* for a CSP instance I is a mapping $\pi : D \setminus \{0\} \to \mathbb{N}$ with $\sum_{a \in D} \pi(a) \le |V|$. A satisfying assignment τ of I satisfies the cardinality constraint π if the number of variables mapped to each $a \in D$ equals $\pi(a)$. We denote by $\mathrm{CCSP}(\Gamma)$ the variant of $\mathrm{CSP}(\Gamma)$ where the input also contains a cardinality constraint π and the size constraint $k = \sum_{a \in D \setminus \{0\}} \pi(a)$ (this constraint is used a parameter); the question is, given an instance I, an integer k, and a cardinality constraint π, whether there is a satisfying assignment of I of size k and satisfying π. (Obviously, $\sum_{a \in D \setminus \{0\}} \pi(a) = k$ must hold, thus specifying k in the input is in a sense redundant. However, for the uniform treatment of both problems, it will be convenient to assume that k appears in the input in both cases.) A *solution* of an OCSP (resp., CCSP) instance is a satisfying assignment that also satisfies the size (resp., cardinality) constraints. So, instances of OCSP (resp., CCSP) are triples $(V, \mathcal{C}, k)$ (resp., quadruples $(V, \mathcal{C}, k, \pi)$). For both $\mathrm{OCSP}(\Gamma)$ and $\mathrm{CCSP}(\Gamma)$, we are interested in fixed-parameter tractability with respect to the parameter k, i.e., the goal is an algorithm with running time $f(k) \cdot n^{O(1)}$ for some computable function f. Note that we are making a distinction between two terms: in "satisfying assignment," we do not require that the cardinality constraint is satisfied, while a "solution" always means that the cardinality constraint is satisfied.

If Γ is a constraint language on the 2-element set $\{0, 1\}$, then to specify a global cardinality constraint it suffices to specify the number of ones we want to have in a solution. This problem is also known as the k-ONES(Γ) problem [10, 24]. Note that this problem can be viewed as both $\mathrm{CCSP}(\Gamma)$ and $\mathrm{OCSP}(\Gamma)$. In the general case, $\mathrm{OCSP}(\Gamma)$ polynomial time reduces to $\mathrm{CCSP}(\Gamma)$.

Lemma 2.1. *For any constraint language Γ over a set D, there is a polynomial-time reduction from $\mathrm{OCSP}(\Gamma)$ to $\mathrm{CCSP}(\Gamma)$.*

Indeed, for any instance $(V, \mathcal{C}, k)$ of $\mathrm{OCSP}(\Gamma)$ it suffices to solve $O(k^{d-1})$ instances $(V, \mathcal{C}, k, \pi)$ of $\mathrm{CCSP}(\Gamma)$ such that $\sum_{a \in D \setminus \{0\}} \pi(a) = k$.

Examples 6.1–6.4 demonstrate some concrete problems that can be expressed in this framework.

2.1   Closures and 0-validity

A constraint language Γ is called *constant closed* (cc-, for short) if along with every (say, n-ary) relation $R \in \Gamma$, any i, $1 \le i \le n$, and any $d \in D$ the relation obtained by *substitution of constants* $R^{|i;d} = \{(a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_n) \mid (a_1, \ldots, a_{i-1}, d, a_{i+1}, \ldots, a_n) \in R\}$, also belongs to

Γ. Substitution of constants $d_1, \ldots, d_q$ for coordinate positions $i_1, \ldots, i_q$ is defined in a similar way; the resulting relation is denoted by $R^{|i_1, \ldots, i_q; d_1, \ldots, d_q}$. We call the smallest cc-language containing a constraint language Γ the *cc-closure* of Γ. It is easy to see that the cc-closure of Γ is the set of relations obtained from relations of Γ by substituting constants. Let f be a satisfying assignment for an instance $I = (V, \mathcal{C}, k)$ of OCSP(Γ) and $S = \{v \mid f(v) \neq 0\}$. We say that the instance $I' = (V', \mathcal{C}', k')$ is *obtained by substituting the nonzero values of f as constants* if I' is constructed as follows: $V' = V \setminus S$, and for each constraint $\langle \mathbf{s}, R \rangle \in \mathcal{C}$ such that $i_1, \ldots i_r$ are the indices of the variables from $\mathbf{s}$ contained in S and $\{j_1, \ldots, j_q\}$ is the set of indices of variables from $\mathbf{s}$ not in S, we include in $\mathcal{C}'$ the constraint $\langle (v_{j_1}, \ldots, v_{j_q}), R^{|i_1, \ldots, i_r; f(v_{i_1}), \ldots f(v_{i_r})} \rangle$. The size constraint k' is set to k minus the size of f (we always make sure that f is such that $k' \geq 0$). Observe that assigning 0 to every variable of I' is a satisfying assignment. This operation is defined similarly for a CCSP(Γ) instance $I = (V, \mathcal{C}, k, \pi)$, but in this case the new cardinality constraint π' is given by $\pi'(d) = \pi(d) - |\{v \in V \mid f(v) = d\}|$ (we always assume that π and f are such that π' is nonnegative).

In general, OCSP(Γ) or CCSP(Γ) can become harder if we replace Γ with its cc-closure, see Examples 6.5 and 6.6. Thus a classification for cc-languages does not immediately imply a classification for all languages.

A relation is said to be *0-valid* if the all-zero tuple belongs to the relation. A constraint language Γ is a *cc0-language* if every $R \in \Gamma$ is 0-valid, and every 0-valid relation R' obtained from R by substitution of constants belongs to Γ (see Example 6.7). An instance I is said to be *0-valid* if the all-0 assignment is a satisfying one. In particular every instance of OCSP(Γ) or CCSP(Γ) for a cc0-language Γ is 0-valid. The following observation is clear (but note that Γ_0 is not necessarily a cc-language, as substitution into a 0-valid relation does not necessary result in a 0-valid relation):

Proposition 2.2. *If Γ is a cc-language and Γ_0 is the set of 0-valid relations in Γ, then Γ_0 is a cc0-language.*

We say that tuple $\mathbf{t}_1 = (a_1, \ldots, a_r)$ is an *extension* of tuple $\mathbf{t}_2 = (b_1, \ldots, b_r)$ if they are of the same length and for every $1 \leq i \leq r$, $a_i = b_i$ if $b_i \neq 0$. Tuple $\mathbf{t}_2$ is then called a *subset* of $\mathbf{t}_1$. A *minimal satisfying extension* of an assignment f is an extension f' of f (where f, f' are viewed as tuples) such that f' is satisfying, and f has no satisfying extension $f'' \neq f'$ such that f' is an extension of f''. We show that the minimal satisfying extensions of size at most k can be enumerated by a simple branching algorithm, implying a bound on the number of such extensions.

Lemma 2.3. *Let Γ be a finite constraint language over D. There are functions $d'_\Gamma(k)$ and $e'_\Gamma(k)$ such that for every instance of CSP(Γ) with n variables, every assignment f has at most $d'_\Gamma(k)$ minimal satisfying extensions of size at most k and all these minimal extensions can be enumerated in time $e'_\Gamma(k)n^{O(1)}$.*

Proof. Let $k_0 \leq k$ be the size of f. The minimal satisfying extensions of f can be enumerated by a bounded-depth search tree algorithm. If f is a satisfying assignment, then f itself is the unique minimal satisfying extension of f. Suppose therefore that a constraint is not satisfied by f. If every variable of the constraint has nonzero value, then the assignment has no satisfying extension. Otherwise, we try to assign a nonzero value to one of the 0-valued variables of the constraint, thus we branch into at most $(|D| - 1)r_{\max}$ directions, where $r_{\max}$ is the maximum arity of the relations in Γ. If the assignment obtained by this modification is still not satisfying, then the process is repeated with some unsatisfied constraint. If the assignment is still not satisfying after making $k - k_0$ variables nonzero, then this branch of the search is terminated. If we obtain a satisfying assignment after assigning a nonzero value to a set S of at most $k' \leq k - k_0$ variables, then we check whether this extension is minimal by trying each of the $2^{k'}$ subsets of the changed variables. This

last check ensures that every extension produced by the algorithm is indeed minimal. To see that every minimal satisfying extension f' of size at most k is eventually enumerated, observe that at every branching we can make a substitution that is compatible with f', i.e., the current assignment is a subset of f'. Thus after at most k steps, the algorithm finds a satisfying assignment that is a subset of f'. As f' is a minimal satisfying extension, this subset cannot be proper and has to be f' itself.

The height of the search tree is at most k and the branching factor is at most $(|D|-1)r_{\max}$, thus at most $((|D|-1)r_{\max})^k$ assignments are enumerated. The running time is polynomial at each node and the final check at each leaf takes time $2^k n^{O(1)}$. Thus the total running time is $e'_\Gamma(k)n^{O(1)}$ for a suitable e'_Γ. $\qquad\square$

A consequence of Lemma 2.3 is that, as in [24], $\mathrm{CCSP}(\Gamma)$ and $\mathrm{OCSP}(\Gamma)$ can be reduced to a set of 0-valid instances. Let k be the parameter of the $\mathrm{OCSP}(\Gamma)$ or $\mathrm{CCSP}(\Gamma)$ instance (i.e., the required size of the solution). We enumerate all the minimal satisfying extensions of size at most k of the all zero assignment and obtain the 0-valid instances by substituting the nonzero values as constants.

Corollary 2.4. *Let Γ be a cc-language and let $\Gamma_0 \subseteq \Gamma$ be the set of all 0-valid relations. If $\mathrm{CCSP}(\Gamma_0)$ is FPT/W[1]-hard/BICLIQUE-hard, then $\mathrm{CCSP}(\Gamma)$ is as well. The same holds for $\mathrm{OCSP}(\Gamma)$ and $\mathrm{OCSP}(\Gamma_0)$.*

By Prop. 2.2, if Γ is a cc-language, then Γ_0 is a cc0-language. Thus Corollary 2.4 reduces the classification of the complexity of cc-languages into the classification of cc0-languages. Thus in the rest of the paper, we have to deal with cc0-languages only.

2.2 Reductions

To obtain the W[1]-hardness results, we use the standard notion of *parameterized reduction* [11, 14].

Definition 2.5. *A parameterized reduction from parameterized problem A to parameterized problem B is a mapping R from the instances of A to the instances of B with the following properties:*

1. *I is a yes-instance of A if and only if $R(I)$ is a yes-instance of B,*

2. *$R(I)$ is computable in time $f(k) \cdot n^{O(1)}$, where n is the size of I, k is the parameter of I, and $f(k)$ is a computable function depending only on k,*

3. *the parameter of $R(I)$ is at most $g(k)$, where k is the parameter of I and $g(k)$ is a computable function depending only on k.*

It is easy to see that if there is a parameterized reduction from A to B and B is FPT, then A is FPT as well.

In the hardness proofs, we will be reducing from the following parameterized problems (t is the parameter of the instance).

- INDEPENDENT SET. Given a graph G with vertices v_i ($1 \le j \le n$), find an independent set of size t.

- MULTICOLORED INDEPENDENT SET: Given a graph G with vertices $v_{i,j}$ ($1 \le i \le t$, $1 \le j \le n$), find an independent set of size t of the form $\{v_{1,y_1}, \ldots, v_{t,y_t}\}$.

- IMPLICATIONS: Given a directed graph G and an integer t, find a set C of vertices with exactly t vertices such that there is no directed edge $\overrightarrow{uv}$ with $u \in C$ and $v \notin C$.

- MULTICOLORED IMPLICATIONS: Given a directed graph G with vertices $v_{i,j}$ ($1 \leq i \leq t$, $1 \leq j \leq n$), find a set $C = \{v_{1,y_1}, \ldots, v_{t,y_t}\}$ with exactly t vertices such that there is no directed edge $\overrightarrow{uv}$ with $u \in C$ and $v \notin C$.

- BICLIQUE. Given a bipartite graph $G(A, B)$, find two sets $A' \subseteq A$ and $B' \subseteq B$, each of size exactly t, such that every vertex of A' is adjacent to every vertex of B'.

INDEPENDENT SET is one of the basic W[1]-hard problems and it is not difficult to show that MULTICOLORED INDEPENDENT SET is W[1]-hard as well (see [13]). IMPLICATIONS was introduced and proved to be W[1]-hard in [24]. Lemma 2.6 below shows that MULTICOLORED IMPLICATIONS is W[1]-hard. The parameterized complexity of BICLIQUE is a longstanding open question, and it is expected to be W[1]-hard (cf. [16, Section 8]).

Lemma 2.6. MULTICOLORED IMPLICATIONS *is* W[1]-*hard.*

Proof. The proof is by reduction from CLIQUE: let H be a graph where a clique of size k has to be found. It can be assumed that the number n of vertices in H equals the number of edges: adding isolated vertices or removing acyclic components does not change the problem. Let u_1, $\ldots$, u_n be the vertices of H and let e_1, $\ldots$, e_n be the edges of H. We construct the graph G of the MULTICOLORED IMPLICATIONS instance as follows. Set $t := k + \binom{k}{2}$. The vertex set of G is $\{v_{i,j} \mid 1 \leq i \leq t, 1 \leq j \leq n\}$. Let $\iota(i,j)$ be a bijective mapping between the 2-element subsets of $\{1, \ldots, k\}$ and the set $\{k+1, \ldots, k + \binom{k}{2}\}$. Intuitively, for $1 \leq i \leq k$, the choice of vertex v_{i,y_i} represents the choice of the i-th vertex of the clique K of H we are looking for. Furthermore, for $1 \leq i < j \leq k$, the choice of vertex $v_{\iota(i,j),y_{\iota(i,j)}}$ represents the edge between the i-th and j-th vertex of the clique K. To enforce this interpretation, for every $1 \leq i < j \leq n$ and $1 \leq s \leq n$, if u_a and u_b ($a < b$) are the endpoints of edge e_s of G, then let us add two directed edges $\overrightarrow{v_{\iota(i,j),s}v_{i,a}}$ and $\overrightarrow{v_{\iota(i,j),s}v_{j,b}}$ to H. This completes the description of the reduction.

Suppose that G has a clique $u_{x_1}, \ldots, u_{x_k}$ of size k ($x_1 < \cdots < x_k$) and $e_{z_{i,j}}$ is the edge connecting u_{x_i} and u_{x_j}. In this case, no directed edge of H leaves the set C that contains v_{i,x_i} for $1 \leq i \leq k$ and $v_{\iota(i,j),z_{i,j}}$ for $1 \leq i < j \leq k$, hence it is a solution of the MULTICOLORED IMPLICATIONS instance.

For the reverse direction, suppose that $C = \{v_{i,y_i} \mid 1 \leq i \leq t\}$ is a solution of the MULTICOLORED IMPLICATIONS instance. We claim that $u_{y_1}, \ldots, u_{y_k}$ is a clique of size k in G. We claim that edge $e_{y_{\iota(i,j)}}$ connects u_{y_i} and u_{y_j} (in particular, this means that $y_i \neq y_j$). Let $s = y_{\iota(i,j)}$ and let u_a and u_b ($a < b$) be the two endpoints of edge e_s. Since $v_{\iota(i,j),s} \in C$, the edges $\overrightarrow{v_{\iota(i,j),s}v_{i,a}}$ and $\overrightarrow{v_{\iota(i,j),s}v_{j,b}}$ imply that $v_{i,a}, v_{j,b} \in C$. As C contains exactly one of the vertices $v_{i,1}, \ldots, v_{i,n}$, it follows that $y_i = a$ and, similarly, $y_j = b$. That is, u_{y_i} and u_{y_j} are connected by the edge e_s. $\square$

Let Γ be a constraint language. A relation R is *intersection definable* in Γ if R is the set of all solutions to a certain instance of CSP(Γ).

Proposition 2.7. *Let Γ be a constraint language and R a relation intersection definable in Γ. Then* CCSP($\Gamma \cup \{R\}$) *(*OCSP($\Gamma \cup \{R\}$)*) is polynomial time reducible to* CCSP(Γ) *(respectively, to* OCSP(Γ)*).*

Proof. Indeed, let I be an instance of CCSP(Γ) (OCSP(Γ)) that expresses R. To reduce we just need to replace every occurrence of R with I. $\square$

3 Properties of constraints

By Corollary 2.4, it is sufficient for proving Theorems 1.1 and 1.2 to consider only cc0-languages. Thus in the rest of the paper, we always assume that constraint languages are cc0-languages.

3.1 Weak separability

In this subsection we introduce the key property regulating the complexity of CSPs with cardinality constraints. In the Boolean case, the tractability of 0-valid constraints depends only on weak separability [24]. This is not true exactly this way for larger domains: as we shall see (Theorems 4.1 and 5.2), the complexity characterizations have further technical conditions.

Definition 3.1. *Two tuples* $\mathbf{t}_1 = (a_1, \ldots, a_r)$ *and* $\mathbf{t}_2 = (b_1, \ldots, b_r)$ *of the same length* r *are called* disjoint *if* $a_i = 0$ *or* $b_i = 0$ *for every* $1 \leq i \leq r$. *The* union *of two disjoint tuples* $\mathbf{t}_1$ *and* $\mathbf{t}_2$ *is the tuple* $\mathbf{t}_1 + \mathbf{t}_2 = (c_1, \ldots, c_r)$ *where* $c_i = a_i$ *if* $a_i \neq 0$ *and* $c_i = b_i$ *otherwise. If* $(a_1, \ldots, a_r)$ *is an* extension *of* $(b_1, \ldots, b_r)$, *then their* difference *is the tuple* $(c_1, \ldots, c_r)$ *where* $c_i = a_i$ *if* $b_i = 0$ *and* $c_i = 0$ *otherwise. A tuple* $\mathbf{t} = (c_1, \ldots, c_r)$ *is* contained *in a set* $C \subseteq D \setminus \{0\}$ *if every nonzero* c_i *is in* C.

The following property plays a crucial role in our classification:

Definition 3.2. *A 0-valid relation* R *is said to be* weakly separable *if the following two conditions hold:*

1. *For every pair of disjoint tuples* $\mathbf{t}_1, \mathbf{t}_2 \in R$, *we have* $\mathbf{t}_1 + \mathbf{t}_2 \in R$.

2. *For every pair of disjoint tuples* $\mathbf{t}_1, \mathbf{t}_2$ *with* $\mathbf{t}_2, \mathbf{t}_1 + \mathbf{t}_2 \in R$, *we have* $\mathbf{t}_1 \in R$.

A constraint language Γ *is said to be* weakly separable *if every relation from* Γ *is weakly separable.*

If constraint language Γ is not weakly separable, then a triple $(R, \mathbf{t}_1, \mathbf{t}_2)$, $R \in \Gamma$, witnessing that is called a *union counterexample* if $\mathbf{t}_1, \mathbf{t}_2$ violate condition (1), while if $\mathbf{t}_1, \mathbf{t}_2$ violate condition (2) it is called a *difference counterexample*. Examples 6.8 and 6.9 demonstrate these notions and show how can we prove hardness in the Boolean case if there is a counterexample to weak separability [24]. However, as Example 6.10 shows, in case of larger domains, OCSP(Γ) or CCSP(Γ) can be fixed-parameter tractable even if the cc0-language Γ is not weakly separable.

Throughout the paper, we often refer to satisfying assignments that are nonzero, but have as few nonzero elements as possible. A satisfying assignment f is said to be a *minimal satisfying assignment* if it has a nonzero value and is not a proper extension of any nonzero satisfying assignment (note that we explicitly exclude the all-0 assignment from this definition). A consequence of Lemma 2.3 bounds the number of minimal satisfying assignments:

Lemma 3.3. *Let* Γ *be a finite constraint language. There are functions* $d_\Gamma(k)$ *and* $e_\Gamma(k)$ *such that for any instance of* CSP(Γ) *with* n *variables, every variable* v *is nonzero in at most* $d_{\Gamma(k)}$ *minimal satisfying assignments of size at most* k *and all these minimal satisfying assignments can be enumerated in time* $e_\Gamma(k) n^{O(1)}$.

Proof. Let $\delta_{v,d}$ be the assignment that assigns value d to variable v and 0 to every other variable. For a fixed variable v and every nonzero $d \in \mathrm{dom}(\Gamma)$, let us use the algorithm of Lemma 2.3 to enumerate all the minimal satisfying extensions of $\delta_{v,d}$. We claim that every minimal satisfying assignment f with $f(v) \neq 0$ appears in the enumerated assignments. Indeed, if $f(v) = d$, then f is a satisfying extension of $\delta_{v,d}$, and clearly it is a minimal satisfying extension, as no nonzero subset of f is satisfying. Thus $d_\Gamma(k) = (|\mathrm{dom}(\Gamma)| - 1) d'_\Gamma(k)$ bounds the number of minimal satisfying assignments where variable v is nonzero. The time bound follows from the time bound of Lemma 2.3. $\square$

The following combinatorial property is the key for solving weakly separable instances (this property does not necessarily hold for arbitrary relations, see Example 6.11):

Lemma 3.4. *Let Γ be a weakly separable finite cc0-language over D and I an instance of $\mathrm{CCSP}(\Gamma)$ or $\mathrm{OCSP}(\Gamma)$.*

1. *Every satisfying assignment f of I is the union of pairwise disjoint minimal satisfying assignments.*

2. *If there is a satisfying assignment f with $f(v) = d$ for some variable v and $d \in D$, then there is a minimal satisfying assignment f' with $f'(v) = d$.*

Proof. 1. The proof is by induction on the size of f; if the size is 0, then there is nothing to show. Let f' be a subset of f that is a minimal satisfying assignment. Let f'' be the difference of f and f'; since Γ is weakly separable, f'' is also a satisfying assignment. By the induction hypothesis, f'' is the disjoint union of minimal satisfying assignments $f_1, \ldots, f_\ell$, and hence f is the disjoint union of $f_1, \ldots, f_\ell, f'$.

2. Since f is the disjoint union of minimal satisfying assignments $f_1, \ldots, f_\ell$, there has to be an $1 \leq i \leq \ell$ for which $f_i(v) = d$. $\qquad\square$

Lemma 3.4 allows us to solve weakly separable instances by enumerating all minimal satisfying assignments and then finding a subset of these assignments that are disjoint and together satisfy the size/cardinality constraint. When finding these disjoint assignments, we can exploit the fact that by Lemma 3.3, each such assignment is non-disjoint from a bounded number of other assignments.

Theorem 3.5. *If Γ over D is a finite weakly separable cc0-language, then $\mathrm{CCSP}(\Gamma)$ and $\mathrm{OCSP}(\Gamma)$ are fixed-parameter tractable.*

Proof. We present the proof for $\mathrm{CCSP}(\Gamma)$; the proof for $\mathrm{OCSP}(\Gamma)$ is analogous and actually simpler. Alternatively, by Lemma 2.1, $\mathrm{OCSP}(\Gamma)$ reduces to $\mathrm{CCSP}(\Gamma)$ in polynomial time. Thus the fixed-parameter tractability of $\mathrm{CCSP}(\Gamma)$ immediately implies those of $\mathrm{OCSP}(\Gamma)$.

First, we enumerate all minimal satisfying assignments of size at most k using Lemma 3.3; let S be the set of all these assignments (note that S has size at most $d_\Gamma(k) \cdot n$, where n is the number of variables). By Lemma 3.4, every solution can be formed as the disjoint union of members of S. Furthermore, by weak separability, the disjoint union of satisfying assignments is always satisfying. Thus the question is whether it is possible to find a subset S' of S that contains pairwise disjoint assignments and their union satisfies the cardinality constraints.

Clearly, $|S'| \leq k$. We can associate a mapping π_f to each assignment f in S, with the meaning that f sets exactly $\pi_f(i)$ variables to value i. Let $K := k \cdot d_\Gamma(k)$ for the function d_Γ appearing in Lemma 3.3. For each mapping π', let $S_{\pi'}$ contain the first K assignments whose associated mapping is π' (or all such assignments if there are less than K of them). Let S^* be the union of all these sets $S_{\pi'}$. We claim that if there is a solution $S' \subseteq S$, then there is a solution which is a subset of S^*. Thus we can find a solution by trying all subsets of size at most k in S^*.

To prove the claim, let S' be a solution such that $|S' \setminus S^*|$ is minimum possible. Let $f \in S' \setminus S^*$; this means that $f \notin S_{\pi_f}$ and hence $|S_{\pi_f}| = K$. Assignments in $S' \setminus \{f\}$ assign nonzero values to less than k variables, denote by X these variables. By Lemma 3.3, each variable is nonzero in at most $d_\Gamma(k)$ assignments of S (and of the subset S_{π_f}), and hence there is at least one assignment $f' \in S_{\pi_f}$ that is zero on every variable of X. Replacing f with $f' \in S^*$ yields a solution with strictly smaller number of assignments not in S^*, contradicting the minimality of S'. $\qquad\square$

3.2 Morphisms

Homomorphisms and polymorphisms are standard tools for understanding the complexity of constraints [7, 18]. We make use of the notion of multivalued morphisms, a generalization of homomorphisms, that in a different context has appeared in the literature for a while (see, e.g. [26])

11

under the guise of hyperoperation. For a constraint language, we introduce a classification of values from its domain into 4 types according to the existence of such morphisms of the language (Definition 3.8). This classification into 4 types of elements and the observation that these types play an essential role in the way the MVM gadgets (Section 3.4) work are the main technical ideas behind the hardness proofs.

For a subset $0 \in D' \subseteq D$ and an n-ary relation R on D, by $R_{|D'}$ we denote the relation $R \cap (D')^n$. For a constraint language Γ, the language $\Gamma_{|D'}$ denotes the set of all relations $R_{|D'}$ for $R \in \Gamma$.

An *endomorphism* of Γ is a mapping $h : \mathrm{dom}(\Gamma) \to \mathrm{dom}(\Gamma)$ such that $h(0) = 0$ and for every $R \in \Gamma$ and $(a_1, \ldots, a_r) \in R$, the tuple $(h(a_1), \ldots, h(a_r))$ is also in R. Note that the requirement $h(0) = 0$ is nonstandard, but it is very natural in our setting. For a tuple $\mathbf{t} = (a_1, \ldots, a_r) \in \mathrm{dom}(\Gamma)^r$, we denote by $h(\mathbf{t})$ the tuple $(h(a_1), \ldots, h(a_r))$. Observe that the mapping sending all elements of $\mathrm{dom}(\Gamma)$ to 0 is an endomorphism of any 0-valid language. An *inner homomorphism* of Γ from D_1 to D_2 with $0 \in D_1, D_2 \subseteq \mathrm{dom}(\Gamma)$ is a mapping $h : D_1 \to D_2$ such that $h(0) = 0$ and $h(\mathbf{t}) \in R$ holds for any relation $R \in \Gamma$ and $\mathbf{t} \in R_{|D_1}$.

A *multivalued morphism* of Γ is a mapping $\phi : \mathrm{dom}(\Gamma) \to 2^{\mathrm{dom}(\Gamma)}$ such that $\phi(0) = \{0\}$ and for every $R \in \Gamma$ and $(a_1, \ldots, a_r) \in R$, we have $\phi(a_1) \times \cdots \times \phi(a_r) \subseteq R$. An *inner multivalued morphism* ϕ from D_1 to D_2 where $0 \in D_1, D_2 \subseteq \mathrm{dom}(\Gamma)$ is defined to be a mapping $\phi : D_1 \to 2^{D_2}$ such that $\phi(0) = \{0\}$ and for every $R \in \Gamma$ and $(a_1, \ldots, a_r) \in R_{|D_1}$, we have $\phi(a_1) \times \cdots \times \phi(a_r) \subseteq R_{|D_2}$. If ϕ is an (inner) multivalued morphism, and $\mathbf{t} = (a_1, \ldots, a_r)$ is a tuple, then we define $\phi(\mathbf{t}) = \phi(a_1) \times \cdots \times \phi(a_r)$. Example 6.12 shows several concrete examples.

Observation 3.6. *Let $\phi : \mathrm{dom}(\Gamma) \to 2^{\mathrm{dom}(\Gamma)}$ be a multivalued morphism $[\psi : D_1 \to 2^{D_2}$ be an inner multivalued morphism] of a constraint language Γ, and let $\phi' : \mathrm{dom}(\Gamma) \to 2^{\mathrm{dom}(\Gamma)}$ [resp., $\psi' : D_1 \to 2^{D_2}]$ be a mapping such that $\phi'(d) \subseteq \phi(d)$ for $d \in \mathrm{dom}(\Gamma)$ [resp., $\psi'(d) \subseteq \psi(d)$ for $d \in D_1$]. Then ϕ' is a multivalued morphism [ψ' is an inner multivalued morphism].*

The *product* $g \circ h$ of two endomorphisms or inner homomorphisms is defined by $(g \circ h)(x) = h(g(x))$ for every $x \in D$. That is, $g \circ h$ means that g is applied first and then h. It is easy to see that $g \circ h$ is also an endomorphism (inner homomorphism). If ϕ and ψ are (inner) multivalued morphisms, then their product $\phi \circ \psi$ is given by $(\phi \circ \psi)(x) = \bigcup_{y \in \phi(x)} \psi(y)$. Finally, let g be an endomorphism or an inner homomorphism, and ϕ is an (inner) multivalued morphism. Then $\phi \circ g$ is given by $(\phi \circ g)(x) = \{g(y) \mid y \in \phi(x)\}$ and $g \circ \phi$ is given by $(g \circ \phi)(x) = \phi(g(x))$. Both mappings are (inner) multivalued morphisms.

We classify the values based on a type of degeneracy defined in the following way (see Example 6.13):

Definition 3.7. *For $x, y \in \mathrm{dom}(\Gamma)$, we say that x produces y in Γ if Γ has a multivalued morphism ϕ with $\phi(x) = \{0, y\}$ and $\phi(z) = \{0\}$ for every $z \neq x$.*

In other words, for every $R \in \Gamma$ and $\mathbf{t} \in R$, replacing an arbitrary subset of the x values with y and making every other coordinate 0 gives another tuple in R. Observe that the relation "x produces y" is transitive, but not necessarily reflexive. The following classification of values plays a central role in the paper:

Definition 3.8. *A value $y \in \mathrm{dom}(\Gamma)$ is*

1. regular *if there is no multivalued morphism ϕ where $0, y \in \phi(x)$ for some $x \in \mathrm{dom}(\Gamma)$,*

2. semiregular *if there is a multivalued morphism ϕ where $0, y \in \phi(x)$ for some $x \in \mathrm{dom}(\Gamma)$, but there is no $x \in \mathrm{dom}(\Gamma)$ that produces y,*

3. self-producing *if y produces y, and for every x that produces y, y also produces x.*

4. degenerate *otherwise.*

Example 6.14 demonstrates all four types. It will sometimes be convenient to say that a value y has *type* 1, 2, 3, or 4 corresponding to the four cases of Definition 3.8, and use the natural order on these types. We need some simple properties of the four types of elements.

Proposition 3.9. *If there is an endomorphism h with $h(x) = y$, then the type of x cannot be larger than that of y.*

Proof. Suppose that x is semiregular, and ψ is a multivalued morphism witnessing that, i.e., $x, 0 \in \psi(a)$ for some value a. Then $0, y \in (\psi \circ h)(a)$, meaning that y cannot be regular. Suppose that some value a produces x, let ψ be the corresponding multivalued morphism. Then $\psi \circ h$ witnesses that a produces y as well. Therefore, if x is self-producing, then y is either self-producing or degenerate. Finally, suppose that x is degenerate and let a be a value such that a produces x, but x does not produce a. Now a produces y: this is shown by taking the product $\psi \circ h$ of a multivalued morphism ψ witnessing that y produces x and the endomorphism h. If y produces a and this is witnessed by multivalued morphism ψ', then (as shown by $h \circ \psi'$) x would produce a, a contradiction. Thus a produces y, but y does not produce a, i.e., y is degenerate as well. $\square$

Proposition 3.10. *Every degenerate value y is produced by a nondegenerate value x.*

Proof. We construct a sequence $x_0, x_1, x_2, \ldots$ as follows: Let $x_0 = y$. If x_i is degenerate, then there is a value $x_{i+1} \neq x_i$ that produces x_i, but x_i does not produce x_{i+1}. As $\operatorname{dom}(\Gamma)$ is finite, either x_i is non-degenerate for some i or $x_i = x_j$ for some $i > j$. If some x_i is non-degenerate, then by transitivity such x_i produces $x_0 = y$ and we are done. Suppose that $x_i = x_j$ for some $i > j$. Since $x_{j+1} \neq x_j$, we have $i > j + 1$. Thus $x_i = x_j$ produces x_{j+1}, contradicting the definition of x_{j+1}. $\square$

Lemma 3.11. *Let Γ be a cc0-language, $R \in \Gamma$ a relation, and h an endomorphism of Γ. If $\mathbf{t}_1$ and $\mathbf{t}_2$ are disjoint tuples such that $\mathbf{t}_1, \mathbf{t}_1 + \mathbf{t}_2 \in R$, then $\mathbf{t}_1 + h(\mathbf{t}_2) \in R$.*

Proof. Without loss of generality, we assume that $\mathbf{t}_1 = (a_1, \ldots, a_r, 0, \ldots, 0)$ and $\mathbf{t}_2 = (0, \ldots, 0, b_1, \ldots, b_q, 0, \ldots, 0)$. Let R' be the relation obtained from R by substituting $a_1, \ldots, a_r$ on the first r coordinates, respectively. Since $\mathbf{t}_1 \in R$, we know that R' is 0-valid and hence it is in Γ. Thus h is an endomorphism of R' as well. From $\mathbf{t}_1 + \mathbf{t}_2 \in R$, we have $(b_1, \ldots, b_q, 0, \ldots, 0) \in R'$. Applying h on this tuple gives $(h(b_1), \ldots, h(b_q), 0, \ldots, 0) \in R'$, which implies, by definition of R', that $(a_1, \ldots, a_r, h(b_1), \ldots, h(b_q), 0, \ldots, 0) = \mathbf{t}_1 + h(\mathbf{t}_2) \in R$. $\square$

We can extend Lemma 3.11 to multivalued morphisms with the same proof:

Lemma 3.12. *Let Γ be a cc0-language, $R \in \Gamma$ a relation, and ϕ a multivalued morphism of Γ. If $\mathbf{t}_1$ and $\mathbf{t}_2$ are disjoint tuples such that $\mathbf{t}_1, \mathbf{t}_1 + \mathbf{t}_2 \in R$, then $\mathbf{t}_1 + \mathbf{t}_2' \in R$ for every $\mathbf{t}_2' \in \phi(\mathbf{t}_2)$.*

3.3 Components

The structure of endomorphisms and inner homomorphisms plays an important role in our study. The following notion helps to capture this structure.

Definition 3.13. *Let Γ be a cc0-language. A retraction to $X \subseteq D \setminus \{0\}$ is a mapping ret_X such that $\operatorname{ret}_X(x) = x$ for $x \in X$ and $\operatorname{ret}_X(x) = 0$ otherwise. A nonempty subset $C \subseteq D \setminus \{0\}$ is a component of Γ if ret_C is an endomorphism of Γ. A component C is minimal if there is no component that is a proper subset of C.*

13

Note that by definition, a component contains only nonzero values. The set $D \setminus \{0\}$ is trivially a component. If a set C is not a component, then there is a relation $R \in \Gamma$ and $\mathbf{t} \in R$ such that $\mathbf{t}' = \mathrm{ret}_C \mathbf{t} \notin R$.

We prove certain combinatorial properties of components. It is easy to see that, as the composition of retractions is a retraction, the set of components is closed under intersection (if the intersection is not empty):

Observation 3.14. *The intersection of two nondisjoint components is also a component. Hence for every nonempty $X \subseteq D \setminus \{0\}$, there is a unique inclusion-wise minimal component that contains X; this component is called the component* generated by X *(or simply the component of X).*

In case of cc0-languages, components are closed also under union:

Proposition 3.15. *If Γ is a cc0-language, then the union of two components is also a component.*

Proof. Suppose that C_1, C_2 are components. For a relation $R \in \Gamma$ and tuple $\mathbf{t}$, let $\mathbf{t}_1 = \mathrm{ret}_{C_1} \mathbf{t}$, $\mathbf{t}_2 = \mathrm{ret}_{C_2 \setminus C_1} \mathbf{t}$, $\mathbf{t}_3 = \mathrm{ret}_{D \setminus (C_1 \cup C_2 \cup \{0\})} \mathbf{t}$; clearly, $\mathbf{t} = \mathbf{t}_1 + \mathbf{t}_2 + \mathbf{t}_3$. As C_1 is a component, we have $\mathbf{t}_1 \in R$. Thus by Lemma 3.11, we have $\mathbf{t}_1 + \mathrm{ret}_{C_2}(\mathbf{t}_2 + \mathbf{t}_3) = \mathbf{t}_1 + \mathbf{t}_2 = \mathrm{ret}_{C_1 \cup C_2} \mathbf{t} \in R$. This is true for every R and $\mathbf{t}$, thus $C_1 \cup C_2$ is a component. $\square$

A consequence of Proposition 3.15 is that the component generated by a subset $X \subseteq D \setminus \{0\}$ is the union $C_X = \bigcup_{d \in X} C_d$, where C_d is the component generated by d. Indeed, C_X is a component by Proposition 3.15, and it is clear that no proper subset of C_X can be a component containing X.

Proposition 3.16. *If Γ is a cc0-language and $d \in \mathrm{dom}(\Gamma)$ is in the component generated by X for some $X \subseteq \mathrm{dom}(\Gamma) \setminus \{0\}$, then there is a $d' \in X$ such that d is in the component generated by d'.*

The difference of two components is not necessarily a component, but in this case there is a difference counterexample:

Proposition 3.17. *If C_1 and C_2 are two components of Γ such that the nonempty set $C_1 \setminus C_2$ is not a component, then Γ has a difference counterexample contained in C_1.*

Proof. As $C_1 \cap C_2$ is also a component by Observation 3.14, we may assume that $C_2 = C_1 \cap C_2$, that is, $C_2 \subseteq C_1$. As $C_1 \setminus C_2$ is not a component, there is an $R \in \Gamma$ and a $\mathbf{t} \in R$ such that $\mathbf{t}_1 = \mathrm{ret}_{C_1 \setminus C_2} \mathbf{t} \notin R$. Since both C_1 and C_2 are components, we have $\mathbf{t}_2 = \mathrm{ret}_{C_2} \mathbf{t} \in R$ and $\mathbf{t}_1 + \mathbf{t}_2 = \mathrm{ret}_{C_1} \mathbf{t} \in R$. Thus $(R, \mathbf{t}_1, \mathbf{t}_2)$ is a difference counterexample. $\square$

The following statement will be used when we restrict the language to a subset of the domain:

Proposition 3.18. *Let $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ be such that $D' \setminus \{0\}$ is a component of Γ. For every $d \in D'$,*

1. *the component generated by d is the same in Γ and $\Gamma_{|D'}$.*

2. *the type of d in $\Gamma_{|D'}$ is not greater than that in Γ.*

Proof. Let C (resp., C') be the component generated by d in Γ (resp., $\Gamma_{|D'}$). Since $D' \setminus \{0\}$ is a component containing d, we have $C \subseteq D' \setminus \{0\}$. As ret_C is an endomorphism of Γ, it is an endomorphism of $\Gamma_{|D'}$ as well, thus $C' \subseteq C$. Furthermore, as $\mathrm{ret}_{D' \setminus 0} \circ \mathrm{ret}_{C'}$ is an endomorphism of Γ, the set C' is a component of Γ, implying $C \subseteq C'$.

For the second statement, suppose that d is semiregular in $\Gamma_{|D'}$ and let ψ be a multivalued morphism witnessing this, i.e., $0, d \in \psi(c)$ for some $c \in D'$. Then $\mathrm{ret}_{D' \setminus \{0\}} \circ \psi$ witnesses that d

cannot be regular in Γ. Let us next show that for any $a, b \in D'$, a produces b in Γ if and only if a produces b in $\Gamma_{|D'}$. The forward direction follows from the fact that for any multivalued morphism ϕ of Γ, the mapping $\phi \circ \mathrm{ret}_{D' \setminus \{0\}}$ is a multivalued morphism of $\Gamma_{|D'}$, and if $\phi(a) = \{0, b\}$, then $(\phi \circ \mathrm{ret}_{D' \setminus \{0\}})(a) = \phi(a) = \{0, b\}$. For the backward direction, let ψ be a multivalued morphism of $\Gamma_{|D'}$ witnessing that a produces b. Then $\mathrm{ret}_{D'} \circ \psi$ witnesses that a produces b in Γ. It is now immediate that if d is degenerate in $\Gamma_{|D'}$, then it is degenerate in Γ as well, and if d is self-producing in $\Gamma_{|D'}$, then (as d produces itself) it cannot be regular or semiregular in Γ. $\qquad\square$

The importance of components comes from the following result: there is a counterexample to weak separability where each of $\mathbf{t}_1$ and $\mathbf{t}_2$ is contained in one component. This observation will be essential in the hardness proofs.

Lemma 3.19. *If Γ is not weakly separable, then there is a counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ which is either*

1. *a union counterexample, and $\mathbf{t}_1$ (resp., $\mathbf{t}_2$) is contained in a component generated by a value a_1 (resp., a_2), or*

2. *a difference counterexample, and both $\mathbf{t}_1$ and $\mathbf{t}_2$ are contained in a component generated by a value a_1.*

Proof. Let $K_1, \ldots, K_r$ be the distinct components generated by the nonzero values in Γ. Assume first that there are two components K_i, K_j that intersect; without loss of generality, we can assume that $K_i \setminus K_j \neq \emptyset$. Let a be a value that generates K_i. Clearly, $a \notin K_i \cap K_j$: otherwise Prop. 3.14 implies that $K_i \cap K_j \subset K_i$ is a component containing a, contradicting the assumption that a generates K_i. Thus $K_i \setminus K_j \subset K_i$ is not a component, since otherwise it would be a strictly smaller component containing a. Now Prop. 3.17 implies that there is a difference counterexample contained in the component K_i generated by a, satisfying the requirements.

Thus in what follows, we can assume that $K_1, \ldots, K_r$ are pairwise disjoint, i.e., they partition D. Suppose that there is a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$. Tuple $\mathbf{t}_1$ can be represented as a union $\mathbf{t}_{1,1} + \cdots + \mathbf{t}_{1,r_1}$ of nonzero disjoint tuples such that each $\mathbf{t}_{1,i} = \mathrm{ret}_{K_{j_i}} \mathbf{t}_1$ is contained in one of the components $K_1, \ldots, K_r$. The tuples $\mathbf{t}_{2,1}, \ldots, \mathbf{t}_{2,r_2}$ are defined similarly. Let $\mathbf{s}_1, \ldots, \mathbf{s}_{r_1 + r_2}$ be an arbitrary ordering of these $r_1 + r_2$ tuples. It is clear that each $\mathbf{s}_i$ is in R, since $K_1, \ldots, K_r$ are components. As the union of these tuples is not in R, there is an integer $j \geq 1$ such that the union of any j of these tuples is in R, but there is a subset of $j + 1$ tuples whose union is not in R. Without loss of generality, suppose that $\bigcup_{i=1}^{j+1} \mathbf{s}_i$ is not in R. If $j = 1$, then $(R, \mathbf{s}_1, \mathbf{s}_2)$ is a required counterexample. If $j > 1$ then define $\mathbf{s}_0 := \sum_{i=1}^{j-1} \mathbf{s}_i$. By assumption, $\mathbf{s}_0 \in R$, hence substituting the nonzero values of $\mathbf{s}_0$ into R as constants gives a 0-valid relation R'. Furthermore, $\mathbf{s}_0 + \mathbf{s}_j, \mathbf{s}_0 + \mathbf{s}_{j+1} \in R$ by the definition of j; let $\mathbf{s}'_j, \mathbf{s}'_{j+1} \in R'$ be the corresponding tuples. Now $(R', \mathbf{s}'_j, \mathbf{s}'_{j+1})$ is a union counterexample: tuples $\mathbf{s}'_j, \mathbf{s}'_{j+1}$ are disjoint and $\mathbf{s}'_j + \mathbf{s}'_{j+1} \notin R'$ follows from $\mathbf{s}_0 + \mathbf{s}_j + \mathbf{s}_{j+1} \notin R$.

Thus we can assume that there is no union counterexample. Suppose that there is difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$. Assume that $(R, \mathbf{t}_1, \mathbf{t}_2)$ is minimal in the sense that $\mathbf{t}_1 + \mathbf{t}_2$ has minimal number of nonzero coordinates among such counterexamples. We claim that $\mathbf{t}_1 + \mathbf{t}_2$ is contained in one of the components K_i defined above. Suppose that $\mathbf{t}_1 + \mathbf{t}_2$ contains nonzero values from components $K_1, \ldots, K_g$ for $g \geq 2$. We show that $\mathrm{ret}_{K_i}(\mathbf{t}_1) \in R$ for every $1 \leq i \leq g$. This clearly holds if $\mathrm{ret}_{K_i}(\mathbf{t}_1)$ equals $\mathrm{ret}_{K_i}(\mathbf{t}_1 + \mathbf{t}_2)$ or the zero tuple. Thus we can assume that $\mathrm{ret}_{K_i}(\mathbf{t}_1 + \mathbf{t}_2) \in R$ is the disjoint union of nonzero tuples $\mathrm{ret}_{K_i}(\mathbf{t}_1)$ and $\mathrm{ret}_{K_i}(\mathbf{t}_2) \in R$. If $\mathrm{ret}_{K_i}(\mathbf{t}_1) \notin R$, then $(R, \mathrm{ret}_{K_i}(\mathbf{t}_1), \mathrm{ret}_{K_i}(\mathbf{t}_2))$ is a difference counterexample, contradicting the minimality of $(R, \mathbf{t}_1, \mathbf{t}_2)$ as $g \geq 2$. Thus $\mathrm{ret}_{K_i}(\mathbf{t}_1) \in R$ for every i. However, the disjoint union of these tuples is also in R

(since by assumption there is no union counterexample), that is, $\mathbf{t}_1 \in R$, a contradiction. It follows that $\mathbf{t}_1 + \mathbf{t}_2$ belongs to some component K_i, i.e., there is a value a that generates K_i. $\square$

3.4 Multivalued morphism gadgets

The main technical tool in the hardness proofs are the gadgets defined in this section. Intuitively, assuming that $D = \{0, 1, \ldots, \Delta\}$, we want a gadget consisting of variables v_0, v_1, $\ldots$, v_Δ that has only two possible satisfying assignments: (1) either 0 appears on every v_i, or (2) value i appears on v_i for every i. Such gadgets would allow us to use a counterexample to weak separability to prove hardness in similar way as hardness is proved in the Boolean case (see Example 6.9). However, due to the endomorphisms of the constraint language, such a gadget is not always possible to create: a nontrivial endomorphism can be used to transform satisfying assignments into new ones.

Therefore, our goal is more modest: we want a gadget that is either fully zero or represents an endomorphism in every satisfying assignment. That is, if variable v_i gets value c_i, then the mapping h defined by $h(i) = c_i$ is an endomorphism. We enforce this requirement by introducing, for every $R \in \Gamma$ and $(a_1, \ldots, a_r) \in R$, a constraint $\langle (v_{a_1}, \ldots, v_{a_r}), R \rangle$. Such a constraint ensures that applying the mapping h given by an assignment to the tuple $(a_1, \ldots, a_r)$ gives a tuple in R.

The gadgets we use in the reductions are more general than the one described in the previous paragraph: instead of a single variable v_i representing value i, we have a bag B_i of a variables representing this value. Setting the size of these bags and the cardinality/size constraint is an essential and delicate part of the reduction. The requirement that we want to enforce now is that if $\psi(i)$ is the set of values appearing on the variables of B_i in a satisfying assignment, then ψ is a multivalued morphism of Γ. This can be ensured in a way similar to the construction in the previous paragraph (see below for details).

A minor technical detail is that we defined morphisms in such a way that 0 is always mapped to 0, thus there is no need to introduce variables representing what 0 is mapped to; it is more convenient to use constant 0's instead. We need the following definition to formulate this conveniently. For a relation R and a tuple $\mathbf{t} \in R$, we denote by $\mathsf{supp}(\mathbf{t})$ the set of coordinate positions of $\mathbf{t}$ occupied by nonzero elements. Let $\mathsf{supp}_\mathbf{t}(R)$ denote the relation obtained by substituting 0 into all coordinates of R except for $\mathsf{supp}(\mathbf{t})$, i.e. if R is r-ary and $\mathsf{supp}(\mathbf{t}) = \{1, \ldots, r\} \setminus \{i_1, \ldots, i_q\}$, then $\mathsf{supp}_\mathbf{t}(R) = R^{|i_1, \ldots, i_q; 0, \ldots, 0}$.

For a cc0-language Γ and some $0 \in D' \subseteq \mathrm{dom}(\Gamma)$, a *multivalued morphism gadget* $\mathrm{MVM}(\Gamma, D')$ consists of $|D'| - 1$ bags of vertices B_d, $d \in D' \setminus \{0\}$. The number of variables in each bag will be specified every time we use the gadget in a proof. The gadget is equipped with the following set of constraints. For every $R \in \Gamma$ and every tuple $\mathbf{t} = (a_1, \ldots, a_r) \in R_{|D'}$, we add all possible constraints $\langle \mathbf{s}, \mathsf{supp}_\mathbf{t}(R) \rangle$ where $\mathsf{supp}(\mathbf{t}) = \{i_1, \ldots, i_q\}$, $\mathbf{s} = (v_{i_1}, \ldots, v_{i_q})$, and $v_{i_j} \in B_{a_{i_j}}$ for every $1 \leq j \leq q$. The *standard assignment* of a gadget assigns a to every variable in bag B_a. Observe that the standard assignment satisfies all the constraints of the gadget. We say that bag B_a and the variables in bag B_a *represent* a. When we say that a gadget is *fully nonzero,* then we mean that all the variables are assigned nonzero values.

Proposition 3.20. *Let $0 \in D' \subseteq \mathrm{dom}(\Gamma)$. Consider a satisfying assignment f of an $\mathrm{MVM}(\Gamma, D')$ gadget. If $h_f : D' \to 2^{\mathrm{dom}(\Gamma)}$ is a mapping such that $h_f(a)$ is the set of values appearing in bag B_a of the gadget and $h_f(0) = \{0\}$, then h_f is an inner multivalued morphism of Γ from D' to $\mathrm{dom}(\Gamma)$.*

Proof. Let R be a relation of Γ and let $\mathbf{t} = (a_1, \ldots, a_r) \in R_{|D'}$. Let $(b_1, \ldots, b_r) \in h_f(\mathbf{t})$. By the definition of h_f, for every $1 \leq i \leq r$, either $a_i = b_i = 0$ or $a_i \neq 0$ and there is a variable v_i in bag B_{a_i} having value b_i. Let $\mathsf{supp}_\mathbf{t} = \{i_1, \ldots, i_{r'}\}$. The gadget is defined such that there is a constraint

$\langle(v_{i_1}, \ldots, v_{i_{r'}}), \mathsf{supp_t}(R)\rangle$, implying that $(b_{i_1}, \ldots, b_{i_{r'}}) \in \mathsf{supp_t}(R)$ and hence $(b_1, \ldots, b_r) \in R$. It follows that $h_f(\mathbf{t}) \subseteq R$. □

Note that if $D' = \mathrm{dom}(\Gamma)$, then h_f is a multivalued morphism of Γ. Mapping h_f, or any mapping h' with $h'(a) \subseteq h_f(a)$ (for $a \in D'$), is said to be an (inner) multivalued morphism *given* by the gadget and assignment f. If $|h'(a)| = 1$ for all $a \in D'$, we call it an endomorphism (inner homomorphism) *given* by the gadget and assignment f.

We define two types of gadgets connecting MVM gadgets. The gadget $\mathrm{NAND}(G_1, G_2)$ on $\mathrm{MVM}(\Gamma, D')$ gadgets G_1, G_2 consists of the following constraints. For every $R \in \Gamma$ and disjoint tuples $\mathbf{t}_1 = (a_1, \ldots, a_r)$, $\mathbf{t}_2 = (b_1, \ldots, b_r)$ in $R_{|D'}$, we add a constraint $\langle \mathbf{s}, \mathsf{supp_{t_1+t_2}}(R)\rangle$, for every $\mathbf{s} = (v_{i_1}, \ldots, v_{i_q})$ with $\{i_1, \ldots i_q\} = \mathsf{supp}(\mathbf{t}_1 + \mathbf{t}_2)$, such that v_{i_j} is in bag $B_{a_{i_j}}$ of G_1 if $a_{i_j} \neq 0$ and v_{i_j} is in bag $B_{b_{i_j}}$ of G_2 if $b_{i_j} \neq 0$.

It is not difficult to see (Lemma 3.21 below) that if one of G_1, G_2 has the standard assignment and the other is fully zero, then all the constraints in $\mathrm{NAND}(G_1, G_2)$ are satisfied. On the other hand, if both G_1 and G_2 have the standard assignment and there is a union counterexample, then $\mathrm{NAND}(G_1, G_2)$ is not satisfied. For the reductions, we need this second conclusion not only if both G_1 and G_2 have the standard assignment, but also assignments that "behave well" in some sense. The right notion for our purposes is the following: An inner homomorphism $h : D' \rightarrow \mathrm{dom}(\Gamma)$ of Γ is $\mathbf{t}$-*recoverable* for some tuple $\mathbf{t}$ if h is invertible on elements of $\mathbf{t}$ in the following sense: Γ has a multivalued morphism ϕ such that $\mathbf{t} \in (h \circ \phi)(\mathbf{t})$. In particular, this is true if there is an endomorphism h' of Γ with $(h \circ h')(\mathbf{t}) = \mathbf{t}$. We say that a $\mathrm{MVM}(\Gamma, D')$ gadget is $\mathbf{t}$-recoverable in a given assignment if at least one of the inner homomorphisms given by it is $\mathbf{t}$-recoverable.

Lemma 3.21. *Let $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ and let there be a $\mathrm{NAND}(G_1, G_2)$ gadget on $\mathrm{MVM}(\Gamma, D')$ gadgets G_1, G_2.*

1. *If one of G_1 and G_2 has the standard assignment and the other gadget is fully zero, then all constraints of $\mathrm{NAND}(G_1, G_2)$ are satisfied.*

2. *Let f be a satisfying assignment of $\mathrm{NAND}(G_1, G_2)$. If h_i is an inner homomorphism given by gadget G_i and assignment f for $i = 1, 2$ and $\mathbf{t}_1, \mathbf{t}_2$ are disjoint tuples in $R_{|D'}$, then $h_1(\mathbf{t}_1) + h_2(\mathbf{t}_2) \in R$.*

3. *If there is a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|D'}$ and gadget G_i is $\mathbf{t}_i$-recoverable in assignment f (for $i = 1, 2$), then some constraint of $\mathrm{NAND}(G_1, G_2)$ is not satisfied by f.*

Proof. 1. Suppose without loss of generality that G_1 has the standard assignment and G_2 is fully zero. Consider a relation $R \in \Gamma$ and disjoint tuples $\mathbf{t}_1 = (a_1, \ldots, a_r)$, $\mathbf{t}_2 = (b_1, \ldots, b_r) \in R_{|D'}$. In a corresponding constraint $\langle(v_1, \ldots, v_r), \mathsf{supp_{t_1+t_2}}(R)\rangle$, if $a_i \neq 0$, then variable v_i has value a_i (since it is in bag B_{a_i} of G_1) and has value 0 otherwise. Thus $(a_1, \ldots, a_r) \in R$ implies that the constraint is satisfied.

2. If the $\mathrm{NAND}(G_1, G_2)$ instance is satisfied, then one of the constraints corresponding to $\mathbf{t}_1$ and $\mathbf{t}_2$ ensures that $h_1(\mathbf{t}_1) + h_2(\mathbf{t}_2) \in R$.

3. For $i = 1, 2$, let h_i be a $\mathbf{t}_i$-recoverable inner homomorphism given by G_i and let ϕ_i be a multivalued morphism such that $\mathbf{t}_i \in (h_i \circ \phi_i)(\mathbf{t}_i)$. Since h_i is an inner homomorphism, we have that $h_1(\mathbf{t}_1), h_2(\mathbf{t}_2) \in R$. Statement 2 implies that $h_1(\mathbf{t}_1) + h_2(\mathbf{t}_2) \in R$. By Lemma 3.12, we have that $h_1(\mathbf{t}_1) + \mathbf{t}'_2 \in R$ for any $\mathbf{t}'_2 \in \phi_2(h_2(\mathbf{t}_2))$; in particular, this means that $h_1(\mathbf{t}_1) + \mathbf{t}_2$ is in R. As $\mathbf{t}_2 \in R$, we can apply Lemma 3.12 once more to get that $\mathbf{t}'_1 + \mathbf{t}_2 \in R$ for any $\mathbf{t}'_1 \in \phi_1(h_1(\mathbf{t}_1))$; in particular, $\mathbf{t}_1 + \mathbf{t}_2 \in R$, a contradiction. □

The IMP(G_1, G_2) gadget is defined similarly, but instead of $\mathbf{t}_1, \mathbf{t}_2 \in R_{|D'}$, we require $\mathbf{t}_2, \mathbf{t}_1 + \mathbf{t}_2 \in R_{|D'}$.

Lemma 3.22. *Let $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ and let there be an IMP(G_1, G_2) gadget on MVM(Γ, D') gadgets G_1, G_2.*

1. *If G_2 has the standard assignment and G_1 either has the standard assignment or fully zero, then all constraints of IMP(G_1, G_2) are satisfied.*

2. *Let f be a satisfying assignment of IMP(G_1, G_2). If h_i is an inner endomorphism given by gadget G_i and assignment f for $i = 1, 2$ and $\mathbf{t}_1, \mathbf{t}_2$ are disjoint tuples such with $\mathbf{t}_2, \mathbf{t}_1 + \mathbf{t}_2 \in R_{|D'}$, then $h_1(\mathbf{t}_1) + h_2(\mathbf{t}_2) \in R$.*

3. *If there is a difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|D'}$, G_1 is $\mathbf{t}_1$-recoverable in assignment f and G_2 gives an inner homomorphism h_2 with $h_2(\mathbf{t}_2) = 0$, then some constraint of IMP(G_1, G_2) is not satisfied.*

Proof. 1. Similarly to the proof of Lemma 3.21, the constraint corresponding to $(R, \mathbf{t}_1, \mathbf{t}_2)$ is satisfied, since $\mathbf{t}_2, \mathbf{t}_1 + \mathbf{t}_2 \in R$.

2. As in Lemma 3.21, follows from the definition of IMP(G_1, G_2).

3. Let h_1 be a $\mathbf{t}_1$-recoverable inner homomorphism given by G_1. Let ϕ_1 be a multivalued morphism such that $\mathbf{t}_1 \in (h_1 \circ \phi_1)(\mathbf{t}_1)$. One of the constraints of IMP(G_1, G_2) ensure that $h_1(\mathbf{t}_1) + h_2(\mathbf{t}_2) = h_1(\mathbf{t}_1)$ is in R. Now $\mathbf{t}_1 \in \phi_1(h_1(\mathbf{t}_1))$ is also in R, a contradiction. $\qquad\square$

When the multivalued morphism gadgets are used in the reductions, it will be essential that the bags of the gadgets have very specific sizes. We will ensure somehow that in a solution each bag is either fully zero or fully nonzero. Our aim is to choose the sizes of the bags in such a way that if the sum of the sizes of a collection of bags add up to a certain integer, then this is only possible if the collection contains exactly one bag of each size.

In most of the reductions, the MVM(Γ, D') gadgets are arranged in t groups (corresponding to the t groups of vertices in the MULTICOLORED INDEPENDENT SET or MULTICOLORED IMPLICATIONS instance we are reducing from). For a gadget in group i, the bag representing $d \in D' \setminus \{0\}$ has size $Z_{i,d}^{t,D'}$, which is defined as follows. Fix an integer t and a set $0 \in D' \subseteq D$. It will be convenient to assume that $D' = \{0, 1, \ldots, \Delta\}$. For $1 \leq i \leq t$ and $1 \leq d \leq \Delta$, we define

$$Z_{i,d}^{t,D'} := (4t\Delta)^{2t\Delta + (i\Delta + d)} + (4t\Delta)^{5t\Delta - (i\Delta + d)}.$$

By $\mathcal{Z}^{t,D'}$ we denote the set of integers $Z_{i,d}^{t,D'}$ for $1 \leq i \leq t$ and $1 \leq d \leq \Delta$. Note that these integers have exactly two nonzero digits if written in base-$(4t\Delta)$ expansion; the positions of these two digits depend on $i\Delta + d$. Furthermore, the "larger nonzero digit" of any number in $\mathcal{Z}^{t,D'}$ is always larger than the "smaller nonzero digit" of any other number in $\mathcal{Z}^{t,D'}$. We will use the following property of these integers:

Lemma 3.23. *Let us fix t and $D' = \{0, 1, \ldots, \Delta\}$. If A is a subset of $\mathcal{Z}^{t,D'}$ and $\mathcal{B}$ is a multiset of values from $\mathcal{Z}^{t,D'}$ such that $|\sum_{S \in A} S - \sum_{S \in \mathcal{B}} S| < (4t\Delta)^{2t\Delta}$, then $\mathcal{B}$ is a set and $\mathcal{B} = A$.*

Proof. It can be assumed that $A \cap \mathcal{B} = \emptyset$, since removing integers from both A and $\mathcal{B}$ does not change the difference of the sums. Let T be the largest integer in $A \cup \mathcal{B}$. Assume first that $T \in \mathcal{B} \setminus A$. This means that A contains only integers strictly smaller than T, and (as A is a set) an integer appears at most once in A. Since there are $t\Delta$ integers in $\mathcal{Z}^{t,D'}$ and every integer smaller than T is

at most $2T/(4t\Delta)$, we have that the sum of the integers in A is at most $T/2$. Thus the difference of the sums is at least $(4t\Delta)^{2t\Delta}$, a contradiction.

Assume now that $T \in A \setminus B$. Suppose that $T = Z_{i,d}^{t,D'}$ and let $X := (4t\Delta)^{2t\Delta + (i\Delta + d)}$. Since T is the largest integer in $A \cup B$ (i.e, $i\Delta + d$ is as small as possible), every integer in $A \cup B$ other than T is divisible by $(4t\Delta)^{2t\Delta + (i\Delta + d + 1)} = X \cdot 4t\Delta$, while T is equal to X modulo $X \cdot 4t\Delta$. Thus $\sum_{S \in A} S$ is X modulo $X \cdot 4t\Delta$, while $\sum_{S \in B} S$ is divisible by $X \cdot 4t\Delta$. Therefore, $|\sum_{S \in A} S - \sum_{S \in B} S|$ is at least $\min\{X, 4t\Delta \cdot X - X\} > (4t\Delta)^{2t\Delta}$, a contradiction. $\square$

3.5 Frequent instances

If a value d "can appear" only on a small number of variables, then we can branch on all possible ways this value appears and then reduce the problem to simpler instances whose domain does not contain d. Formally, we say that an instance of $\mathrm{CCSP}(\Gamma)$ or $\mathrm{OCSP}(\Gamma)$, with parameter k is c-frequent (for some integer c) if for every $d \in \mathrm{dom}(\Gamma) \setminus \{0\}$ there are at least c variables v such that $f(v) = d$ for a satisfying assignment f of size at most k (note that we do not require that these satisfying assignments satisfy the cardinality constraints). The algorithm of Lemma 2.3 can be used to decide whether an instance is c-frequent. As we shall see in Lemma 3.24, if an instance is not c-frequent, then it can be reduced to c-frequent instances by trying all possibilities for the values that appear on less than c variables. We prove a stronger result, where the instances satisfy an additional technical requirement. A subset $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ is closed (with respect to Γ) if Γ has no inner homomorphism from D' to $\mathrm{dom}(\Gamma)$ that maps some element of D' to an element in $\mathrm{dom}(\Gamma) \setminus D'$.

Lemma 3.24. *Let Γ be a finite cc0-language. Given an instance I of $\mathrm{CCSP}(\Gamma)$ or $\mathrm{OCSP}(\Gamma)$ with parameter k and an integer c, we can construct in time $f_\Gamma(k,c)n^{O(1)}$ a set of c-frequent instances such that*

1. *instance I has a solution if and only if at least one of the constructed instances has a solution,*

2. *each instance I_i is an instance of $\mathrm{CCSP}(\Gamma_{|D_i})$, respectively, $\mathrm{OCSP}(\Gamma_{|D_i})$, for some $D_i \subseteq \mathrm{dom}(\Gamma)$ closed in Γ, and*

3. *the parameter k_i of I_i is at most k.*

Proof. We state the proof for $\mathrm{CCSP}(\Gamma)$, the proof is the same for $\mathrm{OCSP}(\Gamma)$. The reduction performs the following branching step repeatedly. Let I' be the current instance, which is an instance of $\mathrm{CCSP}(\Gamma_{|D'})$ for some D'. If I' is not c-frequent, then we branch as follows. Let S_d be the set of those variables where value d can appear in a satisfying assignment of size at most k. This set can be found using the algorithm Lemma 2.3 as follows: To decide if $v \in S_d$, let us assign d to v and find if there is any minimal satisfying extension of this assignment of size at most k. Suppose that $|S_d| < c$. We branch into $\binom{|S_d|}{\pi(d)} < 2^{|S_d|} \le 2^c$ directions by considering every subset $S_d' \subseteq S_d$ of size exactly $\pi(d)$ and creating an assignment that gives value d to the variables of S_d', and 0 to the remaining variables. If this assignment does not satisfy I', then we use Lemma 2.3 to enumerate all the minimal satisfying extensions $f_1', \ldots, f_t'$ of this assignment. For each such satisfying assignment f_i', we obtain an instance I_i' by substituting the nonzero variables as constants. Since f_i' is a satisfying assignment, every relation of I_i' is 0-valid, hence it is an instance of $\mathrm{CCSP}(\Gamma_{|D'})$ as well. Furthermore, since we have already considered all possible appearances of value d, the correctness of the algorithm does not change if we consider I_i' as an instance of $\mathrm{CCSP}(\Gamma_{|D' \setminus \{d\}})$. If the instance I_i' is still not c-frequent, then we repeat the branching step.

In each step, the maximum number of directions we branch into is at most a constant depending only on c, Γ, and the current parameter $k' \leq k$. The depth of the branching tree is at most $|D|$, since we decrease the domain at each step. Thus it is clear that the running time is $f_\Gamma(k,c)n^{O(1)}$, for an appropriate function $f_\Gamma(k,c)$.

Let instance I_i of $\mathrm{CCSP}(\Gamma_{|D_i})$ be a 0-valid c-frequent instance obtained by the algorithm and let k_i be its size constraint. To show that D_i is closed, we argue as follows. Suppose that there is an inner homomorphism $g : D_i \to D$ of Γ such that $g(d) = d'$ for some $d \in D_i$, $d' \in D \setminus D_i$. Since I_i is c-frequent, there are c variables $v_1, \ldots, v_c$ and satisfying assignments $f_1, \ldots, f_c$ of size at most k_i such that $f_j(v_j) = d$. On the branch of the algorithm that produced instance I_i, there has to be an instance $I_i^{(1)}$ of $\mathrm{CCSP}(\Gamma_{|D^{(1)}})$ that is reduced to an instance $I_i^{(2)}$ of $\mathrm{CCSP}(\Gamma_{|D^{(2)}})$ such that $D_i \subseteq D^{(1)}$, $D^{(2)} = D^{(1)} \setminus \{d'\}$, and d' is not c-frequent in $I^{(1)}$. If we consider instance I_i as an instance of $\mathrm{CCSP}(\Gamma_{|D^{(1)}})$, then the assignment $f_j \circ g$ assigns value d' to v_i. Since instance I_i is obtained from instance $I^{(1)}$ via substitutions, we get that variables $v_1, \ldots, v_c$ can get value d' in $I^{(1)}$ in assignments whose size does not exceed the parameter of $I^{(1)}$. This contradicts the assumption that d' is not c-frequent in $I^{(1)}$. $\qquad\square$

4   Classification for size constraints

Unlike in the Boolean case, weak separability of Γ is not equivalent to the tractability of $\mathrm{OCSP}(\Gamma)$: it is possible that Γ is not weakly separable, but $\mathrm{OCSP}(\Gamma)$ is FPT (see Example 6.10). However, if there is a subset $D' \subseteq \mathrm{dom}(\Gamma)$ of the domain such that $\Gamma_{|D'}$ is not weakly separable and D' has "no special problems" in a certain technical sense, then $\mathrm{OCSP}(\Gamma)$ is W[1]-hard. We need the following definitions. A value $d \in \mathrm{dom}(\Gamma)$ is *weakly separable* if $\Gamma_{|\{0,d\}}$ is weakly separable. A *contraction* of Γ to D' with $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ is an endomorphism $h : \mathrm{dom}(\Gamma) \to D'$ such that $h(d) \neq 0$ for any $d \in \mathrm{dom}(\Gamma) \setminus \{0\}$. Contraction h is *proper* if $D' \subset \mathrm{dom}(\Gamma)$. As the contraction can be applied on any solution of an $\mathrm{OCSP}(\Gamma)$ instance without changing the number of nonzero variables, restriction to D' does not change the problem, i.e., replacing every R with $R_{|D'}$ does not change the solvability of the instance (see Example 6.15).

The main result for the size constraints CSP is the following dichotomy theorem.

Theorem 4.1. *Let Γ be a finite cc0-language. If there are two sets $\{0\} \subseteq D_2 \subseteq D_1 \subseteq \mathrm{dom}(\Gamma)$ such that*

1. *D_1 is closed in Γ,*

2. *$\Gamma_{|D_1}$ has a contraction h to D_2,*

3. *$\Gamma_{|D_2}$ has no proper contraction,*

4. *$\Gamma_{|D_1}$ has no weakly separable value that is either degenerate or self-producing, and*

5. *$\Gamma_{|D_2}$ is not weakly separable,*

then $\mathrm{OCSP}(\Gamma)$ is W[1]-hard. If there are no such D_1, D_2, then $\mathrm{OCSP}(\Gamma)$ is fixed-parameter tractable.

We present an algorithm solving the FPT cases of the problem in Section 4.1. Section 4.2 presents an important case of the hardness proof, demonstrating the concepts introduced in Section 3.

4.1 The algorithm

Let $I = (V, \mathcal{C}, k)$ be an instance of $\mathrm{OCSP}(\Gamma)$. Let us use the algorithm of Lemma 3.24 to obtain instances $I_1, \ldots, I_\ell$ such that I_i is a k-frequent instance of $\mathrm{OCSP}(\Gamma_{|D^i})$ for some closed set $D^i \subseteq \mathrm{dom}(\Gamma)$. Fix some i and let h be a contraction of $\Gamma_{|D^i}$ such that $|h(D^i)|$ is minimum possible. Set $D_1 := D^i$ and $D_2 := h(D^i)$.

In the cases where Theorem 4.1 claims fixed-parameter tractability, the pair D_1, D_2 violates one of the properties (1)–(5). By the way D_1 and D_2 defined, it is clear that (1) and (2) hold. For (3), suppose that $\Gamma_{|D_2}$ has a proper contraction g. Then $h \circ g$ is a contraction of $\Gamma_{|D_1}$ such that $|g(h(D_1))|$ is strictly less than $|h(D_1)|$, a contradiction.

If D_1 violates (4), then instance I_i always has a solution. Indeed, suppose that $d \in D_1$ is weakly separable and d is produced by $d' \in D_1$ (possibly $d = d'$). Let k_i be the parameter of I_i; then $k_i \leq k$ by Lemma 3.24(3). Since I_i is k-frequent, if we denote by S the set of variables of I_i where d' can appear in a satisfying assignment of size at most k, then $|S| \geq k$. As d' produces d, $\Gamma_{|D_1}$ has a multivalued morphism ϕ such that $\phi(d') = \{0, d\}$ and $\phi(a) = \{0\}$ for $a \in D_1 \setminus \{d'\}$. Applying multivalued morphism ϕ on *any* satisfying assignment f with $f(v) = d'$ shows that the assignment $\delta_{v,d}$ with $\delta_{v,d}(v) = d$ and 0 everywhere else is a satisfying assignment of I_i. Therefore, for every $v \in S$, assignment $\delta_{v,d}$ satisfies I_i. As d is weakly separable in $\Gamma_{|D_1}$, the disjoint union of k_i such assignments $\delta_{v,d}$ is a solution to I_i. Finally, suppose that (5) is violated and $\Gamma_{|D_2}$ is weakly separable. Instance I_i of $\mathrm{OCSP}(\Gamma_{|D_1})$ has a solution if and only if it has a solution restricted to D_2 (because of the contraction h), and the latter can be decided using the algorithm of Lemma 3.5.

4.2 Hardness

Suppose that we have sets D_1 and D_2 as in Theorem 4.1. The aim of this section is to show that $\mathrm{OCSP}(\Gamma)$ is W[1]-hard in this case. The reduction is based on a counterexample to the weak separability of $\Gamma_{|D_2}$, which exists by condition (5) of Theorem 4.1. To ensure that the MVM gadgets work as intended, we have to make use of conditions (1)–(4) as well. Our first goal is to handle the cases when some value in $\Gamma_{|D_1}$ is not regular (Sections 4.2.1–4.2.2). The main part of the proof is to prove hardness in the case when every value in $\Gamma_{|D_1}$ is regular (Section 4.2.3). This case contains the most important proof ideas; the reader is suggested to skim Sections 4.2.1–4.2.2 and concentrate on Section 4.2.3 on a first reading.

4.2.1 Degenerate and self-producing values

Recall that a relation R is intersection definable in a constraint language Γ if R is the set of all solutions to a certain instance of $\mathrm{CSP}(\Gamma)$. Let U_Γ be the set of all 0-valid unary relations intersection definable in the set of 0-valid relations from Γ.

Lemma 4.2. *Let D_1 be a closed set in Γ. If P is the set of nonzero values produced by $d \in D_1$ in $\Gamma_{|D_1}$, then $P \cup \{0\} \in U_\Gamma$.*

Proof. For every (r-ary) $R \in \Gamma$, every tuple $\mathbf{b} \in R_{|D_1}$ where d appears, and every subset $\mathbf{a} = (a_1, \ldots, a_r)$ of $\mathbf{b}$ that contains only 0 and d (recall that it means that $a_i = b_i$ whenever $a_i \neq 0$), we set $R_{\mathbf{a}} = R^{|i_1, \ldots, i_q; 0, \ldots, 0}$, where $i_1, \ldots, i_q$ are the positions such that $a_{i_j} = 0$. Let T be the unary relation expressed by the instance $(\{v\}, \mathcal{C})$, where $\mathcal{C}$ contains constraints $\{\langle (v, \ldots, v), R_{\mathbf{a}} \rangle\}$ for all such $R \in \Gamma$, $\mathbf{b}$, and $\mathbf{a}$. We claim that $T = P \cup \{0\}$.

For every $a \in P$, the fact that d produces a in $\Gamma_{|D_1}$ implies the tuple obtained from $\mathbf{b} \in R_{|D_1}$ by replacing value d with 0 or a and replacing everything else with 0 gives a tuple of R. Thus setting v to a is a satisfying assignment.

On the other hand, suppose that there is a satisfying assignment with value a on v. It follows that $\psi(d) = \{a, 0\}$ and $\psi(d') = \{0\}$ for every $d' \in D_1 \setminus \{d\}$ is an inner multivalued morphism ψ of Γ from D_1. Indeed, similar to the previous paragraph it means that the tuple obtained from $\mathbf{b} \in R_{|D_1}$ by replacing value d with 0 or a and replacing everything else with 0 gives a tuple of R. By Observation 3.6, the mapping h such that $h(d) = a$ and $h(d') = 0$ for every $d' \in D_1 \setminus \{d\}$ is an inner homomorphism. Since D_1 is closed, $a \in D_1$, thus d produces a in $\Gamma_{|D_1}$. $\square$

The following lemma proves hardness in the case when $\Gamma_{|D_1}$ contains some degenerate and self-producing values. In this case, by condition (4) of Theorem 4.1, none of the values are weakly separable.

Lemma 4.3. *If D_1 is a closed set such that $\Gamma_{|D_1}$ has degenerate or self-producing values, but no such value is weakly separable, then $\mathrm{OCSP}(\Gamma)$ is W[1]-hard.*

Proof. Let $d \in D_1$ be a value that produces at least one nonzero value in $\Gamma_{|D_1}$; let P be the set of (nonzero) values produced by d. By Lemma 4.2, $P \cup \{0\} \in U_\Gamma$. Let $P' \subseteq P$ be a smallest nonempty set such that $P' \cup \{0\} \in U_\Gamma$.

Let x be an arbitrary nonzero element of P'. As x is produced by d, the assumption of the lemma implies that x is not weakly separable in $\Gamma_{|D_1}$. Suppose first that $\Gamma_{|\{0,x\}}$ has a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$, where $R \in \Gamma_{|\{0,x\}}$ is r-ary. Let A, B be the set of coordinates of R such that $\mathbf{t}_1$ equals x in positions from A, $\mathbf{t}_2$ equals x in positions from B, and they are equal to 0 otherwise. By substituting 0's, it can be assumed without loss of generality that $A \cup B = \{1, \dots, r\}$ and $A = \{1, \dots, q\}$. Let R' be the binary relation expressed by the instance $(\{v, w\}, \{\langle(v, \dots, v, w, \dots, w), R\rangle\})$ where v occupies the first q positions. As is easily seen, $(0,0), (x,0), (0,x) \in R'$, but $(x,x) \notin R'$. We show that $(y, y') \notin R'$ for arbitrary nonzero values $y, y' \in P'$. If this is true, then by Proposition 2.7 this binary relation R' and the unary relation restricting to $P' \cup \{0\}$ can be used to reduce INDEPENDENT SET to $\mathrm{OCSP}(\Gamma)$. That is, a binary constraint with relation R' can represent each edge: this ensures that that a set of variables can be simultaneously nonzero if and only if they correspond to an independent set in the graph.

Suppose that $(y, y') \in R'$ for some $y, y' \in P'$ (including the possibility that y or y' is equal to x). If $(y, 0) \notin R'$, then by substituting 0 in the second coordinate we get a 0-valid unary relation that includes x, but does not include y, contradicting the minimality of P' (the intersection of this set with $P' \cup \{0\}$ is a proper subset of $P' \cup \{0\}$). It follows that $(y, x) \in R'$: otherwise, by substituting y in the first coordinate, we would get a 0-valid relation containing y', but not x. Finally, by substituting x in the second coordinate, we get a 0-valid unary relation containing y, but not x, a contradiction.

Suppose next that $\Gamma_{|\{0,x\}}$ has a difference counterexample. Again, without loss of generality, it can be assumed that $(0,0), (0,x), (x,x) \in R'$, but $(x,0) \notin R'$ for a binary relation R' intersection definable in Γ. We claim that $(y, 0) \notin R'$ for any nonzero $y \in P'$: otherwise, by substituting 0 in the second coordinate, we would get a 0-valid unary relation containing y, but not x. Now by Proposition 2.7, we can use this binary relation R' to reduce IMPLICATIONS to $\mathrm{OCSP}(\Gamma)$. $\square$

4.2.2 Semiregular values

Thus in the following, we assume that $\Gamma_{|D_1}$ has no degenerate or self-producing values. Next we prove hardness in the case when there is a semiregular value $d \in D_1$ in $\Gamma_{|D_1}$. We say that a multivalued morphism ϕ *witnesses* that d is semiregular if $0, d \in \phi(c)$ for some $c \in \mathrm{dom}(\Gamma)$.

Lemma 4.4. *If D_1 is a closed set in Γ such that there are no self-producing or degenerate values in $\Gamma_{|D_1}$, but there is a semiregular value $d \in D_1$, then $\mathrm{OCSP}(\Gamma)$ is W[1]-hard.*

Proof. From the semiregular values of $\Gamma_{|D_1}$, let us choose d and its witness ϕ such that the size of $S := \phi(D_1)$ is minimum possible (note that $0 \in S$).

The proof is by reduction from IMPLICATIONS. Let G, t be an instance of IMPLICATIONS. For each vertex v_i of G, we introduce a gadget $\mathrm{MVM}(\Gamma, S)$ denoted by G_i. The size of each bag of each gadget is $Z := t + 1$, except the bag corresponding to d, whose size is 1. We set the cardinality constraint to $k := tZ(|S|-2)+t$. To finish the construction of the instance, we encode the directed edges of the IMPLICATIONS instance by adding the gadget $\mathrm{IMP}(G_x, G_y)$ for each directed edge $\overrightarrow{v_x v_y}$.

Suppose that there is a solution C of size exactly t for the IMPLICATIONS instance. If vertex v_i is in C, then set the standard assignment on gadget G_i. It is clear that this results in an assignment of size exactly $k = tZ(|S|-2)+t$ and the constraints of the $\mathrm{MVM}(\Gamma, S)$ gadgets are satisfied. From the fact that there is no directed edge $\overrightarrow{v_x v_y}$ with $v_x \in C$, $v_y \notin C$ and from Lemma 3.22(1), it follows that the constraints of $\mathrm{IMP}(G_x, G_y)$ are also satisfied.

For the other direction, we have to show that if there is a solution of size exactly k for the $\mathrm{OCSP}(\Gamma)$ instance, then there is a solution C of size exactly t for IMPLICATIONS. Observe first that only values from D_1 can appear in a solution: if some value $c \in D \setminus D_1$ appears on a gadget G_x, then there is a corresponding inner homomorphism g_x of Γ from S to D such that for a certain inner homomorphism from D_1 to D given by $\phi \circ g_x$, value c appears in the image of D_1, contradicting the fact that D_1 is closed.

Next we show that if the variable in bag B_d of G_x is nonzero, then G_x is fully nonzero. Suppose that the variable in bag B_d of G_x is nonzero, but there is a nonzero variable in bag B_c for some some nonzero $c \in S$ (since there is a single variable in bag B_d, we have $c \neq d$). Let g be an inner homomorphism of $\Gamma_{|D_1}$ (from S) given by G_x such that $g(d) \neq 0$ and $g(c) = 0$. The multivalued morphism $\phi \circ g$ witnesses that $g(d)$ is semiregular in $\Gamma_{|D_1}$ (as there are no self-producing or degenerate values by assumption), and $(\phi \circ g)(D_1) = g(S)$ has size strictly less than S, contradicting the minimality of S.

Let us show that for every v_x and $c \in S \setminus \{0, d\}$, the bag B_c of G_x is either fully zero or fully nonzero. Suppose that both 0 and d' appear in this bag. Let ψ'_x be an inner multivalued morphism of $\Gamma_{|D_1}$ from S given by G_x such that $\psi'_x(c) = \{0, d'\}$ and $|\psi'_x(c')| = 1$ for every $c' \in S$, $c' \neq c$. The multivalued morphism $\phi \circ \psi'_x$ witnesses that d' is semiregular in $\Gamma_{|D_1}$ (note that by assumption, there are no self-producing or degenerate values in $\Gamma_{|D_1}$). Thus the minimality of S would be violated by $\psi'_x(d) = \{0\}$, hence the variable in bag B_d is nonzero. In this case, by the previous paragraph, the nonzero variable in bag B_d implies that every variable of G_x is nonzero.

Since the bags not corresponding to d have size Z and the cardinality constraint k equals t modulo Z, there have to be at least t gadgets where bag B_d is nonzero. We have seen that in these gadgets all the other $Z(|S|-2)$ variables are nonzero as well. Thus it follows that there are exactly t gadgets where all the variables are nonzero, and every variable of every other gadget is zero.

Let C be the set such that $v_x \in C$ if and only if the variables of G_x are nonzero; the previous paragraph implies that $|C| = t$. We claim that C is a solution for the IMPLICATIONS instance. Suppose that there is an edge $\overrightarrow{v_x v_y}$ with $v_x \in C$ and $v_y \notin C$. Let c be the value of the variable of G_x in the bag B_d. We arrive to a contradiction by showing that in this case c is produced by some value d' in $\Gamma_{|D_1}$ (recall that by assumption, there are no self-producing or degenerate values in $\Gamma_{|D_1}$). Let $d' \in D_1$ be such that $0, d \in \phi(d')$. To show that d' produces c, let $\mathbf{t} \in R$ for some $R \in \Gamma_{|D_1}$, and let $\mathbf{t}_c$ be a tuple such that c is the only nonzero value appearing in $\mathbf{t}_c$ and whenever c appears in some coordinate of $\mathbf{t}_c$, then d' appears in the same coordinate of $\mathbf{t}$. If we show that every such $\mathbf{t}_c$ is in R, then we prove that d' produces c. Let $\mathbf{t}_d$ be the same as $\mathbf{t}_c$ with every c replaced by d. The multivalued morphism ϕ shows that there is a tuple $\mathbf{t}' \in R_{|S}$ disjoint from $\mathbf{t}_c$ such that $\mathbf{t}_d + \mathbf{t}' \in R_{|S}$. Gadget G_x gives an inner homomorphism f_x with $f_x(d) = c$ and gadget G_y gives an inner homomorphism f_y with $f_y(\mathbf{t}') = \mathbf{0}$, the zero tuple. As both $\mathbf{t}' \in R_{|S}$ and

$\mathbf{t}_d + \mathbf{t}' \in R_{|S}$ hold, it follows from Lemma 3.22(2) that the $\mathrm{IMP}(G_x, G_y)$ constraint implies that $f_x(\mathbf{t}_d) + f_y(\mathbf{t}') = f_x(\mathbf{t}_d) = \mathbf{t}_c \in R$, and we are done. □

4.2.3 Regular values

Let D_1, D_2 be a pair satisfying (1)–(5) of Theorem 4.1. By previous sections, we can assume in the following, that every value is regular in $\Gamma_{|D_1}$. It follows that every value is regular in $\Gamma_{|D_2}$ as well: if ψ is a multivalued morphism of $\Gamma_{|D_2}$ with $0, d \in \psi(c)$ for some nonzero $c, d \in D_2$, then, as h is a contraction, $h \circ \psi$ witnesses that d is not regular in $\Gamma_{|D_1}$.

A technical tool in the proofs is that given a set of endomorphisms that are disjoint in the sense that the image of a value is nonzero in exactly one of the endomorphisms, we would like to construct a mapping that is the "sum" of these mapping, hoping that it is also an endomorphism. Formally, we say that a set $p_1, \ldots, p_\ell$ of endomorphisms of Γ is a *partition set* if, for every $d \in D \setminus \{0\}$, $p_i(d) \neq 0$ for exactly one i. The *sum* of the partition set is the mapping $h : D \to D$ defined such that $h(d)$ is the unique nonzero value in $p_1(d), \ldots, p_\ell(d)$. The partition set is *good* if the sum of these pairwise disjoint endomorphisms is also an endomorphism; otherwise, the partition set is *bad*. We can define partition sets similarly for inner endomorphisms.

The hardness proofs are simpler if we assume that there are no bad partition sets: we prove W[1]-hardness under this assumption in Lemma 4.5 below if there is a union counterexample and in Lemma 4.6 if there is a difference counterexample.

Note that if a partition set is bad, then there is a union counterexample in Γ using the values $\bigcup_{i=1}^{\ell} p_i(\mathrm{dom}(\Gamma))$. Indeed, suppose that there is a relation $R \in \Gamma$ and a tuple $\mathbf{t} \in R$ such that $h(\mathbf{t}) = p_1(\mathbf{t}) + p_2(\mathbf{t}) + \cdots + p_\ell(\mathbf{t}) \notin R$. If $1 < \ell' \leq \ell$ is the smallest value such that $p_1(\mathbf{t}) + p_2(\mathbf{t}) + \cdots + p_{\ell'}(\mathbf{t}) \notin R$, then $\mathbf{t}_1 = p_1(\mathbf{t}) + p_2(\mathbf{t}) + \cdots + p_{\ell'-1}(\mathbf{t})$ and $\mathbf{t}_2 = p_{\ell'}(\mathbf{t})$ is a union counterexample. Lemma 4.7 exploits this union counterexample to show hardness in case there is a bad partition set, completing the proof of Theorem 4.1.

Lemma 4.5. *If every value is regular in $\Gamma_{|D_2}$, there is no bad partition set in $\Gamma_{|D_2}$, and there is a union counterexample in $\Gamma_{|D_2}$, then $\mathrm{OCSP}(\Gamma)$ is W[1]-hard.*

Proof. The reduction is from MULTICOLORED INDEPENDENT SET (see Section 2.2). Assume $D_2 = \{0, \ldots, \Delta\}$. For each vertex $v_{x,y}$ ($1 \leq x \leq t$, $1 \leq y \leq n$), we introduce a gadget $\mathrm{MVM}(\Gamma, D_2)$ denoted by $G_{x,y}$. The bag of $G_{x,y}$ corresponding to value $d \in D_2 \setminus \{0\}$ has size $Z_{x,d}^{t,D_2}$. The size constraint is $k := \sum_{x=1}^{t} \sum_{d \in D_2 \setminus \{0\}} Z_{x,d}^{t,D_2}$. If $v_{x,y}$ and $v_{x',y'}$ are adjacent, then we add the gadget $\mathrm{NAND}(G_{x,y}, G_{x',y'})$. Furthermore, for every $1 \leq x \leq t$, $1 \leq y, y' \leq n$, $y \neq y'$, we add the $\mathrm{NAND}(G_{x,y}, G_{x,y'})$ gadget.

Suppose that there is a solution C for the MULTICOLORED INDEPENDENT SET instance. If vertex $v_{x,y}$ is in C, then set the standard assignment on gadget $G_{x,y}$, otherwise set the zero assignment. It is clear that this results in an assignment satisfying the size constraint. The constraints of the $\mathrm{MVM}(\Gamma, D_2)$ gadgets are satisfied and the constraints of $\mathrm{NAND}(G_{x,y}, G_{x,y'})$ are satisfied as well (by Lemma 3.21(1)).

For the other direction, suppose that there is a solution satisfying the size constraint. First, we observe that a solution contains values only from D_1. Indeed, if $c \notin D_1$ appears in bag B_d of a gadget $G_{x,y}$, then $G_{x,y}$ gives an inner homomorphism g of Γ from D_2 with $g(d) = c$. Now $h \circ g$ maps a value of D_1 to c, contradicting the assumption that D_1 is a closed set. Furthermore, by applying the contraction h on a solution, it can be assumed that only values from D_2 are used. Thus the $\mathrm{MVM}(\Gamma, D_2)$ gadgets give multivalued morphisms of $\Gamma_{|D_2}$. Since every value is regular in $\Gamma_{|D_2}$, each bag is either fully zero or fully nonzero. The sizes of the nonzero bags add up exactly

to the size constraint k. Thus by Lemma 3.23, the only way this is possible if there is exactly one nonzero bag with size $Z_{x,d}^{t,D_2}$ for every $1 \le x \le t$ and $d \in D_2 \setminus \{0\}$. However, the nonzero bags of sizes Z_{x,d_1}^{t,D_2} and Z_{x,d_2}^{t,D_2} could appear in different gadgets.

Take a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|D_2}$; by Lemma 3.19, we can assume that $\mathbf{t}_1$, $\mathbf{t}_2$ are in the components of $\Gamma_{|D_2}$ generated by some $a_1, a_2 \in D_2$, respectively. We show that for every $1 \le x \le t$, there are values y_x^1 and y_x^2 such that every endomorphism of $\Gamma_{|D_2}$ given by G_{x,y_x^1} (resp., G_{x,y_x^2}) is $\mathbf{t}_1$-recoverable (resp., $\mathbf{t}_2$-recoverable). For a fixed x, let $g_1, \ldots, g_n$ be arbitrary endomorphisms of $\Gamma_{|D_2}$ given by $G_{x,1}, \ldots, G_{x,n}$, respectively. Since the sizes of nonzero bags are all different, these endomorphisms are pairwise disjoint and hence they form a partition set. As there is no bad partition set in $\Gamma_{|D_2}$, their sum g is an endomorphism of $\Gamma_{|D_2}$ and in fact a contraction. Since $\Gamma_{|D_2}$ has no proper contraction by assumption, g has to be a permutation and hence g^s is the identity for some $s \ge 1$. There is a unique $1 \le y_x^1 \le n$ such that $g_{y_x^1}(a_1) = g(a_1) \ne 0$. The endomorphism $g_{y_x^1} \circ g^{s-1}$ maps a_1 to $(g \circ g^{s-1})(a_1) = g^s(a_1) = a_1$ and maps every $a \in D_2$ either to 0 or a; i.e., $g_{y_x^1} \circ g^{s-1} = \operatorname{ret}_S$ for some set $S \subseteq D_2 \setminus \{0\}$ containing a_1. As S is a component containing a_1, it has to contain the component generated by a_1 and S contains every value of $\mathbf{t}_1$. It follows that $g_{y_x^1}$ given by $G_{y_x^1}$ is $\mathbf{t}_1$-recoverable: $g^{s-1}(g_{y_x^1}(\mathbf{t}_1)) = \mathbf{t}_1$. A similar argument works for y_x^2, thus the required values y_x^1, y_x^2 exist. Let us observe that it is not possible that $y_x^1 \ne y_x^2$: by Lemma 3.21(3) the constraints of $\operatorname{NAND}(G_{x,y_x^1}, G_{x,y_x^2})$ are not satisfied in this case. Let C contain vertex $v_{x,y}$ if $y = y_x^1 = y_x^2$. It follows that C is a multicolored independent set: if vertices $v_{x,y}$, $v_{x',y'}$ are adjacent, then again by Lemma 3.21(3), some constraint of $\operatorname{NAND}(G_{x,y}, G_{x',y'}) = \operatorname{NAND}(G_{x,y_x^1}, G_{x',y_x^2})$ is not satisfied. $\square$

The proof using a difference counterexample is similar:

Lemma 4.6. *If every value is regular in $\Gamma_{|D_2}$, there is no bad partition set in $\Gamma_{|D_2}$, and there is a difference counterexample in $\Gamma_{|D_2}$, then $\operatorname{OCSP}(\Gamma)$ is W[1]-hard.*

Proof. The reduction is from MULTICOLORED IMPLICATIONS. For each vertex $v_{x,y}$ ($1 \le x \le t$, $1 \le y \le n$), we introduce a gadget $\operatorname{MVM}(\Gamma, D_2)$ denoted by $G_{x,y}$. The bag B_d of $G_{x,y}$ has size $Z_{x,d}^{t,D_2}$. The size constraint is $k := \sum_{i=x}^{t} \sum_{d \in D_2 \setminus \{0\}} Z_{x,d}^{t,D_2}$. If there is a directed edge $\overrightarrow{v_{x,y}, v_{x',y'}}$, then we add a constraint $\operatorname{IMP}(G_{x,y}, G_{x',y'})$.

Suppose that there is a solution C of size exactly t for the MULTICOLORED IMPLICATIONS instance. If vertex v_i is in C, then set the standard assignment on gadget G_{v_i}, otherwise set the zero assignment. It is clear that this results in an assignment satisfying the size constraint. The constraints of the $\operatorname{MVM}(\Gamma, D_2)$ gadgets are satisfied and the $\operatorname{IMP}(G_{x,y}, G_{x,y'})$ constraints are satisfied as well (Lemma 3.22(1)).

For the other direction, suppose that there is a solution satisfying the size constraint. As in Lemma 4.5, we can assume that only values from D_2 are used in the solution. Since every value is regular in $\Gamma_{|D_2}$, every bag is either fully zero or fully nonzero. The sizes of the nonzero bags add up exactly to the size constraint k. Thus by Lemma 3.23, the only way this is possible is if there is exactly one nonzero bag of size $Z_{x,d}^{t,D_2}$ for every $1 \le x \le t$ and $d \in D_2$.

Let us choose a difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|D_2}$; by Lemma 3.19, we can assume that $\mathbf{t}_1 + \mathbf{t}_2$ is in the component C_1 of $\Gamma_{|D_2}$ generated by some $a_1 \in D_2$. We show that for every $1 \le x \le t$, there is an integer y_x such that G_{x,y_x} is $\mathbf{t}_1$-recoverable.

Let $g_1, \ldots, g_n$ be arbitrary endomorphisms given by $G_{x,1}, \ldots, G_{x,n}$, respectively. The uniqueness of the sizes of the nonzero bags implies that these endomorphisms are pairwise disjoint and they form a partition set. We assumed that there is no bad partition set in $\Gamma_{|D_2}$, thus the sum g of the set is an endomorphism. Since $\Gamma_{|D_2}$ has no proper contraction, we have that g is a permutation

25

and g^s is the identity for some $s \geq 1$. There is a $1 \leq y_x \leq n$ such that $g_{y_x}(a_1) \neq 0$. The homomorphism $g_{y_x} \circ g^{s-1}$ maps every value $a \in D_2$ either to 0 or a; i.e., $g_{y_x} \circ g^{s-1} = \mathrm{ret}_S$ for some set $S \subseteq D_2$ containing a_1. This means that S is a component containing a_1, hence $C_1 \subseteq S$. It follows that g_{y_x} is $\mathbf{t}_1$-recoverable. Moreover, as the bag B_a of G_{x,y_x} for $a \in C_1$ is fully nonzero, we have that bag B_a of $G_{x,y}$ for $a \in C_1$ and $y \neq y_x$ is fully zero.

Let $C = \{v_{1,y_1}, \ldots, v_{t,y_t}\}$. It follows immediately that C does not violate any of the implications: if there is a directed edge $\overrightarrow{v_{x,y_x} v_{x',y'}}$, $y' \neq y_{x'}$, then the gadget G_{x,y_x} is $\mathbf{t}_1$-recoverable and $h(\mathbf{t}_2) = 0$ for every homomorphism given by $G_{x',y'}$, as $\mathbf{t}_2$ is contained in C_1, and therefore Lemma 3.22(3) implies that $\mathrm{IMP}(G_{x,y_x}, G_{x',y'})$ is not satisfied. $\qquad\square$

The last step is to handle the case when there is a bad partition set. As mentioned earlier, this implies that there is a union counterexample; the following proof exploits this fact.

Lemma 4.7. *If every value is regular in $\Gamma_{|D_2}$ and there is a bad partition set in $\Gamma_{|D_2}$, then $\mathrm{OCSP}(\Gamma)$ is* W[1]*-hard.*

Proof. Let $p_1, \ldots, p_\ell$ be a minimal bad partition set of $\Gamma_{|D_2}$ in the sense that $D_3 := \bigcup_{i=1}^\ell p_i(D_2)$ has minimum size. Assume $D_3 = \{0, \ldots, \Delta\}$. Because of the bad partition set, $\Gamma_{|D_3}$ contains a union counterexample. Furthermore, every value $d \in D_3 \setminus \{0\}$ is regular in $\Gamma_{|D_3}$: if $\Gamma_{|D_3}$ has a multivalued morphism ψ with $0, d \in \psi(c)$ for some $c \in D_3$, and $p_i(c') = c$ for some $1 \leq i \leq \ell$ and $c' \in D_2$, then $p_i \circ \psi$ witnesses that d is not regular in $\Gamma_{|D_2}$.

The reduction is the same as in Lemma 4.5, with the only difference is that we use $\mathrm{MVM}(\Gamma, D_3)$ gadgets instead of $\mathrm{MVM}(\Gamma, D_2)$ and the sizes of the bags are set using the values $Z_{x,d}^{t,D_3}$. It remains true that a solution for MULTICOLORED INDEPENDENT SET implies a solution for the $\mathrm{OCSP}(\Gamma)$ instance.

For the other direction, let us argue first that only values from D_1 appear in a solution. Suppose that a value $d \notin D_1$ appears in bag B_c of a gadget $G_{x,y}$, which means that $G_{x,y}$ gives an inner homomorphism g from D_3 to D with $g(c) = d$. Let $1 \leq s \leq \ell$ be such that $p_s(c') = c$ for some $c' \in D_2$. Now $h \circ p_s \circ g$ is an inner homomorphism from D_1 to D mapping a value of D_1 to d, contradicting the assumption that D_1 is a closed set. Furthermore, by applying the contraction h on a solution, it can be assumed in the following that only values from D_2 appear in the solution. That is, each multivalued gadget describes an inner multivalued morphism from D_3 to D_2.

We show that every bag is either fully zero or fully nonzero. Suppose that ψ is an inner multivalued morphism of $\Gamma_{|D_2}$ from D_3 given by a gadget with $0, d \in \psi(c)$ for some nonzero $c \in D_3$ and $d \in D_2$. Suppose that $p_s(c') = c$ for some $c' \in D_2$ and $1 \leq s \leq \ell$. Now $p_s \circ \psi$ witnesses that d is not regular in $\Gamma_{|D_2}$, and this contradiction shows that every bag is either zero or fully nonzero. The sizes of the nonzero bags add up exactly to the size constraint k. Thus by Lemma 3.23, there is exactly one nonzero bag with size $Z_{x,d}^{t,D_3}$ for every $1 \leq x \leq t$ and $d \in D_2 \setminus \{0\}$.

We know that there is a union counterexample in $\Gamma_{|D_3}$ (because of the bad partition set whose image is in D_3). Let us choose a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|D_3}$; by Lemma 3.19, we can assume that $\mathbf{t}_i$ is in the component of $\Gamma_{|D_3}$ generated by some $a_i \in D_3$, for $i = 1, 2$. We show that for every $1 \leq x \leq t$, there are values y_x^1 and y_x^2 such that G_{x,y_x^1} (resp., G_{x,y_x^2}) gives a $\mathbf{t}_1$-recoverable (resp., $\mathbf{t}_2$-recoverable) inner homomorphism from D_3 to $\Gamma_{|D_2}$.

Let p be the sum of this bad partition set $p_1, \ldots, p_\ell$ (note that p is *not* an endomorphism of $\Gamma_{|D_2}$). The uniqueness of the sizes of the nonzero bags imply that at most $|D_3| - 1$ of the gadgets $G_{x,1}, \ldots, G_{x,n}$ have nonzero bags. Furthermore, if we choose one inner homomorphism given by each such gadget, then it is clear that these inner homomorphisms $g_1, \ldots, g_m$ form a partition set, i.e., for any $a \in D_3$, the value $g_i(a)$ is nonzero for exactly one i. Let g be the sum of $g_1, \ldots, g_m$ (note that we have no reason to assume that g is an inner homomorphism from D_3 to $\Gamma_{|D_2}$).

We show that $g \circ p$ is a permutation of D_3. Let P be the set of all endomorphisms of $\Gamma_{|D_2}$ that arise in the form $p_{z_1} \circ g_{z_2} \circ p_{z_3}$ for some $1 \le z_1, z_3 \le \ell$, $1 \le z_2 \le m$. Observe that for every $a \in D_2$, there is a unique triple (z_1, z_2, z_3) such that $(p_{z_1} \circ g_{z_2} \circ p_{z_3})(a)$ is nonzero: this follows from the fact that both $p_1, \ldots, p_\ell$ and $g_1, \ldots, g_m$ are partition sets. Thus the endomorphisms in P also form a partition set. Let p^* be the sum of this set. We have $p^*(D_2) \subseteq (g \circ p)(D_3)$: if $p^*(a) = b$, then there is an $a' \in D_3$ with $p_{z_1}(a) = a'$ and $(g_{z_2} \circ p_{z_3})(a') = b$ for some z_1, z_2, z_3. If $g \circ p$ is not a permutation of D_3, then $(g \circ p)(D_3)$ has size strictly smaller than $|D_3|$, and hence $p^*(D_2)$ has size strictly smaller than $|D_3|$ as well. If p^* is an endomorphism of $\Gamma_{|D_2}$, then (as there are no proper contractions by assumption) it has to be a permutation, contradicting $|p^*(D_2)| < |D_3| \le |D_2|$. Otherwise, suppose that p^* is not an endomorphism, i.e., the partition set P is bad. Now $|p^*(D_2)| < |D_3| = |p(D_2)|$ contradicts the minimality of the bad partition set $p_1, \ldots, p_\ell$.

Since $g \circ p$ is a permutation, there is an $s \ge 1$ such that $(g \circ p)^s$ is the identity. This means that for an arbitrary sequence $u_1, u_1', \ldots, u_s, u_s'$, the endomorphism $(g_{u_1} \circ p_{u_1'} \circ \cdots \circ g_{u_s} \circ p_{u_s'})$ is a retraction ret_S of $\Gamma_{|D_3}$, and we can choose the sequence such that $a_1 \in S$. As S is a component containing a_1, it contains all the values of $\mathbf{t}_1$. It follows that g_{u_1} is $\mathbf{t}_1$-recoverable, hence we can set $y_x^1 := u_1$. The values y_x^2 can be defined similarly. From this point, we can finish the proof as in Lemma 4.5. $\square$

5 Classification for cardinality constraints

The characterization of the complexity of $\mathrm{CCSP}(\Gamma)$ requires a new definition, which was not relevant for $\mathrm{OCSP}(\Gamma)$. The *core* of Γ is the component generated by the set of all nondegenerate values in $\mathrm{dom}(\Gamma)$. Note that by Proposition 3.10, the set of nondegenerate values is not empty, and thus the core is not empty. We say that Γ is a core if the core of Γ is $\mathrm{dom}(\Gamma)$ (see Example 6.16).

Lemma 5.1. *Let Γ be a finite cc0-language over D. If $C \subseteq D$ is the core of Γ, then $\Gamma_{|C \cup \{0\}}$ is a core.*

Proof. Every nondegenerate value of Γ is in C. By Lemma 3.18(2), every such value is nondegenerate also in $\Gamma_{|C \cup \{0\}}$ and by Lemma 3.18(1), they generate the same component C in $\Gamma_{|C \cup \{0\}}$ as in Γ. Thus $\Gamma_{|C \cup \{0\}}$ is a core. $\square$

The statement of the classification theorem for CCSP is actually simpler than for OCSP: we prove hardness if some core is not weakly separable.

Theorem 5.2. *Let Γ be a finite cc0-language. If there is a set D' with $0 \in D' \subseteq \mathrm{dom}(\Gamma)$ such that $\Gamma_{|D'}$ is a core and not weakly separable, then $\mathrm{CCSP}(\Gamma)$ is Biclique-hard, and fixed-parameter tractable otherwise.*

Note that our proof shows W[1]-hardness in most of the cases: there is only one specific situation in the proof (Lemma 5.10) where only Biclique-hardness is shown. One can extract from the proof the following sufficient condition for proving W[1]-hardness:

Corollary 5.3. *Let Γ be a core that is not weakly separable and minimal in the sense that there is no subset $0 \in D' \subset \mathrm{dom}(\Gamma)$ such that $\Gamma_{|D'}$ is a core and not weakly separable. If Γ contains at least one semiregular or regular value, then $\mathrm{CCSP}(\Gamma)$ is W[1]-hard.*

5.1 The algorithm

We present an algorithm solving the FPT cases of the problem. The algorithm consists of three steps. First, as a preprocessing step, we use Lemma 3.24 to ensure that every value is "frequent." Next we solve the problem restricted to the core, which is weakly separable by our assumption and hence the algorithm of Lemma 3.5 can be used. Finally, we show that by a postprocessing step, we can extend the solution on the core to the original domain. For this last step, we need the following lemma. For a set of variables let $\delta_{v,d}$ be the assignment that assigns value d to variable v and 0 to every other variable. If Γ is weakly separable, then satisfying assignments of this form can be freely combined together (as there is no union counterexample). The following lemma shows something similar under the weaker assumption that $\Gamma_{|D'}$ is weakly separable whenever it is a core.

Lemma 5.4. *Suppose that for every D' with $0 \in D' \subseteq \mathrm{dom}(\Gamma)$, if $\Gamma_{|D'}$ is a core, then it is weakly separable. Let I be an instance of $\mathrm{CCSP}(\Gamma)$ having the following property: for every nonzero $d \in \mathrm{dom}(\Gamma)$, there are at least $k|\mathrm{dom}(\Gamma)|$ variables v such that $\delta_{v,d}$ is a satisfying assignment. Then I has a solution satisfying the cardinality constraints and such a solution can be found in polynomial time.*

Proof. We prove the statement by induction on $|\mathrm{dom}(\Gamma)|$; for $\mathrm{dom}(\Gamma) = \{0\}$, we have nothing to show. Let K be the core of Γ and π the cardinality constraint in I. By Lemma 5.1, $\Gamma_{|K \cup \{0\}}$ is a core, hence weakly separable by assumption. For every $d \in K$, let V_d be the set of those variables v for which $\delta_{v,d}$ is a satisfying assignment. Since $|V_d| \geq k|\mathrm{dom}(\Gamma)|$, with greedy selection we can find a $V'_d \subseteq V_d$ of size exactly $\pi(d)$ for every $d \in K$ such that these sets are pairwise disjoint. Consider the assignment f that assigns, for every $d \in K$, value d to every variable of V'_d and 0 to every variable that is not in $S := \bigcup_{d \in K} V'_d$. Since f can be obtained as the disjoint union of assignments $\delta_{v,d}$ with $v \in V'_d$ and $\Gamma_{|K \cup \{0\}}$ is weakly separable, we have that f is a satisfying assignment. Let $I' = (V', \mathcal{C}', \pi')$ be the 0-valid instance obtained by substituting the nonzero values of f as constants. Note that $\pi'(d) = 0$ for every $d \in K$, since f assigns value d to exactly $\pi(d)$ variables. It is clear that if I' has a solution, then I has a solution.

For any $v \in V'$ and d, let $\delta'_{v,d}$ be the assignment of I' that assigns d to variable v and 0 to every other variable. By definition, every value in $\mathrm{dom}(\Gamma) \setminus K$ is degenerate in Γ. Thus by Proposition 3.10, for every $c \in \mathrm{dom}(\Gamma) \setminus K$, there is a $d \in K$ such that d produces c in Γ. We claim that $\delta'_{v,c}$ is a satisfying assignment of I' for any variable $V_d \setminus S$. Using the weak separability of $\Gamma_{|K \cup \{0\}}$, we get that $f + \delta_{v,d}$ is a satisfying assignment of I for any $v \in V_d \setminus S$. Thus $\delta'_{v,d}$ on V' is a satisfying assignment of I', and, using that fact that d produces c in Γ, we get that $\delta'_{v,c}$ is a satisfying assignment of I'. As $|S| \leq k$, there are at least $|V_d| - k \geq |\mathrm{dom}(\Gamma)|k - k = (|\mathrm{dom}(\Gamma)| - 1)k \geq |\mathrm{dom}(\Gamma) \setminus K| \cdot k$ variables v such that $\delta'_{v,c}$ is a satisfying assignment of I'. Since $\pi'(d) = 0$ for every $d \in K$, instance I' can be viewed as an instance of $\mathrm{CCSP}(\Gamma_{|\mathrm{dom}(\Gamma) \setminus K})$. Thus we can apply the induction hypothesis to conclude that I' has a solution. $\square$

Lemma 5.5. *Suppose that for every D' with $0 \in D' \subseteq \mathrm{dom}(\Gamma)$, if $\Gamma_{|D'}$ is a core, then it is weakly separable. Then $\mathrm{CCSP}(\Gamma)$ is fixed-parameter tractable.*

Proof. Let $I = (V, \mathcal{C}, k, \pi)$ be an instance of $\mathrm{CCSP}(\Gamma)$. Set $F := k^2(|\mathrm{dom}(\Gamma)| + d_\Gamma(k))$, where $d_\Gamma(k)$ is the function from Lemma 3.3. Let us use the algorithm of Lemma 3.24 to obtain instances I_1, ..., I_ℓ such that I_i is an F-frequent instance of $\mathrm{CCSP}(\Gamma_{|D_i})$ for some set $D_i \subseteq \mathrm{dom}(\Gamma)$.

Consider an instance $I_i = (V_i, \mathcal{C}_i, k_i, \pi_i)$. Let K_i be the core of $\Gamma_{|D_i}$. Let $I'_i = (V'_i, \mathcal{C}'_i, k'_i, \pi'_i)$ be the instance restricted to $K_i \cup \{0\}$, that is, every constraint $\langle \mathbf{s}, R \rangle \in \mathcal{C}_i$ is replaced by $\langle \mathbf{s}, R_{|K_i \cup \{0\}} \rangle$, and $\pi'_i(d) = \pi_i(d)$ for $d \in K_i$ and $\pi'_i(d) = 0$ otherwise. Note that the retraction ret_{K_i} ensures that I'_i is F-frequent as well (by definition K_i, is a component). We show that I_i has a solution if and

28

only if I_i' has. As CCSP$(\Gamma_{|K_i \cup \{0\}})$ is weakly separable by assumption, the algorithm of Theorem 3.5 can be used to check in fpt-time whether I_i' has a solution.

The retraction ret_{K_i} shows that if I_i has a solution f, then I_i' has a solution $f' = pr_{K_i}f$. For the other direction, let f' be a solution of I_i' and let I_i'' be the instance of CCSP$(\Gamma_{|D_i})$ obtained from I_i by substituting the nonzero values of f' as constants. Since f' satisfies π_i', the cardinality constraint is 0 in instance I_i'' for every $d \in K_i$. Thus I_i'' can be viewed as an CCSP$(\Gamma_{|D_i \setminus K_i})$ instance. We show that the conditions of Lemma 5.4 hold for I_i'' (viewed as an CCSP$(\Gamma_{|D_i \setminus K_i})$ instance), hence it has a solution f''. This means that solution f' of I_i' can be extended by f'' to obtain a solution f of I_i.

Let $c \in D_i \setminus K_i$. By Proposition 3.10, there is a $d \in K_i$ producing c in $\Gamma_{|D_i}$. As I_i is F-frequent, I_i has distinct variables $v_1, \ldots, v_F$ and (not necessarily distinct) satisfying assignments $g_1, \ldots, g_F$ of size at most k such that $g_j(v_j) = d$. We can assume that each g_j is contained in $K_i \cup \{0\}$ (as $d \in K_i$ and ret_{K_i} is an endomorphism of $\Gamma_{|D_i}$). Since each g_j has size at most k, there are at least F/k distinct assignments in the sequence $g_1, \ldots, g_F$. By Lemma 3.4(2), we can assume that every g_j is a minimal assignment. By Lemma 3.3, each nonzero variable of f' is nonzero in at most $d_\Gamma(k)$ minimal assignments of size at most k. Hence, among the F/k distinct minimal assignments, there are at most $k \cdot d_\Gamma(k)$ assignments nondisjoint with f', that is, there are at least $F/k - k \cdot d_\Gamma(k) \geq |\mathrm{dom}(\Gamma)|k$ assignments disjoint with f'. Let us consider such an assignment g_j. As g_j and f' are disjoint, both use only values from K_i, and $\Gamma_{|K_i \cup \{0\}}$ is weakly separable, their sum is a satisfying assignment. This means that I_i'' has a satisfying assignment where v_j has value d. Using the fact that d produces c, it follows that $\delta_{v_j,c}$ is a satisfying assignment of I_i''. Thus for every $c \in D_i \setminus K_i$, there are at least $|\mathrm{dom}(\Gamma)|k$ variables v such that $\delta_{v,c}$ is a satisfying assignment of I_i''. By Lemma 5.4, this means that I_i'' has a solution. $\square$

5.2 Hardness

A crucial difference between OCSP(Γ) and CCSP(Γ) is that for every $0 \in D' \subseteq \mathrm{dom}(\Gamma)$, it is trivial to reduce CCSP$(\Gamma_{|D'})$ to CCSP(Γ). Indeed, a CCSP$(\Gamma_{|D'})$ instance can be interpreted as a CCSP(Γ) instance with $\pi(d) = 0$ for every $d \in \mathrm{dom}(\Gamma) \setminus D'$.

Proposition 5.6. *If* CCSP$(\Gamma_{|D'})$ *is* W[1]-*hard for some* $0 \in D' \subseteq \mathrm{dom}(\Gamma)$, *then* CCSP$(\Gamma)$ *is* W[1]-*hard.*

In particular, if $\Gamma_{|\{0,a\}}$ is not weakly separable for some value $a \in D$, then the result of [24] on the Boolean case implies that CCSP$(\Gamma_{|\{0,a\}})$ and hence CCSP(Γ) are W[1]-hard (see also Example 6.9).

Proposition 5.6 allows us to assume that the language Γ satisfies the hardness condition of Theorem 5.2, but no restriction $\Gamma_{|D'}$ satisfies it for any $0 \in D' \subset \mathrm{dom}(\Gamma)$. That is, Γ is a core and not weakly separable, but every core $\Gamma_{|D'}$ with $0 \in D' \subset \mathrm{dom}(\Gamma)$ is weakly separable. Indeed, if $0 \in D' \subset \mathrm{dom}(\Gamma)$ is a set such that $\Gamma_{|D'}$ is a core and not weakly separable, then it is sufficient to prove hardness for the constraint language $\Gamma_{|D'}$ and the hardness for Γ follows by Prop. 5.6.

We proceed in the following way. Lemma 5.8 of Section 5.2.1 proves W[1]-hardness in the case when there is a semiregular value in Γ. Section 5.2.2 considers the case when every element is degenerate or self-producing. The main part of the proof appears in Section 5.2.3, where we prove W[1]-hardness using a counterexample involving regular values; as in Section 4, the reader is encouraged to focus on this part of the proof. The proof of a technical claim is deferred to Section 5.2.4.

5.2.1 Semiregular values

In the case when there is a semiregular value, we can identify a difference counterexample and use it to simulate the constraints in an IMPLICATIONS instance. We say that a multivalued morphism ϕ *witnesses* that d is semiregular if $0, d \in \phi(c)$ for some $c \in \mathrm{dom}(\Gamma)$.

Lemma 5.7. *If Γ contains a semiregular value, then there is a difference counterexample. Moreover, if ϕ witnesses that d is semiregular, then there is a difference counterexample in $\Gamma_{|\phi(\mathrm{dom}(\Gamma))}$.*

Proof. Suppose that $0, d \in \phi(c)$. As d is semiregular, no value produces d. In particular, c does not produce d, thus there is a relation $R \in \Gamma$, a tuple $\mathbf{t} \in R$, and a nonzero tuple $\mathbf{t}_d \notin R$ such that d is the only nonzero value appearing in $\mathbf{t}_d$ and at every coordinate where d appears in $\mathbf{t}_d$, value c appears in the same coordinate of $\mathbf{t}$. Applying ϕ on $\mathbf{t}$ and turning each c into 0 yields a tuple $\mathbf{t}' \in R$ disjoint from $\mathbf{t}_d$. Applying ϕ on $\mathbf{t}$ also shows that $\mathbf{t}' + \mathbf{t}_d \in R$: instead of turning each c into 0, we can turn it to either 0 or d (depending on the tuple $\mathbf{t}_d$). Now $(R, \mathbf{t}_d, \mathbf{t}')$ is a difference counterexample in $\Gamma_{|\phi(\mathrm{dom}(\Gamma))}$. $\qquad\square$

Lemma 5.8. *Let Γ be a core. If there is a semiregular value d in Γ, then $\mathrm{CCSP}(\Gamma)$ is W[1]-hard.*

Proof. Let $\psi : \mathrm{dom}(\Gamma) \to 2^{\mathrm{dom}(\Gamma)}$ be a multivalued morphism witnessing that d is semiregular. Let us choose d and ψ such that

1. the size of $\{a \in \mathrm{dom}(\Gamma) \mid \psi(a) \neq \{0\}\}$ is minimum possible, and

2. among such d and ψ, the size of $S := \psi(\mathrm{dom}(\Gamma))$ is minimum possible.

Observe that we can assume that $\psi(c) = \{0, d\}$ for a unique value $c \in \mathrm{dom}(\Gamma)$ and $|\psi(a)| = 1$ for every $a \neq c$. Furthermore, we can assume that d cannot be produced by any $a \in S$ in $\Gamma_{|S}$. Otherwise, if $\psi_d : S \to 2^S$ is the multivalued morphism witnessing that $a \in S$ produces d in $\Gamma_{|S}$, then $\psi \circ \psi_d$ witnesses that d is produced by some $a' \in \mathrm{dom}(\Gamma)$ in Γ, hence d is not semiregular in Γ. Let $S_1 \subseteq S \setminus \{d\}$ contain the regular and semiregular values in $\Gamma_{|S}$ other than d and let $S_2 \subseteq S$ contain the self-producing and degenerate values (thus $S = S_1 \cup S_2 \cup \{d, 0\}$). Note that by Lemma 5.7, there is a difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|S}$.

We show that $\mathrm{CCSP}(\Gamma_{|S})$ is W[1]-hard, hence (by Proposition 5.6) $\mathrm{CCSP}(\Gamma)$ is W[1]-hard as well. The proof is by reduction from IMPLICATIONS. For each vertex v_i of G, we introduce a gadget $\mathrm{MVM}(\Gamma_{|S}, S)$ denoted by G_i. The size of bag B_c of each gadget is $Z := 2t|S|$ if $c \in S_1$ and it is 1 if $c \in S_2 \cup \{d\}$. We set the cardinality constraint $\pi'(c) = tZ$ for every $c \in S_1$ and $\pi'(c) = t$ for every $c \in S_2 \cup \{d\}$, i.e., the parameter k equals $tZ|S_1| + t|S_2| + t$. To finish the construction of the instance, we encode the directed edges of the IMPLICATIONS instance by adding the constraint $\mathrm{IMP}(G_x, G_y)$ for each directed edge $\overrightarrow{v_x v_y}$ of G.

Suppose that there is a solution C of size exactly t for the IMPLICATIONS instance. If vertex v_i is in C, then set the standard assignment on gadget G_i. It is clear that this results in an assignment of size exactly $tZ|S_1| + t|S_2| + t$ and the constraints of the $\mathrm{MVM}(\Gamma_{|S}, S)$ gadgets are satisfied. From the fact that there is no directed edge $\overrightarrow{v_x v_y}$ with $v_x \in C$, $v_y \notin C$ and from Lemma 3.22(1), it follows that the constraints of $\mathrm{IMP}(G_x, G_y)$ are also satisfied.

For the other direction, we have to show that if there is a solution of the $\mathrm{CCSP}(\Gamma)$ instance satisfying the cardinality constraint, then there is a solution C of size exactly t for IMPLICATIONS.

We show first that if a value $c \in S_1 \cup \{d\}$ appears in bag B_d of G_x, then every variable of G_x is nonzero. Suppose that c appears on the variable in bag B_d, but 0 appears in some variable of bag $B_{d'}$ of G_x for some $d' \in S$. As bag B_d contains only a single variable, we have $d' \neq d$. Let g be an endomorphism of $\Gamma_{|S}$ given by G_x such that $g(d) = c$ and $g(d') = 0$. Now the

multivalued morphism $\psi \circ g$ witnesses that $c = g(d)$ is semiregular in Γ: value $c \in S_1 \cup \{d\}$ is not self-producing or degenerate, hence $\psi \circ g$ cannot show that c is produced by some value. Moreover, $(\psi \circ g)(\mathrm{dom}(\Gamma)) = g(S)$ has size strictly less than $|S|$, contradicting either the first or second minimality condition of ψ. It follows that there can be at most t gadgets where the bag corresponding to d contains a value from $S_1 \cup \{d\}$: otherwise there would be more than $t(Z|S_1| + |S_2| + 1)$ nonzero variables.

Next we show that if a value $c \in S_1 \cup \{d\}$ appears in a bag $B_{c'}$ of a gadget G_x for some $c' \in S_1$, then every variable of that bag is nonzero. Otherwise, let ψ' be a multivalued morphism of $\Gamma_{|S}$ given by G_x with $\psi'(c') = \{0, c\}$ and $|\psi'(b)| = 1$ for every $b \neq c'$. Let h be the endomorphism of Γ defined such that $h(a) = a'$ if $\psi(a) = \{a'\}$ and $h(a) = 0$ if $\psi(a) = \{0, d\}$. Now $h \circ \psi'$ cannot witness that c is produced by some value (as $c \in S_1 \cup \{d\}$), hence it witnesses that c is semiregular in Γ and $|\{a \in \mathrm{dom}(\Gamma) \mid \psi'(a) \neq \{0\}\}| < |\{a \in \mathrm{dom}(\Gamma) \mid \psi(a) \neq \{0\}\}|$, contradicting the minimality of d and ψ.

Consider those bags that contain values from $S_1 \cup \{d\}$. Each such bag represents a value in $S_1 \cup \{d\}$: if $c \in S_1 \cup \{d\}$ appeared in a bag representing a value from S_2, then by Proposition 3.9, c would be self-producing or degenerate in $\Gamma_{|S}$. We have seen that at most t bags representing d can contain values from $S_1 \cup \{d\}$. The total cardinality constraint of these values is $t|S_1|Z + t$. Thus at least $t|S_1|$ bags representing S_1 contain values from $S_1 \cup \{d\}$. Moreover, there are exactly $t|S_1|$ such bags: as shown in the previous paragraph, these bags are fully nonzero, thus $t|S_1| + 1$ such bags would mean that the size of the assignment is at least $(t|S_1| + 1)Z > tZ|S_1| + t|S_2| + t$. As $\sum_{c \in S_1 \cup \{d\}} \pi'(c)$ is exactly $tZ|S_1| + t$, it follows that there are exactly t gadgets where the variable in bag B_d has a nonzero value from $S_1 \cup \{d\}$. We have already observed that the variables of these t gadgets are fully nonzero, and the cardinality constraint π' imply that every variable of every other gadget is zero.

Let us construct the set C such that $v_x \in C$ if and only if the variables of G_x are nonzero; the previous paragraph implies that $|C| = t$. We claim that C is a solution for the IMPLICATIONS instance. Suppose that there is an edge $\overrightarrow{v_x v_y}$ with $v_x \in C$ and $v_y \notin C$. If $v_x \in C$ and h is an endomorphism of $\Gamma_{|S}$ given by G_x, then h has to be a permutation of S: otherwise, $\psi \circ h$ witnesses that $h(d)$ is semiregular and $|(\psi \circ h)(\mathrm{dom}(\Gamma))|$ is strictly less than $|S|$, contradicting the choice of d and ψ (note that h maps every nonzero value to a nonzero value, thus $\psi \circ h$ cannot witness that $h(d)$ is produced by some element). We have seen that there is a difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in $\Gamma_{|S}$. Since h is a permutation, $h^s(\mathbf{t}_1) = \mathbf{t}_1$ for some $s \geq 1$, i.e., h is $\mathbf{t}_1$-recoverable. Thus if G_y is fully zero, then $\mathrm{IMP}(G_x, G_y)$ is not satisfied by Lemma 3.22(3). $\qquad\square$

5.2.2 Self-producing values

In this section, we consider the case when every element is either self-producing or degenerate. By Proposition 3.10, there is at least one self-producing element. It is not hard to see that the component generated by self-producing elements contains only self-producing elements. Indeed, the component generated by a self-producing element $d \in D$ equals $\{d\}$, and by Proposition 3.15 the union of components is a component. Lemmas 5.9 and 5.10 consider the two possibilities in this case: when there is a difference counterexample, and when all counterexamples are union.

Lemma 5.9. *Let Γ be a core and let $(R, \mathbf{t}_1, \mathbf{t}_2)$ be a difference counterexample to weak separability satisfying the conditions of Lemma 3.19, and a_1 is self-producing. Then $\mathrm{CCSP}(\Gamma)$ is W[1]-hard.*

Proof. Since a_1 is self-producing, $\{a_1\}$ is a component, hence a_1 is the only nonzero value appearing in $\mathbf{t}_1$ and $\mathbf{t}_2$. This means that $\Gamma_{|\{0, a_1\}}$ is not weakly separable, hence $\mathrm{CCSP}(\Gamma)$ is W[1]-hard. $\qquad\square$

Lemma 5.10. *Let Γ be a core and let $(R, \mathbf{t}_1, \mathbf{t}_2)$ be a union counterexample satisfying the conditions of Lemma 3.19 such that a_1 and a_2 are self-producing. Then $\text{CCSP}(\Gamma)$ is* BICLIQUE-*hard.*

Proof. We assume that a_1 and a_2 are weakly separable, otherwise we are done. As a_i is self-producing, $\{a_i\}$ is the component generated by a_i, hence $\mathbf{t}_i$ is contained in $\{0, a_i\}$.

First, we show that there is no inner homomorphism h_{12} from $\{0, a_1\}$ with $h_{12}(a_1) = a_2$ and there is no inner homomorphism h_{21} from $\{0, a_2\}$ with $h_{21}(a_2) = a_1$. Note that a_1 produces itself, thus the existence of h_{12} would mean that a_1 produces a_2. Since a_2 is self-producing, this would imply that a_2 produces a_1, and hence h_{21} exists as well. A symmetrical argument shows that the existence of h_{21} implies the existence of h_{12}. Suppose that both homomorphisms exist. In this case, $\mathbf{t}_1 + h_{21}(\mathbf{t}_2) \in R$ follows from $\mathbf{t}_1, h_{21}(\mathbf{t}_2) \in R$ and from the fact that a_1 is weakly separable. By using $\mathbf{t}_1 \in R$ and Lemma 3.11, we get $\mathbf{t}_1 + h_{12}(h_{21}(\mathbf{t}_2)) = \mathbf{t}_1 + \mathbf{t}_2 \in R$, a contradiction.

We reduce BICLIQUE (see Section 2.2) to $\text{CCSP}(\Gamma_{|\{0,a_1,a_2\}})$. Consider the gadget $\text{MVM}(\Gamma_{|\{0,a_1,a_2\}}, \{0, a_1\})$ where the bag B_{a_1} contains only one variable. Setting this variable to 0 or a_1 is a satisfying assignment of the gadget. However, there is no inner homomorphism h from $\{0, a_1\}$ to $\{0, a_1, a_2\}$ with $h(a_1) = a_2$, thus the variable cannot have value a_2. Thus the unary relation $U_1 = \{0, a_1\}$ is intersection definable in $\Gamma_{|\{0,a_1,a_2\}}$ and the same holds for the unary relation $U_2 = \{0, a_2\}$.

Since $(R, \mathbf{t}_1, \mathbf{t}_2)$ is a union counterexample, we first obtain a relation R' from R by substituting constant 0 to positions in which both $\mathbf{t}_1$ and $\mathbf{t}_2$, and then identifying variables to intersection define a binary relation R'' such that $(0, 0), (a_1, 0), (0, a_2) \in R''$, but $(a_1, a_2) \notin R''$. Let us consider the binary relation R''' represented by the CSP instance $(\{x, y\}, \mathcal{C}')$ where $\mathcal{C}' = \{\langle(x, y), R''\rangle, \langle(x), U_1\rangle, \langle(y), U_2\rangle\}$. Clearly, this relation is intersection definable in $\Gamma_{|\{0,a_1,a_2\}}$. It is easy to see that $(0, 0), (a_1, 0), (0, a_2) \in R'''$ and R''' contains no other tuple. Thus as observed in Example 6.1, $\text{CCSP}(R''')$ is equivalent to BICLIQUE. $\square$

5.2.3 Regular values

A significant difference between the hardness proofs of $\text{OCSP}(\Gamma)$ and $\text{CCSP}(\Gamma)$ is that it can be assumed in the case of $\text{OCSP}(\Gamma)$ that no proper contraction exists and this assumption can be used to show that certain endomorphisms have to be permutations. In Section 4.2, we used such arguments to show that gadgets are $\mathbf{t}$-recoverable. For $\text{CCSP}(\Gamma)$, we cannot make this assumption, thus the proof is based on a delicate argument (Claim 5.14), making use of the cardinality constraint, to achieve a similar effect. The following lemma is not used directly in the proof, but it demonstrates how we can deduce in some cases that a multivalued morphism gadget essentially behaves as if it had the standard assignment. Recall that for $D = \{0, 1, \ldots, \Delta\}$, we defined in Section 3.4 the constants

$$Z_{i,d}^{t,D} = (4t\Delta)^{2t\Delta+(i\Delta+d)} + (4t\Delta)^{5t\Delta-(i\Delta+d)}.$$

Lemma 5.11. *Let Γ be a finite constraint language over $D = \{0, 1, \ldots, \Delta\}$. Consider an instance consisting of a single $\text{MVM}(\Gamma, D)$ gadget where bag B_b has size $Z_{1,b}^{1,D}$. Let the cardinality constraint be $\pi(b) = Z_{1,b}^{1,D}$. If ϕ is the maximal multivalued morphism given by the gadget in a solution, then there is a $p \geq 1$ such that $b \in \phi^{p'}(b)$ for every $p' \geq p$ and nonzero $b \in D$. In particular, the gadget is $\mathbf{t}$-recoverable for any tuple $\mathbf{t}$.*

Proof. We prove the statement by induction on b. Suppose that for every $a < b$, there is a p_a such that $a \in \phi^{p'}(a)$ for every $p' \geq p_a$ (this statement is vacuously true if b is the smallest nonzero value). Let $\phi_b = \text{ret}_{\{0,1,\ldots,b\}} \circ \phi$. Let $T = \bigcup_{p \geq 1} \phi_b^p(b)$, that is, those values that can be reached from b by repeated applications of ϕ_b (note that T can contain values larger than b, but because of $\text{ret}_{\{0,1,\ldots,b\}}$ in ϕ_b, such values can appear only during the last application of ϕ_b). As the total

cardinality constraint is exactly the number of variables, all the variables are nonzero. The bag B_b and the bags B_a for $a < b$ and $a \in T$ contain nonzero values only from T by definition. The total size of these bags is

$$\sum_{a \in T, a < b} Z_{1,a}^{1,D} + Z_{1,b}^{1,D}. \tag{1}$$

We claim that $b \in T$. Otherwise, the total cardinality constraint of the values in T is

$$\sum_{a \in T, a < b} Z_{1,a}^{1,D} + \sum_{a \in T, a > b} Z_{1,a}^{1,D}. \tag{2}$$

As $Z_{1,b}^{1,D} > |D| Z_{1,a}^{1,D}$ for every $a > b$, the second term in (1) is strictly larger than the second term of (2), a contradiction. Thus $b \in T$, and therefore $b \in \phi_b^s(b)$ for some $s \geq 1$. Consider the smallest such s. If $s = 1$, then $b \in \phi_b^p(b)$ for every $p \geq 1$. Otherwise, as $b \in \phi_b^{s-1}(\phi_b(b))$, there is some $a < b$, $a \in T$ such that $a \in \phi_b(b)$ and $b \in \phi_b^{s-1}(a)$. By the induction assumption, $a \in \phi^{p'}(a)$ for every $p' \geq p_a$. This means that $b \in \phi^{1+p'+s-1}(b)$ for every $p' \geq p_a$, or in other words, $b \in \phi^{p'}(a)$ for every $p' \geq p_a + s$. Thus $p_b := p_a + s$ proves the induction statement.

To see that the gadget is $\mathbf{t}$-recoverable, observe that $b \in \phi^{p+1}(b)$ implies that there is a $c_b \in \phi(b)$ such that $b \in \phi^p(c_b)$. Let h be the endomorphism that maps each $b \in D$ to such a c_b (note that h is an endomorphism, as it is a subset of ϕ). Then h is t-recoverable, as witnessed by ϕ^p. □

Thus in a sense we can assume that a gadget has the standard assignment, even if the values appearing in the bags are arbitrary. However, the situation is more complicated in an actual hardness proof, where there are several gadgets and moreover value 0 can also appear in some of the bags. The following lemma contains the most generic part of the hardness proof of Theorem 5.2: we are proving hardness using a counterexample to weak separability.

Lemma 5.12. *Let Γ be a core that is not weakly separable, $\Gamma_{|D'}$ is weakly separable for every core $\Gamma_{|D'}$ with $0 \in D' \subset \operatorname{dom}(\Gamma)$, there are no semiregular values in Γ, and there is a regular value in Γ. Then $\operatorname{CCSP}(\Gamma)$ is W[1]-hard.*

Proof. By Lemma 3.19, there is a counterexample with values contained in the component C_1 generated by some value $a_1 \in \operatorname{dom}(\Gamma)$, or with values in $C_1 \cup C_2$, where C_1 (resp., C_2) is the component generated by some value a_1 (resp., a_2). If a_1 is degenerate, then (as Γ is a core) a_1 is in the component generated by the nondegenerate values. Thus by Proposition 3.16, there is a nondegenerate a_1' such that a_1 is in the component C_1' generated by a_1'. Since the intersection of components is also a component, we have $C_1 \subseteq C_1'$. Thus we can assume that a_1 and a_2 are nondegenerate. If a_1 and a_2 are both self-producing, then $C_1 = \{a_1\}$, $C_2 = \{a_2\}$, and hence $\Gamma_{|\{0,a_1,a_2\}}$ is not weakly separable. By Prop. 3.15, $C_1 \cup C_2$ is also a component. Therefore, as a_1 and a_2 are nondegenerate in $\Gamma_{|\{0,a_1,a_2\}}$ by Lemma 3.18(2), we have that $\Gamma_{|\{0,a_1,a_2\}}$ is a core. Thus $\operatorname{dom}(\Gamma) = \{0, a_1, a_2\}$ by the minimality of Γ, implying that there is no regular value in $\operatorname{dom}(\Gamma)$, a contradiction. Similarly, if the counterexample is contained in the component $C_1 = \{a_1\}$ generated by the self-producing value a_1, then $\operatorname{dom}(\Gamma) = \{0, a_1\}$ and again there is no regular value in Γ. By assumption, there are no semiregular values. This means that there are only three cases to consider: we have a counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ that is

1. a union or difference counterexample such that $\mathbf{t}_1 + \mathbf{t}_2$ is contained in the component of a regular value a_1;

2. a union counterexample such that $\mathbf{t}_1$ (resp., $\mathbf{t}_2$) is contained in the component of some regular value a_1 (resp., regular value a_2);

3. a union counterexample such that $\mathbf{t}_1$ (resp., $\mathbf{t}_2$) is contained in the component of some regular value a_1 (resp., some self-producing value a_2);

We present a unified W[1]-hardness proof for the three cases. The reduction is from MULTICOLORED INDEPENDENT SET in the case of a union counterexample, while we are reducing from MULTICOL-ORED IMPLICATIONS in the case of a difference counterexample. Let $v_{x,y}$ $(1 \leq x \leq t, 1 \leq y \leq n)$ be the vertices of the graph in the instance we are reducing from. Let $D := \mathrm{dom}(\Gamma)$; we assume that $D = \{0, \ldots, \Delta\}$. For each vertex $v_{x,y}$, we introduce an $\mathrm{MVM}(\Gamma, D)$ gadget $G_{x,y}$. Without loss of generality, we can assume that the nonzero values in D are ordered such that the regular values precede all the nonregular values. The bag B_d of $G_{x,y}$ has size

$$z_{x,d} = \begin{cases} Z_{x,d}^{t,D} & \text{if } d \text{ is regular,} \\ (4t\Delta)^{2t\Delta-(x\Delta+d)} & \text{otherwise,} \end{cases}$$

where $Z_{x,d}^{t,D}$ is as defined in Section 3.4. Observe that the size of any bag representing a regular value is more than $4t\Delta$ times larger than the size of any bag representing a nonregular value. The cardinality constraint $\pi(d)$ is set to be $\sum_{x=1}^{t} z_{x,d}$. Observe that the cardinality constraint of any regular value is larger than the total cardinality constraint of all the nonregular values.

If the reduction is from MULTICOLORED INDEPENDENT SET, then we introduce a $\mathrm{NAND}(G_{x,y}, G_{x',y'})$ constraint for each edge $v_{x,y}v_{x',y'}$ of the graph. Furthermore, for every $1 \leq x \leq t$ and $1 \leq y, y' \leq n$, $y \neq y'$, we introduce a constraint $\mathrm{NAND}(G_{x,y}, G_{x,y'})$. If the reduction is from MULTICOLORED IM-PLICATIONS, then we introduce a $\mathrm{IMP}(G_{x,y}, G_{x',y'})$ constraint for each edge $\overrightarrow{v_{x,y}v_{x',y'}}$ of the graph. This completes the description of the reduction.

It is easy to see that the reduction works in one direction. Let S be a set of vertices that form a solution for the instance we are reducing from. If $v_{x,y} \in S$, then let us give the standard assignment to the gadget $G_{x,y}$. The fact that S contains exactly one vertex of each color implies that the resulting assignment satisfies the cardinality constraints. Furthermore, if S is an independent set, then by Lemma 3.21(1), all the $\mathrm{NAND}(G_{x,y}, G_{x',y'})$ constraints are satisfied; if S is solution of MULTICOLORED IMPLICATIONS, then by Lemma 3.22(1), all the $\mathrm{IMP}(G_{x,y}, G_{x',y'})$ constraints are satisfied.

For the other direction of the proof, we have to show that multivalued morphisms given by the gadgets have certain properties that allow us to invoke Lemma 3.21(3) or Lemma 3.22(3). For this purpose, we show that for $i = 1, 2$ and for every x, $1 \leq x \leq t$, there is an y_x^i such that values from the component generated by a_i appear only on gadget G_{x,y_x^i} and do not appear on $G_{x,y}$ for any $y \neq y_x^i$. Furthermore, we have to show that G_{x,y_x^i} is $\mathbf{t}_i$-recoverable. The proof of these claims are based on the properties of regular values and the way the cardinality constraints are defined.

Claim 5.13. *Let K be the component generated by a regular value d and let K^* be the regular values in K.*

1. *If $b \in K^*$ appears in bag B_a of some gadget $G_{x,y}$, then every value appearing in the bag is from K^*.*

2. *For every $1 \leq x \leq t$, there is a unique $1 \leq w_{x,d} \leq n$ such that values from K^* appear in bag B_d of $G_{x,w_{x,d}}$ (and by (1), every value in bag B_d of $G_{x,w_{x,d}}$ is nonzero and from K^*).*

Proof. If value $b \in K^*$ appears in bag B_a, then a has to be regular as well by Proposition 3.9. Furthermore, every variable of B_a has to be nonzero and has to belong to K: otherwise the gadget would give a multivalued morphism ϕ such that $0, b \in (\phi \circ \mathrm{ret}_K)(a)$, contradicting the assumption that b is regular.

34

We show next that every value appearing in B_a is actually from K^*, i.e., regular. Let $\mathcal{B}$ be the multiset consisting of the sizes of the bags containing values from K^* (as observed, each such bag represents a regular value hence its size is from $\mathcal{Z}^{t,D}$) and let $A = \{Z^{t,D}_{x,a} \mid 1 \le x \le t,\ a \in K^*\}$. Note that the sum of the numbers in A is exactly the sum of cardinality constraint $\pi(a)$ for every regular value $a \in K^*$. The sum of the numbers in $\mathcal{B}$ cannot be less than that, but it might be larger: the bags where values from K^* appear might contain nonregular values from K as well (but no value outside K). However, the sum of cardinality constraints $\pi(a)$ of the nonregular values a is at most $\Delta(4t\Delta)^{2t\Delta-(\Delta+1)} < (4t\Delta)^{2t\Delta}$. Thus Lemma 3.23 can be used to conclude that $A = \mathcal{B}$ and it follows that the bags where values from K^* appear contain only values from K^*.

The second statement is also an immediate consequence of $A = \mathcal{B}$: there is exactly one bag with size $Z^{t,D}_{x,d}$ where values from K^* appear. $\qquad\square$

Let $\phi_{x,d}$ be the maximal multivalued morphism given by $G_{x,w_{x,d}}$, where $w_{x,d}$ is defined by Claim 5.13(2). Our aim is to show that then ϕ_{x,a_i} is $\mathbf{t}_i$-recoverable. For this purpose, we prove the following claim, which is the main technical ingredient of the proof. The proof idea was demonstrated in Lemma 5.11, but here we need additional arguments to handle zero values appearing on variables, values not in K appearing on variables, and the fact that there are gadgets having bags of different sizes. The proof of the following claim is delicate and technical, hence we defer it to Section 5.2.4 to maintain the flow of the proof.

Claim 5.14. *Let K be the component generated by a regular value d and suppose that d is the smallest value in K. For every $1 \le x \le t$, the following are true:*

1. *$\phi_{x,d}$ is $\mathbf{t}$-recoverable if $\mathbf{t}$ contains nonzero values only from K.*

2. *For any $a \in K$, bag B_a of $G_{x,w_{x,d}}$ is fully nonzero and contains values only from K.*

3. *For any $a \in K$ and any $y \ne w_{x,d}$, bag B_a of $G_{x,y}$ does not contain values from K.*

Assuming Claim 5.14, we consider the following three cases for the counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$.

Case 1: $(R, \mathbf{t}_1, \mathbf{t}_2)$ *is a union or difference counterexample such that $\mathbf{t}_1 + \mathbf{t}_2$ is contained in the component of a regular value a_1.*

Let us observe first that the minimality of Γ implies that $D \setminus \{0\}$ is equal to the component C_1 generated by a_1. Indeed, by Lemma 3.18(1-2), a_1 is regular in $\Gamma_{|C_1 \cup \{0\}}$ and a_1 generates C_1 in $\Gamma_{|C_1 \cup \{0\}}$ (thus $\Gamma_{|C_1 \cup \{0\}}$ is also a core). Set $y_x = w_{x,a_1}$ and let S contain v_{x,y_x} for every $1 \le x \le t$. We may assume that a_1 is the smallest nonzero value. Therefore, Claim 5.14(1) implies that G_{x,y_x} is both $\mathbf{t}_1$- and $\mathbf{t}_2$-recoverable, while Claim 5.14(3) implies that $G_{x,y}$ gives a homomorphism h with $h(\mathbf{t}_2) = 0$ if $y \ne y_x$ (as we have $K = D \setminus \{0\}$, Claim 5.14(3) implies that $G_{x,y}$ is fully zero). Thus we can use Lemma 3.21(3) or Lemma 3.22(3) to show that S is a solution for the instance we are reducing from.

Case 2: $(R, \mathbf{t}_1, \mathbf{t}_2)$ *is a union counterexample such that for $i = 1, 2$, tuple $\mathbf{t}_i$ is contained in the component C_i of some regular value a_i.*

Set $y^1_x = w_{x,a_1}$ and $y^2_x = w_{x,a_2}$. We may assume that a_i is the smallest value in component C_i (if both a_1 and a_2 appear in the same component C_i, then we can set $a_1 = a_2$ and we are in Case 1). Claim 5.14(1) implies that G_{x,y^1_x} is $\mathbf{t}_1$-recoverable and G_{x,y^2_x} is $\mathbf{t}_2$-recoverable. Thus by Lemma 3.21(3), a NAND$(G_{x,y}, G_{x',y'})$ constraint excludes the possibility that $y^1_x = y$ and $y^2_x = y'$ for some $y \ne y'$. In particular, the constraints of the form NAND$(G_{x,y}, G_{x,y'})$, $y \ne y'$ ensure that $y^1_x = y^2_x$ for every $1 \le x \le t$; let S be the set of vertices that contains $v_{x,y}$ if and only if $y = y^1_x = y^2_x$. Note that G_{x,y^1_x} is both $\mathbf{t}_1$- and $\mathbf{t}_2$-recoverable. Now it is easy to see that S is a multicolored

35

independent set: if $v_{x,y}, v_{x',y'} \in S$ are adjacent vertices, then the constraint $\text{NAND}(G_{x,y}, G_{x',y'})$ would be violated (Lemma 3.21(3)).

Case 3: $(R, \mathbf{t}_1, \mathbf{t}_2)$ is a union counterexample such that for $i = 1, 2$, tuple $\mathbf{t}_i$ is contained in the component C_i of value a_i, where a_1 is regular and a_2 is self-producing.

As in Case 1, the minimality of Γ implies that $D = C_1 \cup C_2 \cup \{0\}$; note that $C_2 = \{a_2\}$ as a_2 is self-producing. Furthermore, there are no self-producing values $c \in C_1$: since $\{c\}$ is a component and $C_1 \setminus \{c\}$ is not a component (as C_1 is the smallest component containing $a_1 \neq c$), by Proposition 3.17, this would imply that there is a difference counterexample in C_1, contradicting the minimality of D. Thus a_2 is the only self-producing value in D. We may assume that a_1 is the smallest value of C_1.

Set $y_x^1 = y_x^2 = w_{x,a_1}$. As in the previous case, G_{x,y_x^1} is $\mathbf{t}_1$-recoverable. Let us show that a_2 can only appear in a bag representing a_2. Indeed, suppose that a_2 appears in bag B_c of $G_{x,y}$ for some $c \in C_1$. It is clear that a_2 cannot appear in a bag representing a degenerate value (Proposition 3.9) and there are no self-producing values in C_1, thus c is regular. If $y = w_{x,a_1}$, then Claim 5.13(2) states that bag B_c contains only values from C_1. Thus $y \neq w_{x,a_1}$, and hence by Claim 5.14(3), the bag does not contain any values from C_1. As c is regular, the size of the bag B_c is larger than the cardinality constraint $\pi(a_2)$, implying that 0 also appears in bag B_c. Thus gadget $G_{x,y}$ gives a multivalued morphism ϕ with $0, a_2 \in \phi(c)$. If ϕ_{a_2} is the multivalued morphism witnessing that a_2 is self-producing, then we have $0, a_2 \in (\phi \circ \phi_{a_2})(c)$ and $0 \in (\phi \circ \phi_{a_2})(c')$ for every value c'. This shows that c produces a_2. As c is regular, value a_2 does not produce c, contradicting the assumption that a_2 is self-producing. Thus value a_2 appears only in bags representing a_2, and there are at least t gadgets where a_2 appears. Note that such a gadget is clearly $\mathbf{t}_2$-recoverable. Furthermore, it is not possible that a_2 appears in bag B_{a_2} of $G_{x,y}$ for $y \neq y_x^1$: as G_{y,y_x^1} is $\mathbf{t}_1$-recoverable and $G_{x,y}$ is $\mathbf{t}_2$-recoverable, the $\text{NAND}(G_{x,y_x^1}, G_{x,y})$ constraint would not be satisfied. Thus value a_2 has to appear in the gadgets G_{x,y_x^1}, $1 \leq x \leq t$, implying that these gadgets are both $\mathbf{t}_1$- and $\mathbf{t}_2$-recoverable. From this point, we can finish the proof as in the previous case. $\square$

5.2.4 Proof of Claim 5.14

First, we show that Claim 5.14 follows from the following claim:

Claim 5.15. *Let K be the component generated by a regular value d and suppose that d is the smallest value in K.*

1. *For every $1 \leq x \leq t$ and $a \in K$, bag B_a of $G_{x,w_{x,d}}$ contains only values from K, i.e., $\phi_{x,d}(a) \subseteq K$.*

2. *For every $1 \leq x \leq t$, there is a multivalued morphism β_x such that $\hat{\phi}_{x,d} = \phi_{x,d} \circ \beta_x$ is a multivalued morphism with $\phi_{x,d}(a) \cup \{a\} \subseteq \hat{\phi}_{x,d}(a)$ for every $a \in K$ and $\hat{\phi}_{x,d}(a) = \{0\}$ for every $a \notin K$.*

Note that $\phi_{x,d}(a) \subseteq K$ implies, in particular, $0 \notin \phi_{x,d}(a)$.

Proof (of Claim 5.14 from Section 5.2.3). To see this, we first show that if $\mathbf{t}$ contains nonzero values only from K, then $a \in \hat{\phi}_{x,d}(a)$ implies that a subset of $\phi_{x,d}$ is a $\mathbf{t}$-recoverable endomorphism. Indeed, it means that for every $a \in K$, there is a $c_a \in \phi_{x,d}(a)$ such that $a \in \beta_x(c_a)$. Then any endomorphism that maps a to c_a is $\mathbf{t}$-recoverable, as witnessed by β_x. This proves Statement 1 of Claim 5.14. Statement 2 of Claim 5.14 is the same as Statement 1 of Claim 5.15. Statement 3 of Claim 5.14 follows from Statement 2 of Claim 5.14 and from the fact that the cardinality

requirement for the values in K is exactly the same as the total size of the bags B_a of the gadgets $G_{x,w_{x,d}}$ for every $a \in K$ and $1 \le x \le t$. $\qquad\square$

In the rest of the section, we prove Claim 5.15 by induction on x.

Proof (of Claim 5.15). We show that β_x and $\hat{\phi}_{x,d}$ exist if $\hat{\phi}_{i,d}$ exists for every $1 \le i < x$ (in case of $x = 1$, this condition is vacuously true). Let Ψ_i be the set of those multivalued morphisms that can be obtained as the product of an arbitrary sequence of at least one multivalued morphisms composed from $\mathrm{ret}_K, \hat{\phi}_{1,d}, \ldots, \hat{\phi}_{i,d}$ (possibly using some of them multiple times). Observe that Ψ_i has a unique maximal element ψ_i, i.e, $\psi_i(a) \supseteq \psi'_i(a)$ for every $a \in K$ and $\psi'_i \in \Psi_i$. This follows from the fact that if $\psi'_i, \psi''_i \in \Psi_i$, then $a \in \psi'_i(a)$ and $a \in \psi''_i(a)$ for every $a \in K$ implies $\psi'_i(a) \cup \psi''_i(a) \subseteq (\psi'_i \circ \psi''_i)(a)$. For notational convenience, we define $\psi_0 := \mathrm{ret}_K$. Note that $\psi_i(a) = \{0\}$ for every $a \notin K$ by definition.

First, we prove Claim 5.15 assuming that the following claim is true:

Claim 5.16. *For every $b \in K$, there is an integer $p_b \ge 1$ such that for every $p \ge p_b$, either 0 or b is in $(\phi_{x,d} \circ \psi_{x-1})^p(b)$. Furthermore, for $b = d$ we actually have $d \in (\phi_{x,d} \circ \psi_{x-1})^p(d)$.*

Let $p := \max_{b \in K} p_b$. For every $b \in K$ and $p' \ge p$, either 0 or b is in $(\phi_{x,d} \circ \psi_{x-1})^{p'}(b)$ and in particular $d \in (\phi_{x,d} \circ \psi_{x-1})^{p'}(d)$ holds. For some $p' \ge p$, let S contain those elements b of K for which $b \in (\phi_{x,d} \circ \psi_{x-1})^{p'}(b)$ holds; we have $d \in S \subseteq K$. Observe that S is a component containing d (as ret_S is a subset of $(\psi_{x-1} \circ \phi_{x,d})^{p'}$), thus S has to contain the component generated by d, i.e., $S = K$. Therefore, we can assume that $b \in (\phi_{x,d} \circ \psi_{x-1})^{p'}(b)$ for every $b \in K$ and $p' \ge p$. Let us define $\beta_x := \psi_{x-1} \circ (\phi_{x,d} \circ \psi_{x-1})^p$ and $\hat{\phi}_{x,d} = \phi_{x,d} \circ \beta_x = (\phi_{x,d} \circ \psi_{x-1})^{p+1}$; it is clear that $b \in \hat{\phi}_{x,d}(b)$ for every $b \in K$. Furthermore, $b \in (\phi_{x,d} \circ \psi_{x-1})^p(b)$ implies $\phi_{x,d}(b) \subseteq \hat{\phi}_{x,d}(b)$, proving the second statement of Claim 5.15.

To prove the first statement, observe that we know from Claim 5.13(2) that $\phi_{x,d}(d) \subseteq K$. For $b \neq d$, if $\phi_{x,d}(b)$ contains a value not in K, then $0 \in (\phi_{x,d} \circ \psi_{x-1})^p(b)$ would follow and then $\mathrm{ret}_{K \setminus \{b\}}$ is a subset of $(\phi_{x,d} \circ \psi_{x-1})^p$, again contradicting that K is the component generated by d. This proves the first statement of Claim 5.15. $\qquad\square$

Now to finish the proof of Claim 5.15, it suffices to prove Claim 5.16. We prove Claim 5.16 by double induction: for a fixed $1 \le x \le t$ and $b \in K$, we assume that Claim 5.15 is true for every $i < x$, and that Claim 5.16 is true for x and for every $a < b$. It is clear that such a proof and the proof of Claim 5.15 above together prove both Claim 5.15 and 5.16.

Proof of Claim 5.16. We prove the statement by induction on b. Suppose that the statement holds for every $a < b$; let us prove it for b.

Let $K_b = \{a \mid a \in K, a \le b\}$ and let us define

$$T = \bigcup_{p \ge 1} (\mathrm{ret}_{K_b} \circ \phi_{x,d} \circ \psi_{x-1} \circ \mathrm{ret}_K)^p(b) \setminus \{0\},$$

which is a subset of K. (Note that we have no reason to assume that ret_{K_b} is an endomorphism.) Intuitively, T is the set of values that can be obtained from b by a sequence of endomorphisms using $\hat{\phi}_{1,d}, \ldots, \hat{\phi}_{x-1,d}$ and the endomorphisms given by gadget $G_{x,w_{x,d}}$. The two retractions in the definition of T introduce two additional technical conditions: we never leave K (i.e., we consider every value outside K to be 0) and we apply $\phi_{x,d}$ only on values at most b (i.e., we imagine the bags B_a of $G_{x,w_{x,d}}$ with $a > b$ to be fully zero).

We show that if $b \in T$, then Claim 5.16 follows. Indeed, in that case if $b \in (\phi_{x,d} \circ \psi_{x-1})(b)$, then we can define $p_b := 1$ since $b \in (\phi_{x,d} \circ \psi_{x-1})^p(b)$ for every $p \ge 1$. Thus we are done in this

37

case. Suppose that $b \notin (\phi_{x,d} \circ \psi_{x-1})(b)$. As $b \in T$, this means that there is an $a \in T$, $a < b$ such that $a \in (\phi_{x,d} \circ \psi_{x-1})(b)$ and $b \in (\text{ret}_{K_b} \circ \phi_{x,d} \circ \psi_{x-1} \circ \text{ret}_K)^p(a)$ for some $p \geq 1$. By the induction hypothesis on $a < b$, there is a $p_a \geq 1$ such that either a or 0 is in $(\phi_{x,d} \circ \psi_{x-1})^{p_a+s}(a)$ for every $s \geq 0$. Putting together, we get that either 0 or b is in $(\phi_{x,d} \circ \psi_{x-1})^{1+p_a+s+p}(b)$ for every $s \geq 0$. Thus we can define $p_b := 1 + p_a + p$. To prove the second statement, observe that if $b = d$, then (as d is the smallest element of K), there is no $a < b$ with $a \in T$. Thus we can only get $b \in (\phi_{x,d} \circ \psi_{x-1})(b)$ in this case.

In the rest of the proof our goal is to argue that we can assume $b \in T$. We show next that Claim 5.16 is true if a value not in T (including 0, which is not in T by definition) appears in any of the sets of bags (B1)–(B3) defined below, and then argue that if all values in these bags are from T, then $b \in T$ as well.

(B1) bag B_a $(a \in T)$ of gadget $G_{i,w_{i,d}}$, $1 \leq i < x$,

(B2) bag B_b of $G_{x,w_{x,d}}$

(B3) bag B_a $(a \in T, a < b)$ of gadget $G_{x,w_{x,d}}$

For (B1), we know by induction that Statement 1 of Claim 5.15 holds for every $i < x$, showing that bag B_a $(a \in T)$ of gadget $G_{i,w_{i,d}}$ contains values only from K. This is the point where the definition of ψ_{x-1} becomes crucial. Intuitively, we can consider a sequence of multivalued morphisms showing that $a \in T$, and then append an application of $\phi_{i,d}$ to show that $\phi_{i,d}(a)$ is in T as well; the important point here is that $\psi_{x-1} \circ \phi_{i,x}$ can be replaced by ψ_{x-1}. Formally, if $a \in T$, then there is a $p' \geq 0$ and an $a' \in K$ such that $a' \in \Big((\text{ret}_{K_b} \circ \phi_{x,d} \circ \psi_{x-1} \circ \text{ret}_K)^{p'} \circ (\text{ret}_{K_b} \circ \phi_{x,d}) \Big)(b)$ and $a \in \psi_{x-1}(a')$. The definition of ψ_{x-1} implies that $(\psi_{x-1} \circ \phi_{i,d})(a') \subseteq \psi_{x-1}(a')$, and therefore $\phi_{i,d}(a) \subseteq \psi_{x-1}(a')$ holds as well. This means that $\phi_{i,d}(a) \subseteq (\text{ret}_{K_b} \circ \phi_{i,d} \circ \psi_{x-1} \circ \text{ret}_K)^{p'+1}(b)$, that is, $\phi_{i,d}(a) \subseteq T$. Thus, bags from (B1) cannot have values of K not in T.

For (B2), observe that if we show that only values from K appear in the bag B_b of $G_{x,w_{x,d}}$, then it follows that all these values are in T. For $b = d$, Claim 5.13(2) implies that all these values are in K. If $b \neq d$, then a value not in K appearing in bag B_b of $G_{x,w_{x,d}}$ would imply that $0 \in (\phi_{x,d} \circ \psi_{x-1})(b)$ (recall that $\psi_{x-1}(a) = \{0\}$ for any $a \notin K$), and the statement of Claim 5.16 is true for b with $p_b = 1$.

For (B3), it is again sufficient to show that the bags from this set contain only values from K; by the definition of T, then these values are in T. If $a = d$, then Claim 5.13(2) implies that the bag contains values only from K. Otherwise, suppose that $a \neq d$, $a \in T$, $a < b$, and there is a value $c \notin K$ in the bag B_a of $G_{x,w_{x,d}}$. As $a \in T$, we have that $a \in (\text{ret}_{K_b} \circ \phi_{x,d} \circ \psi_{x-1} \circ \text{ret}_K)^p(b)$ for some $p \geq 1$, which clearly implies $a \in (\phi_{x,d} \circ \psi_{x-1})^p(b)$. Since $0 \in (\phi_{x,d} \circ \psi_{x-1})(a)$ (recall that $\psi_{x-1}(c) = \{0\}$ if $c \notin K$), it follows that $0 \in (\phi_{x,d} \circ \psi_{x-1})^{p+1}(b)$, proving Claim 5.16 for b with $p_b = p + 1$. Note that in these cases we assumed $a < b$, i.e., $b \neq d$ (as d is the smallest value in K) and therefore we do not have to prove the second statement of Claim 5.16.

Therefore, in the following, we can assume that bags (B1)–(B3) contain values only from T. The total size of these bags is

$$\sum_{a \in T} \sum_{i=1}^{x-1} z_{i,a} + \sum_{a \in T, a<b} z_{x,a} + z_{x,b}, \tag{3}$$

while the sum of cardinality constraints of the values in T is (assuming $b \notin T$)

$$\sum_{a \in T} \sum_{i=1}^{t} z_{i,a} = \sum_{a \in T} \sum_{i=1}^{x-1} z_{i,a} + \sum_{a \in T, a<b} z_{x,a} + \sum_{a \in T, a>b} z_{x,a} + \sum_{a \in T} \sum_{i=x+1}^{t} z_{x,a}. \tag{4}$$

38

Let us compare the last term of (3) with the last two terms of (4). Suppose first that b is regular. Then $z_{x,b} \geq (2t\Delta)z_{x,a}$ for any $a > b$ and $z_{x,b} \geq (2t\Delta)z_{i,a}$ for any $a \in T$ and $i \geq x + 1$. Thus (3) is strictly larger than (4) and this contradiction proves that $b \in T$. Suppose now that b is not regular. Then T cannot contain any regular value (Proposition 3.9). Again, we have $z_{x,b} \geq (2t\Delta)z_{x,a}$ for any $a > b$ and $z_{x,b} \geq (2t\Delta)z_{i,a}$ for any $a \in T$ and $i \geq x + 1$, leading to a contradiction. Therefore, we can conclude that $b \in T$, concluding the proof of Claim 5.16. $\square$

6 Examples

Example 6.1. A number of graph problems can be represented in the form of CCSP(Γ) or OCSP(Γ).

INDEPENDENT SET: Given a graph G with vertices v_i ($1 \leq j \leq n$), find an independent set of size t. INDEPENDENT SET is equivalent to CCSP($\{R_{IS}\}$), or, equivalently to OCSP($\{R_{IS}\}$), where R_{IS} is a binary relation on $\{0,1\}$ given by

$$R_{IS} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

(tuples are written vertically). The size constraint is set to be t.

p-COLORABLE SUBGRAPH: Given a graph G and an integer k, find a set S of k vertices that induces a p-colorable subgraph. This problem is equivalent to OCSP($\{R_{p-COL}\}$), where R_{p-COL} is a binary relation on $p + 1$-element set $D = \{0,1,\dots p\}$ given by

$$R_{p-COL} = D^2 \setminus \left\{ \begin{pmatrix} i \\ i \end{pmatrix} \mid i \in \{1,\dots,p\} \right\}.$$

The size constraint is k, the size of the p-colorable graph to be found.

IMPLICATIONS: Given a directed graph G and an integer t, find a set C of vertices with exactly t vertices such that there is no directed edge $\overrightarrow{uv}$ with $u \in C$ and $v \notin C$. IMPLICATIONS can be represented as CCSP($\{R_{IM}\}$) and OCSP($\{R_{IM}\}$), where

$$R_{IM} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

The size constraint is set to be t.

VERTEX COVER: Given a graph G and an integer t, find a set C of vertices such that every edge of G is incident to at least one vertex from C. VERTEX COVER is equivalent to the OCSP(R_{VC}), where

$$R_{VC} = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix},$$

and the size constraint is t.

Some problems reduce to the CSP with cardinality constraints in a less straightforward way.

Example 6.2. BICLIQUE: Given a bipartite graph $G(A, B)$, find two sets $A' \subseteq A$ and $B' \subseteq B$, each of size exactly t, such that every vertex of A' is adjacent with every vertex of B'. As it is mentioned in the introduction, BICLIQUE is equivalent to CCSP($\{R_{BC}\}$), where R_{BC} is a relation on $\{0,1,2\}$ given by

$$R_{BC} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}.$$

A BICLIQUE instance $G(A, B)$ is reduced to CCSP($\{R_{BC}\}$) by first taking the (bipartite) complement $\overline{G}$ of G and imposing constraint $\langle (v, w), R_{BC} \rangle$ on every pair $v \in A$, $w \in B$ such that v, w are adjacent in $\overline{G}$. The cardinality constraint $\pi : \{0, 1, 2\} \to \mathbb{N}$ is chosen to be $\pi(1) = \pi(2) = t$.

The variant of BICLIQUE, where the graph G is not necessarily bipartite, is equivalent to CCSP($\{R'_{BC}\}$), where R'_{BC} is a relation on $\{0, 1, 2\}$ given by

$$R'_{BC} = \begin{pmatrix} 0 & 1 & 0 & 2 & 0 \\ 0 & 0 & 2 & 0 & 1 \end{pmatrix}.$$

The following examples generalize BICLIQUE to finding complete p-partite graphs. Let us first consider the version where the input graph is also p-partite:

Example 6.3. p-PARTITE CLIQUE: Given a p-partite graph G with partition $A_1, \ldots A_p$, find sets $A'_1 \subseteq A_1, \ldots A'_p \subseteq A_p$, each of size exactly t, such that for any i, j, $1 \le i < j \le k$ every vertex from A'_i is adjacent with every vertex of A'_j. The equivalent CCSP problem is CCSP($\{R_{p-MC}\}$) where R_{p-MC} is the p-ary relation given by

$$R_{p-MC} = \{0, 1\} \times \{0, 2\} \times \ldots \times \{0, p\} - \{(1, 2, \ldots, p)\}.$$

Reduction goes as follows. Given a p-partite graph G with partition $A_1, \ldots A_p$ we first take the (p-partite) complement $\overline{G}$ of G, and then introduce constraint $\langle (v_1, \ldots, v_p), R_{p-MC} \rangle$ for each p-tuple $(v_1, \ldots, v_p)$ such that $v_i \in A_i$ and some of the vertices $v_1, \ldots, v_p$ are adjacent in $\overline{G}$. The cardinality constraint is chosen to be $\pi(1) = \ldots = \pi(p) = t$.

Another way to represent p-PARTITE CLIQUE by a CCSP is the following. Let

$$R_{p-PC} = \begin{pmatrix} 0 & 1 & 0 & \cdots & p & 0 \\ 0 & 0 & 1 & \cdots & 0 & p \end{pmatrix},$$

and R_i is the unary relation $\{0, i\}$ for $i \in \{1, \ldots, p\}$. Then p-PARTITE CLIQUE reduces to CCSP($\{R_{p-PC}, R_1, \ldots, R_p\}$) by imposing the constraint $\langle (v), R_i \rangle$ on each $v \in A_i$, and $\langle (v, w), R_{p-PC} \rangle$ on each pair v, w adjacent in $\overline{G}$.

The following example formulates the version of p-PARTITE CLIQUE where the input graph is not p-partite:

Example 6.4. p-PARTITE COMPLETE SUBGRAPH: Given a graph G and integers $t_1, \ldots, t_p$, find sets $S_1, \ldots S_p$ of vertices such that $S_1 \cup \ldots \cup S_p$ induces a complete p-partite graph with partition $S_1, \ldots, S_p$, and $|S_i| = t_i$ for $1 \le i \le p$. This problem is equivalent to CCSP($\{R_{p-CS}\}$), where R_{p-CS} is a binary relation on $p + 1$-element set $D = \{0, 1, \ldots p\}$ given by

$$R_{p-CS} = \{0, 1\}^2 \cup \{0, 2\}^2 \cup \ldots \cup \{0, p\}^2.$$

To reduce the p-PARTITE COMPLETE SUBGRAPH problem to CCSP($\{R_{p-CS}\}$) we, first, take the complement $\overline{G}$ of G, and then impose constraint $\langle (v, w), R_{p-CS} \rangle$ on every pair v, w such that v, w are adjacent in $\overline{G}$. The cardinality constraint is set to be π such that $\pi(i) = t_i$ for all i.

Example 6.5. Let $R = (\{0, 1\} \times \{0, 1\} \times \{0, 2\}) \setminus \{(1, 1, 2)\}$. Let Γ be the cc-closure of R. Note that Γ contains $R^{|3;2} = \{(0, 0), (1, 0), (0, 1)\}$, which is the relation R_{IS} of Example 6.1, showing that OCSP(Γ) is W[1]-hard. On the other hand, we show that OCSP($\{R\}$) is fixed-parameter tractable.

Let S_1 be the set of variables v where value 1 can appear, that is, there is no constraint $\langle (v', v'', v), R \rangle$ on v. Observe that any combination of 0 and 1 on S_1 is a satisfying assignment.

Therefore, if $|S_1| \geq k$, then there is a solution. Let S_2 be the set of variables v where value 2 can appear, that is, there is no constraint $\langle (v, v', v''), R \rangle$ or $\langle (v', v, v''), R \rangle$ on v. Observe that any combination of 0 and 2 on S_2 is a satisfying assignment. Therefore, if $|S_2| \geq k$, then there is a solution. On the variables not in S_1 and S_2, only value 0 can appear. Therefore, if $|S_1|, |S_2| < k$, then the number of possible assignments that we need to try is $2^{|S_1|+|S_2|} < 2^{2k}$.

Example 6.6. Let $R = (\{0, 1, 2\} \times \{0, 1, 2\} \times \{0, 2\}) \setminus \{(1, 1, 2)\}$. Let Γ be the cc-closure of R. Note that Γ contains $R^{|3;2} = (\{0, 1, 2\} \times \{0, 1, 2\}) \setminus \{(1, 1)\}$. Therefore, $\Gamma_{|\{0,1\}}$ contains the relation $\{(0, 0), (1, 0), (0, 1)\}$, which is the relation R_{IS} of Example 6.1, showing that $CCSP(\Gamma_{|\{0,1\}})$ is W[1]-hard. We can reduce $CCSP(\Gamma_{|\{0,1\}})$ to $CCSP(\Gamma)$ by setting the cardinality constraint of value 2 to 0, thus the W[1]-hardness of $CCSP(\Gamma)$ follows. On the other hand, we show that $CCSP(\{R\})$ is fixed-parameter tractable.

Let S be the set of variables v where value 1 can appear, that is, there is no constraint $\langle (v', v'', v), R \rangle$ on v. Observe that any combination of 0, 1, and 2 on S is a satisfying assignment. Therefore, if $|S| \geq k$, then there is a solution. If $|S| < k$, then we can try every possible substitution of constants into S and obtain instances where 1 can no longer appear. As the problem on $\Gamma_{|\{0,2\}}$ is trivial, fixed-parameter tractability follows.

Example 6.7. The language consisting of the relations

$$R_1 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 2 & 2 \end{pmatrix} \quad R_2 = \begin{pmatrix} 0 & 0 \\ 0 & 2 \end{pmatrix} \quad R_3 = \begin{pmatrix} 0 & 2 \end{pmatrix} \quad R_4 = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \quad R_5 = \begin{pmatrix} 0 & 1 \end{pmatrix} \quad R_6 = \begin{pmatrix} 0 \end{pmatrix}$$

is a cc0-language, but not a cc-language: substituting 1 into the first coordinate of R_1 results in the (non 0-valid) relation $\{(1, 0), (1, 2)\}$, which is not in the language.

Example 6.8.
- Relation R_{IS} is not weakly separable: $(R_{IS}, (1, 0), (0, 1))$ is a union counterexample (i.e., $(1, 1) \notin R_{IS}$).

- Relation R_{IM} is not weakly separable: $(R_{IM}, (1, 0), (0, 1))$ is a difference counterexample.

- The r-ary relation

$$R_{\text{even}} = \{(a_1, \ldots, a_r) \in \{0, 1\}^r \mid a_1 + \cdots + a_r \text{ is even}\}$$

  is weakly separable. Indeed, if there is an even number of 1's in each of $\mathbf{t}_1$, $\mathbf{t}_2$ and they are disjoint, then their union also contains an even number of 1's. Similarly, if $\mathbf{t}_1 + \mathbf{t}_2$ and $\mathbf{t}_2$ contain even number of 1's, then so does $\mathbf{t}_1$.

- We can generalize R_{even} the following way: let us define the r-ary $R_{\text{mod-}p}$ relation over the domain $\{0, \ldots, d\}$ the following way:

$$R_{\text{mod-}p} = \{(a_1, \ldots, a_r) \in \{0, \ldots, d\}^r \mid a_1 + \cdots + a_r = 0 \mod p\}.$$

  It is easy to see that this relation is weakly separable as well.

Example 6.9. We demonstrate how hardness is proved in the Boolean case [24] if there is a relation that is not weakly separable. Let Γ be a cc0-language over $\{0, 1\}$ that is not weakly separable. Suppose that there is a union counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$ in Γ; suppose for example that

$$\mathbf{t}_1 = (\overbrace{1, \ldots, 1}^{p}, 0, \ldots, 0), \quad \mathbf{t}_2 = (0, \ldots, 0, \overbrace{1, \ldots, 1}^{q}, 0, \ldots, 0).$$

41

Since Γ is a cc0-language, by substituting 0's in the last coordinates, we can obtain a relation $R' \in \Gamma$ and a union counterexample $(R', \mathbf{t}_1', \mathbf{t}_2')$ with

$$\mathbf{t}_1' = (\overbrace{1, \ldots, 1}^{p}, 0, \ldots, 0), \ \mathbf{t}_2' = (0, \ldots, 0, \overbrace{1, \ldots, 1}^{q}).$$

Now it is easy to see that the W[1]-hard problem $\mathrm{OCSP}(R_{IS})$ is reducible to $\mathrm{OCSP}(\Gamma)$: a constraint $\langle (x, y), R_{IS} \rangle$ can be simulated by a constraint $\langle (x, \ldots, x, y, \ldots, y), R' \rangle$. The correctness of this reduction follows from

$$(0, \ldots, 0, 0, \ldots, 0) \in R'$$

$$(\overbrace{1, \ldots, 1}^{p}, 0, \ldots, 0) \in R'$$

$$(0, \ldots, 0, \overbrace{1, \ldots, 1}^{q}) \in R'$$

$$(\overbrace{1, \ldots, 1, 1, \ldots, 1}^{p+q}) \notin R'.$$

Suppose now that there is a difference counterexample $(R, \mathbf{t}_1, \mathbf{t}_2)$. As above, let us suppose that there is also a difference counterexample $(R', \mathbf{t}_1', \mathbf{t}_2')$ with $\mathbf{t}_1' = (1, \ldots, 1, 0, \ldots, 0)$ and $\mathbf{t}_2' = (0, \ldots, 0, 1, \ldots, 1)$. Now we can reduce the W[1]-hard problem $\mathrm{OCSP}(R_{IM})$ to $\mathrm{OCSP}(\Gamma)$. Since we have $(0, \ldots, 0), (1, \ldots, 1), (0, \ldots, 0, 1, \ldots, 1) \in R'$ and $(1, \ldots, 1, 0, \ldots, 0) \notin R'$, now a constraint $\langle (x, \ldots, x, y, \ldots, y), R' \rangle$ expresses the relation R_{IM}.

Example 6.10. Consider the cc0-closure Γ of the following relation:

$$R = \begin{pmatrix} 0 & 1 & 2 & 0 & 2 \\ 0 & 0 & 0 & 2 & 2 \end{pmatrix},$$

Clearly, Γ is not weakly separable: $(R, (1, 0), (0, 2))$ is a union counterexample. Nevertheless, both $\mathrm{OCSP}(\Gamma)$ and $\mathrm{CCSP}(\Gamma)$ are fixed-parameter tractable. In the case of $\mathrm{OCSP}(\Gamma)$, every 1 in a solution can be replaced by 2. Hence it is sufficient to solve the problem restricted to $\{0, 2\}$, in which case the problem is trivial, as every combination of 0 and 2 is a satisfying assignment. For $\mathrm{CCSP}(\Gamma)$, we argue as follows. Let S be the set of variables v where value 1 can appear, that is, there is no $\langle (v', v), R \rangle$ or any unary constraint excluding 1 on v. Observe that any combination of 0, 1, and 2 on S is a satisfying assignment. Therefore, if $|S| \geq k$, then there is a solution. If $|S| < k$, then we can try every possible substitution of constants into S and obtain instances where 1 can no longer appear. As $\mathrm{CCSP}(\Gamma_{|\{0,2\}})$ is trivial, solving these instances is fixed-parameter tractable.

Example 6.11. The relation R_{IM} of Example 6.1 is not weakly separable. Consider a CSP instance on three variables v_1, v_2, v_3 having two constraints $\langle (v_1, v_3), R_{IM} \rangle$ and $\langle (v_2, v_3), R_{IM} \rangle$. The assignment $(0, 0, 1)$ is the only minimal satisfying assignment. Assignment $(1, 1, 1)$ is satisfying, but it cannot be obtained as the union of pairwise disjoint satisfying assignments (even if we do not require *minimal* satisfying assignments).

Example 6.12. • The cc0-closure Γ_{p-PC} of $\{R_{p-PC}\}$ (defined in Example 6.3) consists of R_{p-PC} itself and unary relations $\{(0)\}$ and $D = \{0, 1, \ldots, p\}$. It is easy to see that every mapping $h : D \to D$ with $h(0) = 0$ is an endomorphism of Γ_{p-PC}, for any sets $0 \in D_1, D_2 \subseteq D$, any mapping $f : D_1 \to D_2$ with $f(0) = 0$ is an inner homomorphism of Γ_{p-PC}. Finally, any mapping $\phi : D_1 \to 2^{D_2}$ such that $0 \in D_1, D_2 \subseteq D$ and $\phi(0) = \{0\}$ is a multivalued morphism or an inner multivalued morphism of Γ_{p-PC}.

- Multivalued morphisms of relation R_{p-MC} and relations from its cc0-closure are the mappings $\phi : D \to 2^D$ satisfying $\phi(d) \subseteq \{0, d\}$ for every $d \in D$. A mapping $g : D \to D$ is an endomorphism of R_{p-MC} if and only if $g(d) \in \{0, d\}$ for each $d \in D - \{0\}$ and $g(0) = 0$. Inner homomorphisms and inner multivalued morphisms can be described in a similar way.

- The constraint $R_{\leq,d}$ generalizes R_{IM} to the domain $\{0, 1, \ldots, d\}$: $R_{\leq,d} = \{(x, y) \in \{0, 1, \ldots, d\}^2 \mid x \leq y\}$. A function h with $h(0) = 0$ is an endomorphism of $R_{\leq,d}$ if and only if it is monotone, i.e., $h(x) \leq h(y)$ for every $x \leq y$. $R_{\leq,d}$ does not have a multivalued morphism that is not an endomorphism: if ψ is a multivalued morphism of $R_{\leq,d}$ with $a, b \in \psi(x)$, then $(x, x) \in R_{\leq,d}$ implies that both (a, b) and (b, a) are in $R_{\leq,d}$.

- The constraint $R_{<,d}$ is defined similarly to $R_{\leq,d}$, but we also add the tuple $(0, 0)$ to make it 0-valid: $R_{\leq,d} = (0, 0) \cup \{(x, y) \in \{0, 1, \ldots, d\}^2 \mid x < y\}$. In this case, there are nontrivial multivalued morphisms: for example, $\psi(0) = \psi(1) = \psi(2) = \{0\}$, $\psi(3) = \{1, 2\}$, $\psi(4) = \{3\}$ is a multivalued morphism of $R_{<,4}$.

Example 6.13. In the cc0-closure of the relation

$$R = \begin{pmatrix} 0 & 1 & 0 & 1 & 2 \\ 0 & 0 & 1 & 1 & 2 \end{pmatrix},$$

both 1 and 2 produce 1, but neither 1 nor 2 produces 2.

Example 6.14. Consider the cc0-closure of the following two relations:

$$R_1 = \begin{pmatrix} 0 & 1 & 1 & 2 & 0 & 2 & 3 & 0 & 3 & 4 & 4 \\ 0 & 0 & 1 & 0 & 2 & 2 & 0 & 3 & 3 & 5 & 4 \end{pmatrix} \quad R_2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 3 & 4 & 5 \\ 0 & 0 & 1 & 1 & 3 & 4 & 5 \end{pmatrix}.$$

Value 2 produces 2 and 3, but there are no other producing relations. Thus 3 is degenerate and 2 is self-producing. The mapping $\psi(0) = \psi(1) = \psi(2) = \psi(3) = \{0\}$, $\psi(4) = \{1\}$, $\psi(5) = \{0, 1\}$ is a multivalued morphism, thus 1 is semiregular. Values 4 and 5 are regular.

Example 6.15. Consider the cc0-closure Γ of following two relations:

$$R_1 = \begin{pmatrix} 0 & 3 & 2 \\ 0 & 1 & 2 \\ 0 & 2 & 1 \end{pmatrix} \quad R_2 = \begin{pmatrix} 0 & 1 & 0 & 1 & 2 & 0 & 2 & 0 & 3 & 2 & 1 & 2 & 1 \\ 0 & 0 & 1 & 1 & 0 & 2 & 2 & 3 & 3 & 1 & 2 & 3 & 3 \end{pmatrix}.$$

Γ is not weakly separable: $(R_2, (3, 0), (0, 3))$ is a difference counterexample. Let us observe that $h(0) = 0$, $h(1) = 2$, $h(2) = 1$, $h(3) = 2$ is an endomorphism (a proper contraction) of Γ. Therefore, by applying h on a solution for an OCSP(Γ) instance, we can obtain a solution using only the values 0, 1, and 2. Γ restricted to $\{0, 1, 2\}$ is weakly separable, thus finding such solutions is FPT.

Example 6.16. We have seen in Example 6.10 that CCSP(Γ) is fixed-parameter tractable for the cc0-closure Γ of the following relation:

$$R = \begin{pmatrix} 0 & 1 & 2 & 0 & 2 \\ 0 & 0 & 0 & 2 & 2 \end{pmatrix}.$$

Let us verify that Theorem 5.2 indeed classifies CCSP(Γ) as fixed-parameter tractable. Γ is not weakly separable: $(R, (1, 0), (0, 2))$ is a union counterexample. However, Γ is not a core: value 1 is self-producing, 2 is degenerate, and the component generated by 1 is $\{1\}$. $\Gamma_{|\{0,1\}}$ and $\Gamma_{|\{0,2\}}$ are cores, but they are weakly separable. Thus by Theorem 5.2, CCSP(Γ) is fixed-parameter tractable.

43

Consider the cc0-closure Γ of following relation:

$$R = \begin{pmatrix} 0 & 0 & 0 & 3 & 3 & 0 & 3 \\ 0 & 1 & 2 & 0 & 2 & 4 & 4 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Value 3 produces 3, 1 produces 4, and both 1 and 4 produces 2, but there are no other producing relations. Thus 2 and 4 are degenerate and 3 is self-producing. We can also see that 1 is regular. The core of Γ is the component generated by $\{1, 3\}$ which is $K = \{1, 2, 3\}$. Thus Γ is not a core. However, $\Gamma_{|\{0,1,2,3\}}$ is a core and it is not weakly separable: $(R, (3, 0, 0), (0, 1, 2))$ is a union counterexample. Thus by Theorem 5.2, CCSP(Γ) is W[1]-hard.

References

[1] L. Barto, M. Kozik, and T. Niven. Graphs, polymorphisms and the complexity of homomorphism problems. In *STOC*, pages 789–796, 2008.

[2] C. Bessière, E. Hebrard, B. Hnich, and T. Walsh. The complexity of global constraints. In *AAAI*, pages 112–117, 2004.

[3] S. Bourdais, P. Galinier, and G. Pesant. HIBISCUS: A constraint programming application to staff scheduling in health care. In *CP*, pages 153–167, 2003.

[4] A. Bulatov and D. Marx. Constraint satisfaction parameterized by solution size. In *38th International Colloquium on Automata, Languages and Programming*, volume 6755 of *Lecture Notes in Computer Science*, pages 424–436. Springer, 2011.

[5] A. A. Bulatov. A dichotomy theorem for constraint satisfaction problems on a 3-element set. *J. ACM*, 53(1):66–120, 2006.

[6] A. A. Bulatov. Complexity of conservative constraint satisfaction problems. *ACM Trans. Comput. Log.*, 12(4):24, 2011.

[7] A. A. Bulatov, P. Jeavons, and A. A. Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, 2005.

[8] A. A. Bulatov and D. Marx. The complexity of global cardinality constraints. *Logical Methods in Computer Science*, 6:1–27, 2010.

[9] A. A. Bulatov and D. Marx. Constraint satisfaction problems and global cardinality constraints. *Commun. ACM*, 53(9):99–106, 2010.

[10] N. Creignou, H. Schnoor, and I. Schnoor. Nonuniform Boolean constraint satisfaction problems with cardinality constraint. *ACM Trans. Comput. Log.*, 11(4), 2010.

[11] R. G. Downey and M. R. Fellows. *Parameterized Complexity*. Monographs in Computer Science. Springer, New York, 1999.

[12] T. Feder and M. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Computing*, 28:57–104, 1998.

[13] M. R. Fellows, D. Hermelin, F. A. Rosamond, and S. Vialette. On the parameterized complexity of multiple-interval graph problems. *Theor. Comput. Sci.*, 410(1):53–61, 2009.

[14] J. Flum and M. Grohe. *Parameterized Complexity Theory*. Springer, Berlin, 2006.

[15] S. Gaspers and S. Szeider. Kernels for global constraints. In *IJCAI*, pages 540–545, 2011.

[16] M. Grohe. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *J. ACM*, 54(1), 2007.

[17] P. Jeavons, D. Cohen, and M. Gyssens. Closure properties of constraints. *J. ACM*, 44:527–548, 1997.

[18] P. Jeavons, D. Cohen, and M. Gyssens. How to determine the expressive power of constraints. *Constraints*, 4:113–131, 1999.

[19] S. Khanna, M. Sudan, L. Trevisan, and D. P. Williamson. The approximability of constraint satisfaction problems. *SIAM J. Comput.*, 30(6):1863–1920, 2001.

[20] S. Kratsch, D. Marx, and M. Wahlström. Parameterized complexity and kernelizability of Max Ones and Exact Ones problems. In *MFCS*, pages 489–500, 2010.

[21] S. Kratsch and M. Wahlström. Preprocessing of Min Ones problems: A dichotomy. In *ICALP (1)*, pages 653–665, 2010.

[22] A. Krokhin and D. Marx. On the hardness of losing weight. *ACM Trans. Algorithms*, 8(2):19, 2012.

[23] G. Kun and M. Szegedy. A new line of attack on the dichotomy conjecture. In *STOC*, pages 725–734, 2009.

[24] D. Marx. Parameterized complexity of constraint satisfaction problems. *Computational Complexity*, 14(2):153–183, 2005. Special issue "Conference on Computational Complexity (CCC) 2004.".

[25] J.-C. Régin and C. P. Gomes. The cardinality matrix constraint. In *CP*, pages 572–587, 2004.

[26] I. Rosenberg. Multiple-valued hyperstructures. In *Proceedings of the 28th International Symposium on Multiple-Valued Logic (ISMVL '98)*, pages 326–333, 1998.

[27] M. Samer and S. Szeider. Tractable cases of the extended global cardinality constraint. *Constraints*, 16(1):1–24, 2011.

[28] T. J. Schaefer. The complexity of satisfiability problems. In *STOC*, pages 216–226. ACM, 1978.

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