Consistency for counting quantifiers

Florent R. Madelaine
LIMOS, Université d'Auvergne, Clermont-Ferrand, France

Barnaby Martin
Department of Computer Science, Durham University, U.K.

Abstract

We apply the algebraic approach for Constraint Satisfaction Problems (CSPs) with counting quantifiers, developed by Bulatov and Hedayaty, for the first time to obtain classifications for computational complexity. We develop the consistency approach for expanding polymorphisms to deduce that, if H has an expanding majority polymorphism, then the corresponding CSP with counting quantifiers is tractable. We elaborate some applications of our result, in particular deriving a complexity classification for partially reflexive graphs endowed with all unary relations. For each such structure, either the corresponding CSP with counting quantifiers is in P, or it is NP-hard.

2012 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Quantified Constraints, Constraint Satisfaction, Logic in Computer Science, Universal Algebra, Computational Complexity

Digital Object Identifier 10.4230/LIPIcs.MFCS.2018.11

1 Introduction

The *constraint satisfaction problem*, CSP($\mathcal{B}$), originating in artificial intelligence, is known to admit several equivalent formulations. Two of the best known consider the parameter $\mathcal{B}$ to be a relational structure and may be phrased as the problem of query evaluation of primitive positive (pp) sentences – those involving only $\{\exists, \wedge, =\}$ – on $\mathcal{B}$, and the homomorphism problem to $\mathcal{B}$ (see, e.g., [19]). For finite $\mathcal{B}$, CSP($\mathcal{B}$) is NP-complete in general, and a great deal of effort was expended in classifying its complexity in various different classes. It was conjectured by Feder and Vardi [13] that all such CSP($\mathcal{B}$) are either in P or NP-complete and this was finally proved last year independently by Bulatov [6] and Zhuk [23].

A popular generalisation of the CSP involves considering the query evaluation problem for the logic involving only $\{\forall, \exists, \wedge, =\}$. (This logic admits various names but we will leave it nameless in this work as was the case in the foundational [2].) The resulting *Quantified Constraint Satisfaction Problem*, QCSP($\mathcal{B}$), allows for a broader class, used in artificial intelligence to capture non-monotonic reasoning, whose complexities rise to Pspace-complete.

In this paper, we study counting quantifiers of the form $\exists^{\geq j}$, which allow one to assert the existence of at least j elements such that the ensuing property holds. Thus, on a structure $\mathcal{B}$ with domain of size n, the quantifiers $\exists^{\geq 1}$ and $\exists^{\geq n}$ are precisely $\exists$ and $\forall$, respectively. Counting quantifiers have been fiercely studied in finite model theory (see [12, 22]), where the focus is on supplementing the descriptive power of various logics. Of wider interest is the majority quantifier $\exists^{\geq n/2}$ (on a structure of domain size n), which sits broadly midway between $\exists$ and $\forall$. Majority quantifiers turn up across diverse fields of logic and have various practical applications, e.g. in cognitive appraisal and voting theory [11].

We postulate variants of CSP($\mathcal{B}$) in which the input sentence to be evaluated on $\mathcal{B}$ (of size $|B|$) remains positive conjunctive in its quantifier-free part, but is quantified by various counting quantifiers from some non-empty set.

43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018).
Editors: Igor Potapov, Paul Spirakis, and James Worrell; Article No. 11; pp. 11:1–11:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

For $X \subseteq \{1, \ldots, |B|\}$, $X \neq \emptyset$, the X-CSP($\mathcal{B}$), introduced in [21], takes as input a sentence given by a conjunction of atoms quantified by quantifiers of the form $\exists^{\geq j}$ for $j \in X$ (this logic is termed X-pp). It then asks whether this sentence is true on $\mathcal{B}$. In the present paper, we will mostly consider the situation in which all counting quantifiers are present, and we will denote this problem CQCSP($\mathcal{B}$), instead of $\{1, \ldots, |B|\}$-CSP($\mathcal{B}$). The corresponding logic, involving only $\{\exists^{\geq 1}, \ldots, \exists^{\geq |B|}, \wedge, =\}$, we will call cq-pp.

The algebraic method has been very potent in understanding the complexity of CSPs and QCSPs [5, 6, 23, 10]. Recently, an algebraic theory tailored to counting quantifiers has been given [8] (early version was [7]).

A *polymorphism* of a structure $\mathcal{B}$ is a homomorphism from $\mathcal{B}^k$ to $\mathcal{B}$, for some k. Let $\{1\} \subseteq X \subseteq \{1, \ldots, |B|\}$. Call a function $f : B^k \to B$ *expanding on* X if, for all $X_1, \ldots, X_k \subseteq B$ such that $|X_1| = \ldots = |X_k| = j \in X$, we have $|f(X_1, \ldots, X_k)| \geq j$. This condition at $j = 1$ is trivial (it says that f is a function) and at $j = |B|$ asserts surjectivity. If $X = \{1, \ldots, |B|\}$ we simply term f *expanding.*

▶ **Lemma 1** (Theorem 8 [7]; Corollary 14 [8]). *The relations that are cq-pp-definable over $\mathcal{B}$ are exactly those that are preserved by the expanding polymorphisms of $\mathcal{B}$.*

In this paper, we will only make use of the "easy" direction of Lemma 1, that is, any relation that is cq-pp-definable over $\mathcal{B}$ is preserved by the expanding polymorphisms of $\mathcal{B}$.

The *list homomorphism* problem, which we will call List-CSP($\mathcal{B}$), is defined as CSP($\mathcal{B}$), save that one gives lists for each input variable stating which elements of the domain B that variable may be evaluated on. This is equivalent to CSP($\mathcal{B}^*$), where $\mathcal{B}^*$ is $\mathcal{B}$ endowed with additional unary relations for each subset of B. Indeed, this class of CSPs was among the first to be proved in line with the Feder-Vardi dichotomy conjecture [4]. The key class of polymorphisms here is known as *conservative* and the property they have is that $f(x_1, \ldots, x_k) \in \{x_1, \ldots, x_k\}$, for all $x_1, \ldots, x_k$ in the domain. Let us give explicitly the classification for this problem in the special case of graphs. We call a k-ary operation *near-unanimity*, for $k \geq 3$, if it returns the repeated argument when all but at most one of its arguments is the same. Ternary near-unanimity operations are called *majority.* We refer to a graph as *partially reflexive* to indicate that each vertex may or may not have a self-loop.

▶ **Theorem 2** (From Theorem 5.3 [3] and Theorem 2.1 [15]). *Let $\mathcal{H}^*$ be a partially reflexive graph expanded with all possible unary relations. Then either $\mathcal{H}^*$ admits a conservative majority polymorphism and* CSP*($\mathcal{H}^*$) is in* P*; or* CSP*($\mathcal{H}^*$) is* NP*-complete.*

Contribution

It is easy to see, but does not appear to have been noted, that conservative polymorphisms are expanding polymorphisms *in excelsis.* That is, they are the most natural examples of such polymorphisms that one is likely to imagine.

▶ **Lemma 3.** *Let f be a k-ary operation that is conservative. Then f is also expanding.*

Proof. Consider k subsets of the domain A of f, $A_1, \ldots, A_k$, each of size $m \leq |A|$. We need to argue that $|f(A_1, \ldots, A_k)| \geq m$. We proceed by induction on m where the base case $m = 1$ is trivial. Suppose it holds for m but does not hold for $m+1$. Take $A'_1, \ldots, A'_k$, each of size $m+1 \leq |A|$. There must be $a'_1 \in A'_1, \ldots, a'_k \in A'_k$ so that none of $a'_1, \ldots, a'_k \in f(A'_1, \ldots, A'_k)$, since $|f(A'_1, \ldots, A'_k)| < m + 1$. By inductive hypothesis, $|f(A'_1 \setminus \{a'_1\}, \ldots, A'_k \setminus \{a'_k\})| \geq m$. But $f(a'_1, \ldots, a'_k) \in \{a'_1, \ldots, a'_k\}$ by conservativity, which is a contradiction. ◀

We prove that if a finite structure $\mathcal{B}$ admits an expanding majority polymorphism, then CQCSP($\mathcal{B}$) is in P. In doing so, we answer Question 1 of [21], for the case in the paragraph immediately after it. The algorithm is rather more sophisticated than in the case of CSP or QCSP. We note that a majority that is not expanding can appear as a polymorphism of $\mathcal{B}$ despite that CQCSP($\mathcal{B}$) is NP-hard. We derive as a corollary a complexity classification for CQCSP($\mathcal{H}^*$), where $\mathcal{H}^*$ is a partially reflexive graph endowed with all unary relations. This classification is in line with that of Theorem 2. We further derive a classification for successive approximations to CQCSP($\mathcal{B}$), where $\mathcal{B}$ is a binary first-order expansion of $(\mathbb{Z}; succ)$, whose relations (as digraphs) have bounded-degree. We then make some further observations on the usefulness of expanding majority polymorphisms and relate our work to some recent developments in surjective CSP involving the concept of endo-triviality.

Structure of the paper

This paper is organised as follows. After the preliminaries, Section 3 elaborates the consistency algorithm, and Section 4 gives some applications of this algorithm to complexity classifications. In Section 5, we close with some final remarks about the relationship between List-CSP and CQCSP. Owing to reasons of space, some proofs are deferred to the appendix.

2 Preliminaries

The reader will probably already have picked up that, if $\mathcal{B}$ is a relational structure, then B is its domain and $|B|$ the size of its domain. A *homomorphism*, from a structure $\mathcal{A}$ to a structure $\mathcal{B}$ over the same signature σ, is a function $h : A \to B$ such that, for each relation $R \in \sigma$, if $(x_1, \dots, x_r) \in R^{\mathcal{A}}$, then $(h(x_1), \dots, h(x_r)) \in R^{\mathcal{B}}$. A *$k$-ary polymorphism* of $\mathcal{B}$ is a k-ary operation f on B so that, $(x^1_1, \dots, x^1_r), \dots, (x^k_1, \dots, x^k_r) \in R^{\mathcal{B}}$, then also $(f(x^1_1, \dots, x^k_1), \dots, f(x^1_r, \dots, x^k_r)) \in R^{\mathcal{B}}$.

Given a set B, and an integer $i \geq 0$, we denote its ith power by B^i (B^0 being $\emptyset$). For an integer $c \geq 1$ We write $\binom{B}{c}$ for the following set of subsets of B : $\{S \subseteq B$ such that S has c elements$\}$. A *Skolem (partial) function* g_x for a variable x quantified as $\exists^{\geq c} x$ in the sentence is a partial function to $\binom{B}{c}$, whose arity is the number of variables coming before x in the quantifier prefix of the formula.

The Skolem functions g_i from B^{i-1} to $\binom{B}{c_i}$ $(1 \leq i \leq m)$ *witness that φ holds in $\mathcal{B}$* iff $\forall b_1 \in g_1 \forall b_2 \in g_2(b_1) \dots \forall b_n \in g_n(b_1, b_2, \dots, b_{n-1})$ $\mathcal{B} \models \varphi(b_1, b_2, \dots, b_m)$. If there are such Skolem functions then $\mathcal{B}$ models φ.

For a r-ary relation R in σ and sets $B_1, B_2, \dots, B_r$, we write that $R(B_1, B_2, \dots, B_r)$ holds in $\mathcal{B}$ iff for every $1 \leq i \leq r$ and every b_i in B_i, it is the case that $R(b_1, b_2, \dots, b_r)$ holds in $\mathcal{B}$.

Let us note that counting quantifiers of the same cardinality do not in general commute. In particular, for every choice of $1 < i < n$, there exists a structure $\mathcal{B}$ over the signature of digraph (a single binary predicate E) of size $|B| = n$, such that $\exists^{\geq i} x \exists^{\geq i} y\, E(x, y)$ holds in $\mathcal{B}$ but $\exists^{\geq i} y \exists^{\geq i} x\, E(x, y)$ does not. For more on this, see [21].

3 An algorithm for consistency

In this section we will prove the following main theorem.

▶ **Theorem 4.** *Suppose $\mathcal{B}$ has an expanding majority polymorphism. Then CQCSP($\mathcal{B}$) is in* P.

Just as in the case of CSP and QCSP, by monotonicity, a sentence does not hold if any subsentence does not. Here, by subsentence we mean the sentence induced by selecting some variables. This means that for any structure, a not necessarily complete but polynomial algorithm consists in selecting some subsentences of bounded size and checking whether they hold : if one subsentence fails to hold, then we may answer no. A slightly cleverer way of doing this consists in propagating a potential solution from subsentences with overlapping variables. This is a basic approach known as enforcing local consistency, which is known to imply global consistency for CSP whenever the constraint language is closed under a majority operation [16, 18]. Our algorithm is a careful adaptation to our context.

The consistency argument will be somewhat more fiddly than for CSP. This is due to the fact that quantifiers do not commute and also that we have counting quantifiers and need to keep track of Skolem functions that witness (un)satisfiability of a sentence with counting quantifiers.

The consistency algorithm for establishing our Theorem 4 that we propose does this for the constraints induced by subsentences obtained by selecting up to 3 variables of the prefix and the atoms involving them in the quantifier-free part (we assume w.l.o.g. that the sentence is in prenex form) and maintaining consistency between the witnesses. These witnesses are sets of suitable size, namely the range of the Skolem functions corresponding to the counting quantifiers.

In the following and unless specified otherwise, subset means subset of the domain B of the structure $\mathcal{B}$. We assume some arbitrary order over B and subsets are ordered accordingly.

3.1 Sentences with three variables

Let us examine first a 3 variable sentence φ of the following form :

$$\exists^{\geq c_1} x_1 \exists^{\geq c_2} x_2 \exists^{\geq c_3} x_3 R_{1,2}(x_1, x_2) \wedge R_{2,3}(x_2, x_3) \wedge R_{1,3}(x_1, x_3).$$

For a subset S of size c_1, and subsets T_i of size c_2, we write $OK_{1,2}(S, T_1, \ldots, T_{c_1})$ whenever $R_{1,2}(s_i, T_i)$ holds for all s_i in S (recall that sets are ordered). We proceed similarly to define the $c_1 + 1$-ary predicate $OK_{1,3}$ between a subset of size c_1 and c_1 subsets of size c_3 and the $c_2 + 1$-ary predicate $OK_{2,3}$ between a subset of size c_2 and c_2 subsets of size c_3. The sentence φ holds whenever there is a subset S of size c_1, subsets T_i of size c_2 with $1 \leq i \leq c_1$, subsets $U_{i,j}$ of size c_3 with $1 \leq j \leq c_2$ such that :

$$OK_{1,2}(S, T_1, \ldots, T_{c_1}) \wedge \bigwedge_{1 \leq i \leq c_1} OK_{2,3}(T_i, U_{i,1}, \ldots, U_{i,c_2}) \wedge \bigwedge_{1 \leq j_1 \leq c_2} \cdots \bigwedge_{1 \leq j_{c_1} \leq c_2} OK_{1,3}(S, U_{1,j_1}, \ldots, U_{c_1,j_{c_1}}).$$

3.2 Data structure

With this small example in mind, the following data structure used by our algorithm should become clearer.

- Each variable $\exists^{\geq c_i} x_i$ is represented by a domain that consists of subsets S of size c_i.
- We maintain a $c_i + 1$-ary predicate $OK_{i,j}$ as in the above example between the domains of any pair of variables x_i, x_j as long as x_j comes after x_i in the prefix of quantification and that x_i and x_j occur both in some atom.

3.3 Binary Predicates Only

Of course, unlike in our small example, the input sentence φ' may well have non binary atoms and the parameter structure $\mathcal{B}'$ corresponding relations of arity 3 or more. We project almost in the usual fashion all atoms/relations involving two variables x_1 and x_2 into a single binary constraint R_{x_1,x_2} (if there are constraints, otherwise there is no binary constraint). Unlike in the CSP case, *we check that counting requirements induced by the sentence are met.* Formally, for every pair of distinct variables x_1, x_2 quantified as $\exists^{\geq c_1} x_1 \exists^{\geq c_2} x_2$, we consider the binary constraint R_{x_1,x_2} to be the intersection of the binary relations R'_{x_1,x_2} induced by atoms $R'(\bar{y})$ such that both x_1 and x_2 occur in $\bar{y}$ as follows. $R'_{x_1,x_2}(b_1, b_2)$ holds whenever for any variable y distinct from both x_1 and x_2 with quantifier prefix $\exists^{\geq c_y} y$ occurring at position i in $\bar{y}$ (to distinguish the potentially many occurrences of y, we will write y_i for the occurrence of y at position i) there exists a set B_i of size at least c_y such that $R'(\pi(\bar{y}))$ holds where $\pi(x_1) = b_1, \pi(x_2) = b_2$ and $\pi(y_i) = B_i$.

We denote by φ this sentence with binary atoms and by $\psi(\bar{x})$ its subsentence induced naturally by the variables $\bar{x}$. We write $\mathcal{B}$ for the structure with binary relations. Note that these relations are cq-pp interpretations of the relations of $\mathcal{B}'$.

▶ Proposition 1. If $\mathcal{B}'$ has an expanding majority f, then
(i) $\mathcal{B}$ has also f as an expanding majority
(ii) $\mathcal{B}$ models φ (binary setting) iff $\mathcal{B}'$ models φ' (general setting).

Proof. Since $\mathcal{B}$ was obtained by cq-pp interpretation from $\mathcal{B}'$, it follows that $\mathcal{B}$ has also a majority polymorphism f (via the easy direction of the Galois connection of Lemma 1).

We now show that a collection of Skolem functions witnesses φ iff it does also for φ'. The right to left implication holds by construction and for any structure $\mathcal{B}'$. We only need to establish the left to right implication in the presence of an expanding majority f.

Let $g_1, g_2, \ldots, g_n$ be a collection of Skolem functions witnessing that $\mathcal{B} \models \varphi$. Let $b_1 \in g_1, b_2 \in g_2(b_1) \ldots b_n \in g_n(b_1, b_2, \ldots, b_{n-1})$. We write $g_i(\bar{b})$ as an abbreviation for $g_i(b_1, b_2, \ldots, b_{i-1})$.

Let $R(x_{i_1}, x_{i_2}, \ldots, x_{i_r})$ be some r-ary atom of φ' with $r \geq 3$. We write c_{i_j} to denote the counting requirement on variable i_j for $1 \leq j \leq r$.

Since $R_{x_{i_1},x_{i_2}}(g_{i_1}(\bar{b}), g_{i_2}(\bar{b}))$ holds in $\mathcal{B}$, by construction there are some set of values $S_{i_3}, S_{i_4}, \ldots, S_{i_r}$ of respective sizes $c_{i_3}, c_{i_4}, \ldots, c_{i_r}$. Similarly, there are some sets of the correct count such that $R(g_{i_1}(\bar{b}), S'_{i_2}, g_{i_3}(\bar{b}), S'_{i_4}, \ldots, S'_{i_r})$ and $R(S''_{i_1}, g_{i_2}(\bar{b}), g_{i_3}(\bar{b}), S''_{i_4}, \ldots, S''_{i_r})$. Applying f, since it is a majority, it means the following holds.

$$R(g_{i_1}(\bar{b}), g_{i_2}(\bar{b}), g_{i_3}(\bar{b}), f(S_{i_4}, S'_{i_4}, S''_{i_4}), \ldots, f(S_{i_r}, S'_{i_r}, S''_{i_r})).$$

Since it is expanding, we may select arbitrarily subsets $\widetilde{S_{i_4}} \subseteq f(S_{i_4}, S'_{i_4}, S''_{i_4}) \ldots \widetilde{S_{i_r}} \subseteq f(S_{i_r}, S'_{i_r}, S''_{i_r})$ of respective sizes $c_{i_4}, \ldots, c_{i_r}$ such that the following holds.

$$R(g_{i_1}(\bar{b}), g_{i_2}(\bar{b}), g_{i_3}(\bar{b}), \widetilde{S_{i_4}}, \ldots, \widetilde{S_{i_r}}).$$

Note that there is nothing special about the position 1, 2 and 3 within the tuple R. The same argument applies to any choice of three positions. Furthermore, there is nothing special in our argument using the fact that we have only three positions that agree with the value of the Skolem functions. So we can bootstrap the same argument to extend progressively the tuple by one position and show eventually that : $R(g_{i_1}(\bar{b}), g_{i_2}(\bar{b}), g_{i_3}(\bar{b}), \ldots, g_{i_r}(\bar{b}))$ holds. ◀

From now on, instead of considering a structure $\mathcal{B}'$, in the light of Proposition 1, we will concentrate on the corresponding binary structure $\mathcal{B}$ (to fulfill this we may need to expand the signature but it will still remain finite).

3.4 The Algorithm : path consistency for counting quantifiers (PCCQ)

3.4.0.1 Initialisation

- The domain of x_i contains all subsets that are consistent with all unary atoms involving x_i, that is $\{S \in \binom{B}{c_i}$ such that $S \subseteq M^{\mathcal{B}}$ for every unary atom $M(x_i)$ of $\varphi\}$
- For every binary relation $R_{i,j}$, the predicate $OK_{i,j}$ holds between any set S in the domain of x_i and c_i sets $T_1, \dots, T_{c_i}$ in the domain of x_j whenever $R_{i,j}(s_k, T_k)$ holds for any $1 \leq k \leq c_i$

3.4.0.2 Maintaining consistency

Do
  For all triples of variables $x_{i_1}, x_{i_2}, x_{i_3}$ (in the order of quantification),
    For every distinct k, l in $\{i_1, i_2, i_3\}$,
      For every S in the domain of x_k,
        If there are no OK tuple $OK_{k,l}$ mentioning S (in the first coordinate), then
          discard S and all other OK tuples that mention S.
      For every $OK_{k,l}$ tuple t
        if there are no additional OK tuples witnessing that t participates in a solution to $\varphi(x_{i_1}, x_{i_2}, x_{i_3})$
            Remove the $OK_{k,l}$ tuple t.
            If there are no more $OK_{k,l}$ tuples then reject.
Loop until no further OK tuples are deleted.

3.5 Properties of the PCCQ algoritm

▶ Proposition 2. PCCQ runs in polynomial time.

Proof. Let $\sharp v$ denote the number of variables of φ. The data structure needs to store at most $|B|^j \leq 2^{|B|}$ sets of size at most $j \leq |B|$ for each variable associated with a count j. One OK tuple originating from this variable with count j to a variable with count k will relate at most $j+1$ sets, one of size j and the others of size k. There are therefore at most $2^j.(2^k)^j \leq (2^{|B|})^{|B|+1}$ such OK tuples for one binary constraint. There are at most $\sharp v(\sharp v - 1)$ such constraints. The algorithm runs clearly in time polynomial in these quantities, and $(2^{|B|})^{|B|+1}$ is a constant since $|B|$ is fixed. ◀

Let $\overline{OK_{i,j}(S, T_1, \dots, T_{c_i})}$ be a list of some OK tuples, as many as the arity of an expanding polymorphism f. Applying f coordinate wise, as we would for an ordinary tuple, we have $f(\overline{S}) = S'$ and $f(\overline{T_j}) = T'_j$ for any $1 \leq j \leq c_i$. However, the images $S', T'_1, \dots, T'_{c_i}$ may be too large to feature in an OK tuple. We will say that an OK tuple (with aptly sized sets) $OK_{i,j}(S'', T''_1, \dots, T''_{c_i})$ belongs to $f(\overline{OK_{i,j}(S, T_1, \dots, T_{c_i})})$, whenever $S'' \subseteq S', T''_j \subseteq T'_j$ for all $1 \leq j \leq c_i$.

We say that a set $\mathcal{R}$ of OK tuples are *preserved* by f if, and only if, for any OK tuples $\overline{OK_{i,j}(S, T_1, \dots, T_{c_i})}$ in $\mathcal{R}$, any OK tuple that belongs to $f(\overline{OK_{i,j}(S, T_1, \dots, T_{c_i})})$, also belongs to $\mathcal{R}$.

▶ Proposition 3. Let f be an expanding polymorphism of $\mathcal{B}$. If the algorithm PCCQ does not reject, the OK tuples that remain when the algorithm stops are preserved by f.

Proof. Let $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ be an OK tuple in the image $f(\overline{OK_{i,j}(S, T_1, \ldots, T_{c_i})})$ under f of remaining OK tuples $\overline{OK_{i,j}(S, T_1, \ldots, T_{c_i})}$. We prove that $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ can not be removed by the algorithm as follows.

Initially, the relations are preserved under f, so it is straightforward to verify that OK tuples are also closed under f. So this removal of $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ must happen after initialisation. We shall assume further that $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ is the first OK tuple in the image of f of remaining OK tuples that is removed by the algorithm PCCQ.

Assume further that $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ is removed by the algorithm while checking the sentence with some other variable k. Assume for now that the order of quantification induces the order i, j, k over the indices.

Since the tuples $\overline{OK_{i,j}(S, T_1, \ldots, T_{c_i})}$ are remaining OK tuples, there must be *remaining* tuples $OK_{i,k}$ and $OK_{j,k}$ witnessing that each of them participate in a solution to $\varphi(x_i, x_j, x_k)$.

Taking the image of these witnesses under f provide us with $OK_{i,k}$ and $OK_{i,k}$ witnessing that $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$ participate in a solution to $\varphi(x_i, x_j, x_k)$.

By time minimality of the removal of $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$, these last witnesses may not be remaining tuples but they must remain at the time of removal of $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$. This contradicts the fact that the algorithm could remove $OK_{i,j}(S'', T''_1, \ldots, T''_{c_i})$.

To conclude the proof, note further that the above argument applies independently of the quantification order of i, j and k. ◀

▶ Proposition 4. If $\mathcal{B}$ has an expanding majority f and the algorithm PCCQ does not reject, then $\mathcal{B}$ models φ.

Proof. Let $x_1, x_2, \ldots, x_n$ be the variables occurring in φ. For any choice of variables $\overline{x}$ in $\{x_1, x_2, \ldots, x_n\}$, we denote by $\psi(\overline{x})$ the subsentence of φ induced by the variables $\overline{x}$.

We prove by induction on $2 \leq i < n$ that : for any choice of i variables $\overline{x}$, for any additional variable z occurring after the variables $\overline{x}$ in the order of quantification, any Skolem witnesses $\{g_1, g_2, \ldots, g_i\}$ for $\psi(\overline{x})$ can be extended by an i-ary Skolem function g_z for the variable z such that $\{g_1, g_2, \ldots, g_i, g_z\}$ witnesses that $\varphi(\overline{x}, z)$ holds. Moreover, this Skolem function ranges over sets that were not removed by the algorithm from the domain of z.

The base case for $i = 2$ holds : this is precisely the property that is enforced by the consistency algorithm we outlined.

We proceed to show the induction step. Let $x_1, x_2, x_3, \ldots, x_i$ be a choice of $i \geq 3$ variables and z a variable occurring after them. Let $\{g_1, g_2, g_3, \ldots, g_i\}$ be a collection of Skolem functions witnessing that $\psi(x_1, x_2, x_3, \ldots, x_i)$ holds.

We write I_1 for the image of g_1 and for $1 < j \leq i$, we write I_j for $g_j(I_1, \ldots, I_{j-1})$. Let $\alpha : \emptyset \to I_1$, $\beta : I_1 \to I_2$ and $\gamma : I_1 \times I_2 \to I_3$. We pick only such functions that are consistent with the fact that $\{g_1, g_2, g_3, \ldots, g_i\}$ are Skolem functions, namely we insist that for any b_1 in I_1, $\beta(b_1)$ belongs to the image of $g_2(b_1)$ and for any b_1 in I_1, and any b_2 in $g_2(b_1)$, $\gamma(b_1, b_2)$ lies in the image of $g_3(b_1, b_2)$.

We derive naturally three collections of $i - 1$ Skolem functions by essentially fixing the first, second or third coordinate of the i Skolem functions at hand. Each collection witnesses the subsentence obtained by removal of x_1, x_2 or x_3.

- Let the Skolem functions $\{g_2^\alpha, g_3^\alpha, \ldots, g_i^\alpha\}$ be defined as $g_j^\alpha(x_2, \ldots, x_{j-1}) = g_j(\alpha, x_2, \ldots, x_{j-1})$[1]. By construction, they are witnessing that $\psi(x_2, x_3, \ldots, x_i)$ holds. By the

[1] If g_j is undefined, we let g_j^α be also undefined. Alternatively, we could have defined our Skolem functions precisely where we cared, e.g. for any x_2 in $g_1(\alpha)$, any x_3 in $g(\alpha, x_2)$, etc. But this would only introduce

induction hypothesis, they can be extended by some $(i-1)$-ary function g_z^α witnessing $\psi(x_2, x_3, \ldots, x_i, z)$.

- Similarly, we derive Skolem functions $\{g_1^\beta, g_3^\beta, \ldots, g_i^\beta\}$ witnessing $\psi(x_1, x_3, \ldots, x_i)$ from $\{g_1, g_2, \ldots, g_i\}$ by setting $g_1^\beta = g_1$ and for any $3 \leq j \leq i$, and any b_1 in I_1, we define $g_j^\beta(b_1, x_3, x_4, \ldots, x_{j-1}) := g_j(b_1, \beta(b_1), x_3, \ldots, x_{j-1})$. By the induction hypothesis, they can be extended by some $(i-1)$-ary function g_z^β witnessing $\psi(x_1, x_3, \ldots, x_i, z)$.
- Finally, we derive Skolem functions $\{g_1^\gamma, g_2^\gamma, g_4^\gamma \ldots, g_i^\gamma\}$ witnessing $\psi(x_1, x_2, x_4 \ldots, x_i)$ from $\{g_1, g_2, \ldots, g_i\}$ by setting $g_1^\gamma = g_1$, $g_2^\gamma = g_2$ and for any $4 \leq j \leq i$ any b_1 in I_1 and any b_2 in $g_2(b_1)$ that $g_j^\gamma(b_1, b_2, x_4, \ldots, x_{j-1}) := g_j(b_1, b_2, \gamma(b_1, b_2), x_4, \ldots, x_{j-1})$. By the induction hypothesis, they can be extended by some $(i-1)$-ary function g_z^γ witnessing $\psi(x_1, x_2, x_4, \ldots, x_i, z)$.

We will define the Skolem function g_z piecewise for each choice of the first three variables.

For specific b_1 in I_1 and b_2 in $g_2(b_1)$ and b_3 in $g_3(b_1, b_2)$, we set $\alpha() := b1$, $\beta(b_1) := b_2$, and $\gamma(b_1, b_2) := b_3$. The other values of β and γ are arbitrary but constrained as explained above.

Recall that f is an expanding majority of $\mathcal{B}$.

We define the Skolem function g_z as follows for this choice to the first three variables :

$$g_z(b_1, b_2, b_3, x_4, \ldots, x_i) := f(g_z^\alpha(b_2, b_3, x_4 \ldots, x_i), g_z^\beta(b_1, b_3, x_4 \ldots, x_i), g_z^\gamma(b_1, b_2, x_4 \ldots, x_i)).$$

The fact that f is expanding[2] implies that g_z has a range of correct size.

Note that this definition ensures that indeed g_z ranges over sets that were not filtered out by the algorithm from the domain of z by the (previous) Proposition 3.

The fact that f is a majority will allow us to derive that g_z is indeed an extension of $\{g_1, g_2, g_3, g_4, \ldots, g_i\}$ witnessing $\varphi(x_1, x_2, x_3, x_4 \ldots, x_i, z)$. We need only check this independently for each pair of variables x_j, z, since all atoms are binary. Since, we defined g_z piecewise, we can also check this independently for each piece, induced by the choices of b_1, b_2, b_3. For simplicity, we denote by R an atom that should hold between x_j and z.

- If $j \geq 4$, then applying majority on the variants α, β and γ works naturally, since the value for j is the same for each variant by construction and f is idempotent.
  With full notational details : by assumption $R(g_j^\alpha(b_2, b_3, x_4, \ldots, x_{j-1}), g_z^\alpha(b_2, b_3, x_4 \ldots, x_i))$ holds and $R(g_j^\beta(b_1, b_3, x_4, \ldots, x_{j-1}), g_z^\beta(b_1, b_3, x_4 \ldots, x_i))$ holds and
  $$R(g_j^\gamma(b_1, b_2, x_4, \ldots, x_{j-1}), g_z^\gamma(b_1, b_2, x_4 \ldots, x_i))$$
  holds. By construction of $g_j^\alpha, g_j^\beta, g_j^\gamma$ and the specific choice of values b_1, b_2, b_3, we have
  $$g_j^\alpha(b_2, b_3, x_4, \ldots, x_{j-1}) = g_j^\beta(b_1, b_3, x_4, \ldots, x_{j-1}) = g_j^\gamma(b_1, b_2, x_4, \ldots, x_{j-1}) = g_j(b_1, b_2, b_3, x_4, \ldots, x_{j-1}).$$
  Hence the image of the first coordinate under f is $g_j(b_1, b_2, b_3, x_4, \ldots, x_{j-1})$ since f is idempotent. The second coordinates is precisely the value we defined for g_z. Thus we conclude that $R(g_j(b_1, b_2, b_3, x_4, \ldots, x_{j-1}), g_z(b_1, b_2, b_3, x_4, \ldots, x_i))$ holds as required.
- If $j = 1$. The value for $g_z^\alpha(b_2, b_3, x_4 \ldots, x_i)$ occurs as a set in the domain of the variable z after variable x_1. So the algorithm must have left an OK tuple between x_1 and z that mentions $g_z^\alpha(b_2, b_3, x_4 \ldots, x_i)$. This means that there is a singleton b_1' such that $R(b_1', g_z^\alpha(b_2, b_3, x_4 \ldots, x_i))$ holds. Further, by assumption $R(g_1^\beta, g_z^\beta(b_1, b_3, x_4 \ldots, x_i))$ holds

---

unnecessary notation.

and $R(g_1^\gamma, g_z^\gamma(b_1, b_2, x_4 \dots, x_i))$ holds. Since $g_1^\beta = g_1^\gamma = b_1$, applying f we obtain b_1 for the first coordinate since f is a majority operation. For the second coordinate we obtain the value we defined for g_z. Thus we conclude that $R(b_1, g_z(b_1, b_2, b_3, x_4, \dots, x_i))$ holds as required.

- If $j = 2$, then similarly to the previous case, there is some singleton b_2' in the domain of x_2 such that $R(b_2', g_z^\beta(b_1, b_3, x_4 \dots, x_i))$ holds. Further, by assumption $R(g_2^\alpha(b_1), g_z^\alpha(b_2, b_3, x_4 \dots, x_i))$ holds and $R(g_2^\gamma(b_1), g_z^\gamma(b_1, b_2, x_4 \dots, x_i))$ holds. Since $g_2^\alpha(b_1) = g_2^\gamma(b_1) = b_2$, applying f we obtain b_2 for the first coordinate since f is a majority operation. For the second coordinate we obtain the value we defined for g_z. Thus we conclude that $R(b_2, g_z(b_1, b_2, b_3, x_4, \dots, x_i))$ holds as required.
- If $j = 3$, then similarly to the two previous cases, there is some singleton b_3' in the domain of x_3 such that $R(b_3', g_z^\gamma(b_1, b_2, x_4 \dots, x_i))$ holds. Further, by assumption $R(g_3^\alpha(b_2), g_z^\alpha(b_2, b_3, x_4 \dots, x_i))$ holds and $R(g_3^\beta(b_1), g_z^\beta(b_1, b_2, x_4 \dots, x_i))$ holds. Since $g_3^\alpha(b_2) = g_3^\beta(b_1) = g_3(b_1, b_2) = b_3$, applying f we obtain b_3 for the first coordinate since f is a majority operation. For the second coordinate we obtain the value we defined for g_z. Thus we conclude that $R(b_3, g_z(b_1, b_2, b_3, x_4, \dots, x_i))$ holds as required.

◀

We can now wrap-up to complete the proof of our main theorem.

Proof of Theorem 4. By Proposition 1, we reduce the question whether φ' holds on $\mathcal{B}'$ to the question whether φ holds on $\mathcal{B}$. This can be achieved in polynomial time, since we assume we assume a fixed signature, and have therefore bounded arity. We know that $\mathcal{B}$ is also preserved by the same expanding majority, thus we can appeal to Proposition 4, which states that if PCCQ does not reject then the sentence φ holds in $\mathcal{B}$. Since PCCQ runs in polynomial time by Proposition 2, we are done. ◀

Suppose now that X is some strict subset of $\{1, \dots, |B|\}$. The variant of Lemma 1 that talks of X-pp-definability and polymorphisms that expand at cardinalities in X is not explicit in [8]. However, the easy direction, that X-pp-definability entails preservation by polymorphisms that expand at cardinalities in X, is straightforward to prove.

▶ **Theorem 5.** *Suppose $\mathcal{B}$ has an majority polymorphism that expands at cardinalities $\{c_1, \dots, c_m\}$. Then $\{c_1, \dots, c_m\}$-CSP($\mathcal{B}$) is in* P.

3.6 Expanding polymorphisms are necessary

We will now argue that the condition of expansion was necessary in Theorem 4, since there is a structure admitting non-expanding majority whose CQCSP is NP-hard. Let $\mathcal{H}_4$ be the 4-vertex graph built from the irreflexive triangle $\mathcal{K}_3$ on $\{1, 2, 3\}$ by adding a dominating vertex 0 with a self-loop. It is easy to verify that $\mathcal{H}_4$ enjoys the majority polymorphism f that maps any tuple of distinct arguments to 0. This f is clearly not conservative and it even violates the condition of expansion because $|f(\{0,1\}, \{0,2\}, \{0,3\})| = 1$.

▶ **Lemma 6.** *CQCSP($\mathcal{H}_4$) is* NP*-hard.*

Proof. By reduction from 3-COL, a.k.a. CSP($\mathcal{K}_3$). Take an input φ for CSP($\mathcal{K}_3$) and build an input ψ for CQCSP($\mathcal{H}_4$) by changing all $\exists$ quantifiers to $\exists^{\geq 2}$.

($\mathcal{K}_3 \models \varphi$ implies $\mathcal{H}_4 \models \psi$.) Evaluate each variables v in ψ according to its evaluation φ but additionally with the second possibility 0.

($\mathcal{H}_4 \models \psi$ implies $\mathcal{K}_3 \models \varphi$.) Evaluate each variable v in φ according to one of the possibilities for v in ψ that is not equal to 0. ◀

4 Applications of our result

We will now see that conservative majority polymorphisms demarcate tractability in diverse places.

▶ **Corollary 7.** *Let $\mathcal{H}^*$ be a partially reflexive graph $\mathcal{H}$ endowed with all unary relations. Either $\mathcal{H}^*$ admits an expanding majority and CQCSP($\mathcal{H}^*$) is in* P*, or CQCSP($\mathcal{H}^*$) is* NP*-hard.*

Proof. We know all polymorphisms of $\mathcal{H}^*$ are conservative since it has all unary relations. From Theorem 2 we further know that either $\mathcal{H}^*$ admits a conservative majority polymorphism or CSP($\mathcal{H}^*$) is NP-hard. The result follows from Lemma 3 and Theorem 4. ◀

The following is a strengthening of Theorem 7.16 of [21] in the case of paths.[3]

▶ **Corollary 8.** *Let $\mathcal{P}$ be an irreflexive (undirected) path. Then CQCSP($\mathcal{P}$) is in* P*.*

Proof. Suppose $\mathcal{P}$ is over vertices $\{1, \ldots, n\}$ so that $(i, i+1) \in E^{\mathcal{P}}$. Then $\mathcal{P}$ admits the conservative majority polymorphism m communicated to us by Tomás Feder: $m(x, y, z)$ is defined to be the median of x, y, z, if they all have the same parity; otherwise it is the smaller of the pair with repeated parity. The result follows from Lemma 3 and Theorem 4. ◀

Sadly we cannot use conservative majorities for irreflexive trees, since it is well-known that the tree $\mathcal{T}_{10}$, built from three paths on four vertices by identifying one end of each of these three paths as a single vertex, does not admit a conservative majority. This has been known, based on complexity-theoretic assumptions, since [14, 4] but we have checked also using the polymorphism program of Miklós Maróti[4].

We will now see how to apply our result to infinite-domain (CQ)CSPs. The (d-)*modular median* operation of [1] is defined on $\mathbb{Z}$ as follows. $f(x, y, z) = \mathit{median}(x, y, z)$, if $x \equiv y \equiv z \bmod d$. If two among $\{x, y, z\}$ are equivalent mod d, then $f(x, y, z)$ is the minimum of these two; otherwise $f(x, y, z) = x$. Note that these modular median operations are conservative majorities.

▶ **Corollary 9.** *Let $\mathcal{B}$ be a finite-signature binary first-order expansion of $(\mathbb{Z}; \mathit{succ})$ whose relations, viewed as digraphs, have bounded degree. Either $\mathcal{B}$ admits a modular median polymorphism, and, for each j, $\{1, \ldots, j\}$-CSP($\mathcal{B}$) is in* P*, or CSP($\mathcal{B}$) is* NP*-hard.*

Proof. By Proposition 6 in [1],[5] we know that if $\mathcal{B}$ omits all modular median operations, then CSP($\mathcal{B}$) is NP-hard. Thus, we are left with the question of tractability. Let e be maximal so that $(x, x+e)$ appears in some relation of $\mathcal{B}$. Let ϕ be an input for CQCSP($\mathcal{B}$) involving n variables. Now, we can see that ϕ is true on $\mathcal{B}$ just in case it is true on the substructure $\mathcal{B}'$of $\mathcal{B}$ induced by the interval $[0, ne]$. $\mathcal{B}'$ admits the same conservative majority that $\mathcal{B}$ does and the result follows from Propositions 4 when we consider from the proof of Proposition 2 that the size of subsets in the OK tuples is bounded by j. This is because the number of OK tuples per binary constraint is bound by $(j(ne)^j)^{j+1}$ (which takes the place of the term $(2^{|B|})^{|B|+1}$ in the calculation for complexity in Proposition 2). ◀

---

[3] Theorem 7.16 of [21] deals with $\{1, 2\}$-CSP on trees, but its very long proof does not become much simpler if one restricts to paths.

[4] See: `http://www.math.u-szeged.hu~maroti/applets/GraphPoly.html`

[5] Proposition 6 lacks a counterpart in the journal version of [1] For a proof, see Proposition 35 in v2 of the arxiv version.

To consider CQCSP over an infinite-domain structure, albeit with a finite signature, one must consider how to encode i in $\exists^{\geq i}$. The most natural encoding here is binary. We leave as an open question whether CQCSP($\mathcal{B}$) is in P, whenever $\mathcal{B}$ is a finite-signature binary first-order expansion of ($\mathbb{Z}$; *succ*) whose relations, viewed as digraphs, have bounded degree, which admits a modular median polymorphism. Note that this question remains open even if we choose the unary encoding for i.

4.1 Endo-triviality

The concept of endo-triviality has recently been introduced in the context of surjective CSPs [20]. We note here that endo-triviality is strong enough to deduce results also for CQCSPs. An *endomorphism* of a digraph $\mathcal{H}$ is a homomorphism from $\mathcal{H}$ to itself. Call $\mathcal{H}$ a *core* if all of its endomorphisms of $\mathcal{H}$ are automorphisms (the importance of cores is discussed, e.g., in [17]). Call $\mathcal{H}$ *endo-trivial* if all of its endomorphisms either have range of size 1 or are automorphisms.

The *retraction* problem Ret($\mathcal{H}$) takes as input a graph $\mathcal{G}$ containing $\mathcal{H}$ as an induced substructure and asks whether there is a homomorphism from $\mathcal{G}$ to $\mathcal{H}$ that is the identity on $\mathcal{H}$ (such an endomorphism of $\mathcal{G}$ is termed a *retraction to* $\mathcal{H}$)

The proofs of the following are deferred to the appendix.

▶ **Lemma 10.** *Let $\mathcal{H}$ be a graph that is endo-trivial. The there is a polynomial-time reduction from Ret($\mathcal{H}$) to CQCSP($\mathcal{H}$).*

▶ **Corollary 11.** *Let $\mathcal{C}$ be a reflexive directed cycle. If $\mathcal{C}$ is of length* 2 *then CQCSP($\mathcal{C}$) is in* L*, otherwise CQCSP($\mathcal{C}$) is* NP*-hard.*

5 Final remarks

Near-unanimity polymorphisms. Note that Theorem 4 relativises to any subset of counts $X \subset \{1, 2 \ldots, |B|\}$ for the problem X-CSP($\mathcal{B}$) with the weaker hypothesis that requires that $\mathcal{B}$ has a majority f that is expanding on X. Note that, if $1 \notin X$, one has to move to partial polymorphisms. Indeed, we do not need f to be a majority, only that it satisfies the identities of a majority where we replace uniformly the variables by set variables of the same size from X.

We can also generalise the algorithm and the proof principle to a larger class of structures.

▶ **Theorem 12.** *If $\mathcal{B}$ has an expanding near unanimity polymorphism. Then CQCSP($\mathcal{B}$) is in* P.

CQCSP and List-CSP. We have seen that conservative operations are expanding, but what is the actual relationship between CQCSP and List-CSP? Does ability to quantify set cardinalities with $\exists^{\geq j}$ relate to talking about subsets of size j? For this latter question, it seems the answer is no. Designate $\{1, 2\}$-List-CSP the restriction of List-CSP in which only subsets of size 1 and 2 are available. Recall the tree $\mathcal{T}_{10}$, built from three paths on four vertices by identifying one end of each of these three paths as a single vertex. List-CSP($\mathcal{T}_{10}$) is known to be NP-complete since [14]. NP-completeness for $\{1, 2\}$-List-CSP($\mathcal{T}_{10}$) follows from [4]. On the other hand, $\{1, 2\}$-CSP($\mathcal{T}_{10}$) is in P, as proved in Theorem 7.16 of [21]. However, we are still missing an exemplar $\mathcal{B}$ so that one of CQCSP($\mathcal{B}$) and List-CSP($\mathcal{B}$) is tractable and the other is not.

CQCSP and Retraction In Lemma 10, we show a sufficient condition for which Ret($\mathcal{B}$) is polynomially reducible to CQCSP($\mathcal{B}$). It should be possible to reconstruct the argument from [20] in order to prove that, if $\mathcal{H}$ is a reflexive tournament, then either $\mathcal{H}$ has a conservative majority polymorphism (the median) and CQCSP($\mathcal{B}$) is in P; or Ret($\mathcal{H}$) can be polynomially reduced to CQCSP($\mathcal{H}$) and both are NP-hard. Note that a classification for QCSP on reflexive tournaments is not yet known. However, what we would like is much stronger : is it the case that for all finite $\mathcal{B}$, Ret($\mathcal{H}$) can be polynomially reduced to CQCSP($\mathcal{H}$)? That is, are all constants cq-pp-definable up to isomorphism?

Core-ness and finite categoricity Closely related to the previous question is whether all non-isomorphic finite structures can be distinguished by cq-pp. Let us explore this question through the Weisfeiler-Lehman (WL) method, as discussed in [9] (where logics with counting also play a central role). The *degree sequence* of a graph is a non-increasing list of positive integers that list the degrees of its vertices. This can be thought of as a 0-dimensional WL descriptor. Obviously, if two graphs are isomorphic, then they have the same degree sequence, but the converse is not necessarily true. Cq-pp can not specify vertex degree but it can specify a lower bound for it. Firstly, then, two graphs on vertex sets of distinct sizes can be distinguished by some $\exists^{\geq a_1} x\,(x = x)$. For two graphs with vertex sets the same size, if their two degree sequences differ, with the first being lexicographically the larger, then counting down from the top until the first difference, one will find necessarily some a_1, a_2 so that $\exists^{\geq a_1} x_1 \exists^{\geq a_2} x_2\, E(x_1, x_2)$ is true on the first graph but false on the second. We do this by setting $a_1 - 1$ to be the number of vertices before the degree sequence differs and a_2 to be the degree at which the degree sequences diverge.

The 1-dimensional WL descriptor is defined inductively by expanding each integer associated with a vertex from the 0-dimensional WL descriptor into a tree of depth one whose leaves list, in descending order, the degrees of that vertex's neighbours. These leaves are now associated with that corresponding neighbour. The process is then iterated, and would go on for ever, save that we stop it when a fixed-point is reached in terms of the subtrees added being endlessly the same. Now suppose two graphs each give rise to a forest built in this fashion and let k be the height at which these forests first differ (else they are indistinguishable by 1-dimensional WL) and let the first graph be lexicographically the smaller (apply closeness to the root as higher in the lexicography). We can follow the previous reasoning, and the path through the forests on which the graphs differ, to find some $\exists^{\geq a_1} x_1 \exists^{\geq a_2} x_2 \ldots \exists^{\geq a_k} x_k \exists^{\geq a_{k+1}} x_{k+1}\, E(x_1, x_2) \wedge \ldots \wedge E(x_k, x_{k+1})$ that is true on the first graph but not the second.

The 1-dimensional WL descriptor does not capture isomorphism, and unfortunately, we do not see an implementation of the more general r-dimensional WL descriptor in cq-pp, since this can measure isomorphism type of an induced subgraph of size r.

References

1 Manuel Bodirsky, Víctor Dalmau, Barnaby Martin, and Michael Pinsker. Distance constraint satisfaction problems. In *Mathematical Foundations of Computer Science 2010, 35th International Symposium, MFCS 2010, Brno, Czech Republic, August 23-27, 2010. Proceedings*, pages 162–173, 2010. URL: https://doi.org/10.1007/978-3-642-15155-2_16, doi:10.1007/978-3-642-15155-2_16.

2 F. Börner, A. Krokhin, A. Bulatov, and P. Jeavons. Quantified constraints and surjective polymorphisms. Technical Report PRG-RR-02-11, Oxford University, 2002.

3 Richard C. Brewster, Tomás Feder, Pavol Hell, Jing Huang, and Gary MacGillivray. Near-unanimity functions and varieties of reflexive graphs. *SIAM J. Discrete Math.*,

22(3):938–960, 2008. URL: https://doi.org/10.1137/S0895480103436748, doi:10.1137/S0895480103436748.

4 A. Bulatov. Tractable conservative constraint satisfaction problems. In *Proceedings of LICS'03*, pages 321–330, 2003.

5 A. Bulatov, A. Krokhin, and P. G. Jeavons. Classifying the complexity of constraints using finite algebras. *SIAM Journal on Computing*, 34:720–742, 2005.

6 Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs. *Proc. FOCS 2017*, pages 319–330, 2017.

7 Andrei A. Bulatov and Amir Hedayaty. Counting predicates, subset surjective functions, and counting csps. In *42nd IEEE International Symposium on Multiple-Valued Logic, ISMVL 2012*, pages 331–336, 2012.

8 Andrei A. Bulatov and Amir Hedayaty. Galois correspondence for counting quantifiers. *Multiple-Valued Logic and Soft Computing*, 24:405–424, 2015.

9 Jin-Yi Cai, Martin Fürer, and Neil Immerman. An optimal lower bound on the number of variables for graph identification. *Combinatorica*, 12(4):389–410, Dec 1992. URL: https://doi.org/10.1007/BF01305232, doi:10.1007/BF01305232.

10 Hubie Chen. The complexity of quantified constraint satisfaction: Collapsibility, sink algebras, and the three-element case. *SIAM J. Comput.*, 37(5):1674–1701, 2008. doi:http://dx.doi.org/10.1137/060668572.

11 Robin Clark and Murray Grossman. Number sense and quantifier interpretation. *Topoi*, 26(1):51–62, 2007.

12 H.-D. Ebbinghaus and J. Flum. *Finite Model Theory.* Springer, 1999. 2nd edition.

13 T. Feder and M. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory. *SIAM Journal on Computing*, 28:57–104, 1999.

14 Tomás Feder, Pavol Hell, and Jing Huang. List homomorphisms and circular arc graphs. *Combinatorica*, 19(4):487–505, 1999.

15 Tomás Feder, Pavol Hell, and Jing Huang. Bi-arc graphs and the complexity of list homomorphisms. *J. Graph Theory*, 42:61–80, 2003.

16 E.C. Freuder. A sufficient condition for backtrack-free search. *Journal of the ACM*, 29(1):24–32, 1982.

17 P. Hell and J. Nešetřil. *Graphs and Homomorphisms.* OUP, 2004.

18 Peter Jeavons, David Cohen, and Martin Cooper. Constraints, consistency and closure. *AI*, 101(1-2):251–265, 1998.

19 P. G. Kolaitis and M. Y. Vardi. *Finite Model Theory and Its Applications (Texts in Theoretical Computer Science. An EATCS Series)*, chapter A logical Approach to Constraint Satisfaction. Springer-Verlag New York, Inc., 2005.

20 Benoit Larose, Barnaby Martin, and Daniël Paulusma. Surjective h-colouring over reflexive digraphs. In Rolf Niedermeier and Brigitte Vallée, editors, *35th Symposium on Theoretical Aspects of Computer Science, STACS 2018, February 28 to March 3, 2018, Caen, France*, volume 96 of *LIPIcs*, pages 49:1–49:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018. URL: https://doi.org/10.4230/LIPIcs.STACS.2018.49, doi:10.4230/LIPIcs.STACS.2018.49.

21 Barnaby Martin, Florent R. Madelaine, and Juraj Stacho. Constraint satisfaction with counting quantifiers. *SIAM J. Discrete Math.*, 29(2):1065–1113, 2015. URL: https://doi.org/10.1137/140981332, doi:10.1137/140981332.

22 M. Otto. *Bounded variable logics and counting – A study in finite models*, volume 9. Springer-Verlag, 1997. IX+183 pages.

23 Dmitriy Zhuk. A proof of CSP dichotomy conjecture. *Proc. FOCS 2017*, pages 331–342, 2017.