Affine Systems of Equations and Counting Infinitary Logic*

Albert Atserias[1]** and Andrei Bulatov[2] and Anuj Dawar[3]

[1] Universitat Politécnica de Catalunya, Barcelona, Spain.
[2] Simon Fraser University, Burnaby BC, Canada.
[3] University of Cambridge Computer Laboratory, Cambridge, UK.

Abstract. We study the definability of constraint satisfaction problems (CSP) in various fixed-point and infinitary logics. We show that testing the solvability of systems of equations over a finite Abelian group, a tractable CSP that was previously known not to be definable in Datalog, is not definable in an infinitary logic with counting and hence that it is not definable in least fixed point logic or its extension with counting. We relate definability of CSPs to their classification obtained from tame congruence theory of the varieties generated by the algebra of polymorphisms of the template structure. In particular, we show that if this variety admits either the unary or affine type, the corresponding CSP is not definable in the infinitary logic with counting. We also study the complexity of determining whether a CSP omits unary and affine types.

1 Introduction

The classification of constraint satisfaction problems (CSP) according to their tractability has been a major research goal since Feder and Vardi first formulated their dichotomy conjecture [1]. The general form of the *constraint satisfaction problem* takes as instance two finite relational structures $\mathbf{A}$ and $\mathbf{B}$ and asks if there is a homomorphism from $\mathbf{A}$ to $\mathbf{B}$. We think of the elements of $\mathbf{A}$ as the variables of the problem and the universe of $\mathbf{B}$ as the domain of values which these variables may take. The individual tuples in the relations of $\mathbf{A}$ act as constraints on the values that must be matched to the relations holding in $\mathbf{B}$. The general form of the problem is NP-complete. In this paper we are mainly concerned with the non-uniform version of the problem which gives rise, for each fixed finite structure $\mathbf{B}$ to a different decision problem that we denote CSP($\mathbf{B}$), namely the problem of deciding whether a given $\mathbf{A}$ maps homomorphically to $\mathbf{B}$. For many fixed $\mathbf{B}$, this problem is solvable in polynomial time, while for others it remains NP-complete.

In the present paper we are concerned with classifying constraint satisfaction problems according to their definability in a suitable logic. This is an approach that has proved useful in studying the tractability of constraint satisfaction problems [1–3]. In particular, it is known that many natural constraint satisfaction

* Research supported by the Isaac Newton Institute, LAA programme.
** Supported in part by CICYT TIN2004-04343

problems that are tractable are definable (or, to be precise, their complements are definable) in Datalog, the language of function-free Horn clauses. Any class of structures that is definable in Datalog is necessarily decidable in polynomial time, but there are known constraint satisfaction problems that are tractable but are not definable in Datalog. A classical example is the solvability of systems of linear equations over the two-element field [1], which we denote $\mathrm{CSP}(\mathbb{Z}_2)$. Bulatov [4] (see also [5]) provides a uniform explanation for the tractability of these by showing that any constraint language that has a Mal'tsev polymorphism is solvable in polynomial time. Furthermore, there are NP-complete constraint satisfaction problems, such as 3-colourability of graphs, which we can show are not Datalog-definable, without requiring the assumption that P is different from NP. Indeed, the class of constraint satisfaction problems whose complements are definable in Datalog appears to be a robust, natural class of problems with many independent and equivalent characterisations [6, 7].

A natural question that arises is whether we can offer any explanation based on logical definability for the tractability of problems such as the satisfiability of systems of linear equations over a finite field. Is there a natural logic such that all problems definable in this logic are polynomial-time decidable and that can express $\mathrm{CSP}(\mathbb{Z}_2)$? In particular, is this problem definable in LFP—the extension of first-order logic with least fixed points or $\mathrm{LFP} + \mathrm{C}$—the extension of LFP with counting? Both these logics have been extensively studied in the context of descriptive complexity as characterising natural fragments of polynomial time. Interestingly, Blass et al. [8] proved that $\mathrm{LFP} + \mathrm{C}$ is able to define the class of non-singular square matrices over any fixed finite field, so it would not be very surprising if this logic were able to express $\mathrm{CSP}(\mathbb{Z}_2)$. Despite this, it is a consequence of our results that neither of these logics is able to express the solvability of systems of linear equations over any finite field. Indeed, we show that these problems are not definable in $\mathrm{C}^{\omega}_{\infty\omega}$, the infinitary logic with bounded number of variables and counting, a logic much more expressive than $\mathrm{LFP} + \mathrm{C}$. Combined with the result of Blass, Gurevich and Shelah about non-singular matrices, our result exhibits a fine-grained distinction between the problem of computing the rank of a square matrix and the problem of computing its determinant.

Another important means of classifying constraint satisfaction problems is on the basis of the algebra of the template structure $\mathbf{B}$. A polymorphism of a structure is an operation of its universe that preserves all its relations (see Section 2 for precise definitions). It is known that whether or not $\mathrm{CSP}(\mathbf{B})$ is tractable depends only on the algebra $\mathcal{B}$ obtained from the universe of $\mathbf{B}$ endowed with its polymorphisms. Indeed, it depends only on the variety generated by this algebra. This is established in [9] by showing that if the algebra $\mathcal{B}'$ of structure $\mathbf{B}'$ is obtained from $\mathcal{B}$ as a power, subalgebra or homomorphic image, then $\mathrm{CSP}(\mathbf{B}')$ is polynomial-time reducible to $\mathrm{CSP}(\mathbf{B})$. We show in the present paper that this can be improved to Datalog-definable reductions. These are weak reductions that, in particular, preserve definability in LFP and $\mathrm{C}^{\omega}_{\infty\omega}$. This allows us to establish that definability of a CSP in these logics is also determined by $\mathrm{var}(\mathcal{B})$, the variety generated by the algebra of $\mathbf{B}$.

Using the tool of Datalog-reductions, which we expect to be useful for other applications in the area, we relate definability of constraint satisfaction problems in $C^{\omega}_{\infty\omega}$ to the classification of varieties of finite algebras from tame congruence theory [10]. It is known [9] that CSP(**B**) is NP-complete if var($\mathcal{B}$) admits the unary type (also known as type **1**), and it is conjectured that CSP(**B**) is in P otherwise. Similarly, Larose and Zádori showed [11] that CSP(**B**) is not definable in Datalog if var($\mathcal{B}$) admits the unary or affine types (types **1** and **2**), and conjectured the converse. It is a consequence of our results that we can strengthen the assertion by replacing Datalog with $C^{\omega}_{\infty\omega}$. This implies that, if the Larose-Zádori conjecture is true, we obtain a dichotomy of definability whereby, for every **B**, either CSP(**B**) is definable in Datalog or it is not definable in $C^{\omega}_{\infty\omega}$.

Finally, we consider the meta-problems of deciding, given a structure **B** or an algebra $\mathcal{B}$ whether or not var($\mathcal{B}$) omits the unary and affine types. For algebras, the problem was shown decidable in polynomial time in [12], while for structures we show it is NP-complete.

2 Preliminaries

Structures and graphs A vocabulary σ is a finite collection of relation symbols, each with an associated arity. A σ-structure $\mathbf{A}$ consists of a finite set A with a relation $R^{\mathbf{A}} \subseteq A^r$ for each r-ary relation symbol R in σ. A graph is a structure with a binary relation that is symmetric and irreflexive. A *homomorphism* from a σ-structure $\mathbf{A}$ to a σ-structure $\mathbf{B}$ is a map $h : A \to B$ such that for each R in σ and each $\mathbf{a} \in A^r$, if $\mathbf{a} \in R^{\mathbf{A}}$ then $h(\mathbf{a}) \in R^{\mathbf{B}}$. We write $\mathbf{A} \to \mathbf{B}$ to denote that there exists a homomorphism from $\mathbf{A}$ to $\mathbf{B}$. We write CSP(**B**) for the class of finite structures $\mathbf{A}$ such that $\mathbf{A} \to \mathbf{B}$ and also for the decision problem of determining membership in this class.

For the standard definition of the treewidth of a graph, we refer the reader to [13]. In our proofs we will use the following alternative characterization in terms of the *cops and robber game* [14]. The game is played by two players, one of whom controls the set of k cops attempting to catch a robber controlled by the other player. The cop player can move any set of cops to any vertices of the graph, while the robber can *simultaneously* move along any path in the graph as long as there is no cop currently on the path. It is known [14] that the cop player has a winning strategy on a graph using $k+1$ cops if and only if the graph has treewidth at most k. The treewidth of a graph G is denoted $tw(G)$.

Logic A formula is *positive quantifier-free* if it is formed from the atomic formulas using conjunctions and disjunctions. A formula is *existential positive* if it is formed from the atomic formulas using conjunctions, disjunctions and existential quantification. Datalog is the extension of *existential positive formulas* with a recursion mechanism. Similarly, LFP is the extension of *full first-order logic* with an operator for forming the least fixed points of positive formulas. Finally, LFP+C is the extension of LFP with a counting mechanism. For formal definitions, which we will not need in this paper, we refer the reader to [15]. It is known that every class of structures definable in LFP + C is decidable in polynomial time.

The formulas of the logic $C_{\infty\omega}$ are obtained from the atomic formulas using negation, infinitary conjunction and disjunction, and counting quantifiers ($\exists^i x\phi$ for any integer $i \geq 0$). The fragment $C^k_{\infty\omega}$ consists of those formulas of $C_{\infty\omega}$ in which only k distinct variables appear and $C^\omega_{\infty\omega} = \bigcup_{k\in\omega} C^k_{\infty\omega}$. The significance of $C^\omega_{\infty\omega}$ is that fixed-point logics can be translated into it. That is, any formula of Datalog or LFP, and indeed of LFP + C is equivalent to one of $C^\omega_{\infty\omega}$. Moreover, these translations into infinitary logics have provided some of the most effective tools for proving inexpressibility results for the fixed-point logics. See [16, 17] for a discussion of this and the role of these logics in descriptive complexity.

The expressive power of $C^\omega_{\infty\omega}$ is characterised by a game known as the *bijective game* [18]. This is played by two players, Spoiler and Duplicator, on a pair of structures **A** and **B**, with k pairs of pebbles (x_i, y_i) for $1 \leq i \leq k$. At each stage of the game, some of the pebbles may be on elements of the structures with x_i on an element of A and y_i on an element of B. We write a_i for the element currently pebbled by x_i, and b_i for the element pebbled by y_i. For each move, Spoiler chooses a pair of pebbles (x_i, y_i), Duplicator chooses a bijection $f : A \to B$ such that $f(a_j) = b_j$ for $i \neq j$, and Spoiler chooses $a \in A$ and places x_i on a and y_i on $f(a)$. If, after some move, the map $a_i \mapsto b_i (1 \leq i \leq k)$ is not a partial isomorphism, Spoiler wins; Duplicator wins infinite plays. By a result of Hella [18], Duplicator has a winning strategy if, and only if, **A** and **B** cannot be distinguished by any formula of $C^k_{\infty\omega}$, a fact denoted by $\mathbf{A} \equiv^{C^k} \mathbf{B}$.

Universal algebra An n-ary operation f on a set A is a *polymorphism* of a relation $R \subseteq A^r$ if, for any tuples $\mathbf{a}_1, \dots \mathbf{a}_n \in R$, the r-tuple obtained by applying f component-wise also belongs to R. We say that R is invariant under f.

A set with a collection of operations on it is called an *algebra*. Every structure **A** can be naturally associated with an algebra $\mathrm{Al}(\mathbf{A})$, called the *algebra* of **A**, whose base set is the universe of **A**, and whose operations are the polymorphisms of **A**. A *variety* is a class of algebras which, if it contains $\mathcal{A}$ also contains every subalgebra of $\mathcal{A}$, every homomorphic image of $\mathcal{A}$, and every direct power of $\mathcal{A}$. The smallest variety containing $\mathcal{A}$ is called the *variety generated* by $\mathcal{A}$ and denoted by $\mathrm{var}(\mathcal{A})$. For further background on universal algebra, see [19].

3 Definability of Equations

In this section we show that the problem of determining the solvability of linear equations over the two-element field, which we mentioned above as a canonical example of a tractable CSP whose complement is not definable in Datalog, is also not definable in $C^\omega_{\infty\omega}$. Indeed, we prove a more general result by showing that the solvability of equations over a finite Abelian group $\mathcal{G}$ with at least two elements is not definable in $C^\omega_{\infty\omega}$. In the following we will write + for the group operation in $\mathcal{G}$ and 0 for the identity.

Consider the following formulation of the problem.

Definition 1. *Let $\mathcal{G}$ be a finite Abelian group over a set G and r be a positive integer. We define the structure $\mathbf{E}_{\mathcal{G},r}$ to have universe G and, for each $a \in G$*

and $1 \leq j \leq r$*, it has a relation* R_a^j *of arity* j *that consists of the set of tuples* $(x_1, \dots, x_j) \in G^j$ *that satisfy the equation* $x_1 + \cdots + x_j = a$*.*

Thus, any structure $\mathbf{A}$ in the signature of $\mathbf{E}_{\mathcal{G},r}$ can be seen as a set of equations in which at most r variables occur in each equation. The universe of $\mathbf{A}$ is the set of variables and the occurrence of a tuple $(x_1, \dots, x_j)$ in a relation R_a^j signifies the equation $x_1 + \cdots + x_j = a$. This set of equations is solvable if, and only if, $\mathbf{A} \to \mathbf{E}_{\mathcal{G},r}$. In the sequel we will say "the equation $x_1 + \cdots + x_j = a$ occurs in $\mathbf{A}$" to mean that the tuple $(x_1, \dots, x_j)$ is in R_a^j.

Our aim now is to exhibit, for each non-trivial finite Abelian group $\mathcal{G}$ and each positive integer k, a pair of structures $\mathbf{A}$ and $\mathbf{B}$ such that $\mathbf{A} \equiv^{\mathrm{C}^k} \mathbf{B}$ and such that $\mathbf{A} \in \mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ and $\mathbf{B} \notin \mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$. This will show that $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ is not definable in $\mathrm{C}^{\omega}_{\infty\omega}$. This, of course, implies the result for all $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},r})$ with $r \geq 3$. The structures we construct are sets of equations derived from 3-regular graphs of large treewidth. From now on, fix a non-trivial finite Abelian group $\mathcal{G}$ over a set G, a 3-regular graph H, and a distinguished vertex u of H. We define, for each $a \in G$, a set of equations $\mathbf{E}_a H^u$ as follows (note that $\mathbf{E}_a H^u$ is a structure over the vocabulary of $\mathbf{E}_{\mathcal{G},3}$):

For each vertex $v \in V^H$ and each edge $e \in E^H$ that is incident on v, we have m distinct variables $x_i^{v,e}$ where i ranges over G. Since each vertex has three edges incident on it, there are $3m$ variables associated to each vertex. For every vertex v *other than* u, let e_1, e_2, e_3 be the three edges incident on v. We then include the following equation in $\mathbf{E}_a H^u$ for all $i, j, k \in G$:

$$x_i^{v,e_1} + x_j^{v,e_2} + x_k^{v,e_3} = i + j + k. \tag{1}$$

For the distinguished vertex u, instead of the above, we include the following equation, again for all $i, j, k \in G$:

$$x_i^{u,e_1} + x_j^{u,e_2} + x_k^{u,e_3} = i + j + k + a. \tag{2}$$

In addition, for each edge $e \in E^H$ let v_1, v_2 be its endpoints. We include the following equations in $\mathbf{E}_a H^u$ for all $i, j \in G$:

$$x_i^{v_1,e} + x_j^{v_2,e} = i + j. \tag{3}$$

We refer to equations of the form (1) and (2) as *vertex equations* and equations of the form (3) as *edge equations.*

Lemma 2. $\mathbf{E}_a H^u$ *is satisfiable if, and only if,* $a = 0$

Proof. To see that $\mathbf{E}_0 H^u$ is satisfiable, just take the assignment that gives the variable $x_i^{v,e}$ the value i. To see that $\mathbf{E}_a H^u$ is unsatisfiable when $a \neq 0$, consider the subsystem S_0 of equations involving only the variables $x_0^{v,e}$ with subscript 0. Note that each such variable occurs exactly twice in S_0, once in a vertex equation and once in an edge equation. Thus, if we add up the left hand sides of all equations in S_0, we get $2 \sum x_0^{v,e}$. Note also that each variable $x_0^{v,e}$ has a companion variable $x_0^{v',e}$ where v' is the other endpoint of the edge e and we have the equation $x_0^{v,e} + x_0^{v',e} = 0$. Thus $2\sum_{v,e} x_0^{v,e} = 2\sum_e (x_0^{v,e} + x_0^{v',e}) = 0$. On the other hand, the right-hand side of all equations is 0 except for the one vertex equation for u, which has right-hand side a. Thus summing the right-hand sides of all equations gives the sum a. Since $a \neq 0$, this shows that the subsystem S_0 and hence the system of equations $\mathbf{E}_a H^u$ is unsatisfiable.

Lemma 3. *If* $\mathrm{tw}(H) > k$ *and* H *is connected, then* $\mathbf{E}_0 H^u \equiv^{\mathrm{C}^k} \mathbf{E}_a H^u$ *for any* $a \in \mathcal{G}$.

Proof. Our aim is to exhibit a winning strategy for Duplicator in the k-pebble bijective game played on the two structures $\mathbf{A} = \mathbf{E}_0 H^u$ and $\mathbf{B} = \mathbf{E}_a H^u$. Since $\mathrm{tw}(H) > k$, we know that in the k cops and robber game played on H, robber has a winning strategy and Duplicator will make use of this strategy.

For each vertex $v \in V^H$ let X^v denote the set of variables $x_i^{v,e}$ for edges e incident on v. Similarly, for each $e \in E^H$, let X^e denote the set of variables involving e.

We say that a bijection $f : \mathbf{A} \to \mathbf{B}$ is *good* for a vertex $v \in V^H$ if the following conditions hold:

1. for all $w \in V^H$, $fX^w = X^w$;
2. for all $e \in E^H$, $fX^e = X^e$;
3. for all x, y, if $x + y = i$ is an equation in $\mathbf{A}$ then $f(x) + f(y) = i$ is an equation in $\mathbf{B}$; and
4. for all x, y, z, if $x + y + z = i$ is an equation in $\mathbf{A}$, then
   - $f(x) + f(y) + f(z) = i$ is an equation in $\mathbf{B}$ if $x, y, z \notin X^v$; and
   - $f(x) + f(y) + f(z) = i + a$ is an equation in $\mathbf{B}$ if $x, y, z \in X^v$.

Note that the identity is a bijection that is good for u. Also, note that a bijection that is good for v preserves all equations except the vertex equations for v.

Claim. Given a bijection $f : \mathbf{A} \to \mathbf{B}$ that is good for v, if there is a path in H from v to w avoiding $u_1, \dots, u_k$ then there is a bijection $f' : \mathbf{A} \to \mathbf{B}$ that is good for w such that $f|_{(X^{u_1} \cup \cdots \cup X^{u_k})} = f'|_{(X^{u_1} \cup \cdots \cup X^{u_k})}$.

Proof. : Let the path from v to w avoiding $u_1, \dots, u_k$ be $v = v_1, \dots, v_n = w$. For each edge $e = \{v_i, v_{i+1}\}$ along this path, write x_j^{e-} for the variable $x_j^{v_i,e}$ and x_j^{e+} for the variable $x_j^{v_{i+1},e}$. We then define f' by $f'(x_j^{e-}) = f(x_{j-a}^{e-})$ and $f'(x_j^{e+}) = f(x_{j+a}^{e+})$; and f' agrees with f everywhere else. In particular, since the path from v to w avoids $u_1, \dots, u_k$, f' agrees with f on $X^{u_1} \cup \cdots \cup X^{u_k}$.

We now describe Duplicator's winning strategy in the bijective k-pebble game. Duplicator responds to Spoiler's first move with the identity bijection. She maintains a board on the side which describes a position in the k cops and robber game played on the graph H. At any point, if Spoiler's pebbles are on the position $x_1, \dots, x_k$ in $\mathbf{A}$ and $v_1, \dots, v_k$ are the vertices of H to which these variables correspond, then the current position of the cops and robber game has k cops on the vertices $v_1, \dots, v_k$. If the robber's position according to its winning strategy is v, then Duplicator will play a bijection that is good for v.

To see that Duplicator can do this forever, suppose Spoiler lifts a pebble from x_i. Duplicator responds with a current bijection f that is good for v. Since the only equations not preserved by f are those associated with the vertex v, Spoiler must place at least three pebbles on variables associated with v to win the game. However, Duplicator responds to Spoiler placing the pebble on a new position x'_i by updating the position of the cops and robber game. Suppose robber's winning strategy dictates that the robber move from v to w. Since robber's move must be along a path avoiding the current cop positions, by Claim 3, Duplicator can update the bijection f to an f' that is good for w and agrees with f on all currently pebbled positions. It is now clear that Duplicator can play forever.

Theorem 4. *Let $\mathcal{G}$ be a non-trivial finite Abelian group. Then* $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ *is not definable in* $\mathrm{C}^{\omega}_{\infty\omega}$

Proof. Suppose, to the contrary, that there is a k such that $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ is definable in $\mathrm{C}^{k}_{\infty\omega}$. Let H be any connected, 3-regular graph with $\mathrm{tw}(H) > k$ and u any vertex of H. For instance, H could be a sufficiently large brick graph. Let a be any element of $\mathcal{G}$ distinct from 0. Then, by Lemma 2, $\mathbf{E}_0 H^u \in \mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ and $\mathbf{E}_a H^u \notin \mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$. But, by Lemma 3, $\mathbf{E}_0 H^u \equiv^{\mathrm{C}^k} \mathbf{E}_a H^u$, a contradiction.

4 Logical Reductions

4.1 Definition

Let σ and $\tau = (R_1, \ldots, R_s)$ be two relational vocabularies. A *k-ary interpretation with p parameters of τ in σ* is an $(s+1)$-tuple $\mathbf{I} = (\varphi_U, \varphi_1, \ldots, \varphi_s)$ of formulas over the vocabulary τ, where $\varphi_U = \varphi_U(\mathbf{x}, \mathbf{y})$ has $k+p$ free variables $\mathbf{x} = (x^1, \ldots, x^k)$ and $\mathbf{y} = (y_1, \ldots, y_p)$, and $\varphi_i = \varphi_i(\mathbf{x}_1, \ldots, \mathbf{x}_r, \mathbf{y})$ has kr free variables where r is the arity of R_i and each $\mathbf{x}_j = (x_j^1, \ldots, x_j^k)$ and $\mathbf{y} = (y_1, \ldots, y_p)$.

Let $\mathbf{A}$ be a σ-structure. A tuple $\mathbf{c} = (a_1, \ldots, a_p)$ of of pairwise different points of $\mathbf{A}$ is called *proper*. The *interpretation of* $\mathbf{A}$ *through* $\mathbf{I}$ *with parameters* $\mathbf{c}$, denoted by $\mathbf{I}(\mathbf{A}, \mathbf{c})$, is the τ-structure whose universe is $\{\mathbf{a} \in A^k : \mathbf{A} \models \varphi_U(\mathbf{a}, \mathbf{c})\}$, and whose interpretation for R_i is the set of tuples $(\mathbf{a}_1, \ldots, \mathbf{a}_r) \in (A^k)^r$ such that $\mathbf{A} \models \varphi_U(\mathbf{a}_1, \mathbf{c}) \wedge \cdots \wedge \varphi_U(\mathbf{a}_r, \mathbf{c}) \wedge \varphi_i(\mathbf{a}_1, \ldots, \mathbf{a}_r, \mathbf{c})$. If each formula in $\mathbf{I}$ belongs to a class of formulas Θ, we say that $\mathbf{I}$ is a Θ-interpretation.

Now we are ready to define the notion of logical reduction:

Definition 5. *Let $\mathcal{C}$ be a class of σ-structures, $\mathcal{D}$ a class of τ-structures closed under isomorphisms, and Θ be a class of formulas. A Θ-interpretation with p parameters $\mathbf{I}$ of τ in σ is a Θ-reduction from $\mathcal{C}$ to $\mathcal{D}$ if, for every σ-structure $\mathbf{A}$ with at least p elements, $\mathbf{A} \in \mathcal{C}$ if, and only if, $\mathbf{I}(\mathbf{A}, \mathbf{c}) \in \mathcal{D}$ for some proper $\mathbf{c}$.*

If such a reduction exists, we say that $\mathcal{C}$ reduces to $\mathcal{D}$ under Θ-reductions, and write $\mathcal{C} \leq_{\Theta} \mathcal{D}$. We use the collections of positive quantifier-free formulas, existential positive formulas, and datalog formulas (i.e. datalog programs) and write $\leq_{\mathrm{pqf}}$, $\leq_{\mathrm{ep}}$ and $\leq_{\mathrm{datalog}}$, respectively. These are reductions of increasing power, and definability in $\mathrm{C}^{\omega}_{\infty\omega}$ is preserved downwards by all three.

4.2 Expansions by reduced invariant relations

Let A be a set and let $R \subseteq A^s$ be a relation on A. We define an equivalence relation $\theta(R)$ on $\{1, \ldots, s\}$ by setting $(i, j) \in \theta(R)$ if, and only if, $a_i = a_j$ for every $(a_1, \ldots, a_s) \in R$. We say that R is *reduced* if $\theta(R)$ is the trivial equivalence relation (i.e. equality). Note that the equality relation on A is not reduced. The proof of the following lemma appears in the full version of this paper.

Lemma 6. *Let* $\mathbf{B}$ *be a finite structure, and* $\mathbf{D}$ *be an expansion of* $\mathbf{B}$ *by a reduced relation invariant under all polymorphisms of* $\mathbf{B}$*. Then,* $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B})$.

Next we show that reduced relations are general enough. First a piece of notation: Let $\mathbf{a} = (a_1, \ldots, a_m)$ be a sequence and let $I = (i_1, \ldots, i_r)$ be a sequence of indices, where $1 \leq i_j \leq m$ for every $j \in \{1, \ldots, r\}$. We write $\mathbf{a}_I$ for the sequence $(a_{i_1}, \ldots, a_{i_r})$. Now let R be a relation of arity s and I a sequence of indices from $\{1, \ldots, s\}$. Then $\mathrm{pr}_I R$ denotes the relation $\{\mathbf{a}_I : \mathbf{a} \in R\}$.

Let R be a relation of arity s and $\theta(R)$, the equivalence relation on $\{1, \ldots, s\}$ as defined in the previous section. Let I be a set of representatives of the equivalence-classes of $\theta(R)$, ordered in an arbitrary way, and define $\mathrm{red}(R) = \mathrm{pr}_I R$. Note that $\mathrm{red}(R)$ does not depend on the choice of I. Besides, for every $i \notin I$ there exists some $j \in I$ such that $a_i = a_j$ for every tuple $(a_1, \ldots, a_s) \in R$. We call $\mathrm{red}(R)$ the reduced version of R. A reduced structure is a structure all whose relations are reduced. To every structure $\mathbf{B}$ we can associate a reduced structure, called the *reduced version of* $\mathbf{B}$, whose universe is the universe of $\mathbf{B}$ itself and whose relations are the reduced versions of the relations of $\mathbf{B}$. Note that the vocabularies of a structure and its reduced version may be different. Note that the polymorphisms of $\mathbf{B}$ and its reduced version are the same.

Lemma 7. *Let* $\mathbf{B}$ *a finite structure and let* $\mathbf{D}$ *be the reduced version of* $\mathbf{B}$*. Then* $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{D})$ *and* $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B})$.

Proof: For space constraints, we only sketch the reduction $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{D})$. Let $\mathbf{A}$ be an instance of $\mathrm{CSP}(\mathbf{B})$. We define an instance $\mathbf{C}$ of $\mathrm{CSP}(\mathbf{D})$. The universe of $\mathbf{C}$ is A itself. For the relations, the basic idea is to project every relation $R^{\mathbf{A}}$ to the coordinates of a set of representatives I of the θ-classes, where $\theta = \theta(R)$. However, before we do that, we need to *close* each $R^{\mathbf{A}}$ under all equalities implied by the equivalences $(i, j) \in \theta$. We do that using Datalog-definable intermediate relations.

Let E be the binary relation on A defined by the following Datalog program:

$$\begin{aligned} E(x_i, x_j) &\ :-\ R(x_1, \ldots, x_s) \\ E(x, y) &\ :-\ E(y, x) \\ E(x, z) &\ :-\ E(x, y) \wedge E(y, z), \end{aligned}$$

where the first rule is introduced for every symbol R in σ and every $(i, j) \in \theta(R)$. It is obvious that E is an equivalence relation on A; reflexivity follows from the fact that $(i, j) \in \theta(R)$ in the first rule, symmetry is enforced by the second rule, and transitivity is enforced by the third. Next, for every r-ary symbol R in σ, let R' be the relation defined by $R'(\mathbf{x}_I) \ :- \ R(y_1,\dots,y_s) \wedge E(x_1,y_1) \wedge \dots \wedge E(x_s,y_s)$, where I is a set of representatives of the $\theta(R)$-classes ordered in an arbitrary way. This defines $\mathbf{C}$, and we defined it by a Datalog program interpreted on $\mathbf{A}$. It remains to argue that this datalog-interpretation is indeed a reduction. The argument may be found in the full version of this paper. □

4.3 Powering, subalgebras, and homomorphic images

In this subsection we show how the basic algebraic constructions of powering, subalgebra and homomorphic images can be handled by Datalog-reductions. In the following, fix a finite structure $\mathbf{B}$ and its corresponding algebra $\mathcal{B}$.

Suppose $\mathcal{B}'$ is an algebra that has a homomorphic image $\mathcal{A} = h(\mathcal{B}')$ that is a reduct of $\mathcal{B}$. Note that $A = B = h(B')$, i.e. the universes of $\mathcal{A}$ and $\mathcal{B}$ are the same and are the image of the universe of $\mathcal{B}'$ under h. We define a new structure $\mathbf{B}' = \mathrm{pre}(\mathbf{B}, h)$, the *preimage* of $\mathbf{B}$, whose universe is B' and whose relations are the preimages $h^{-1}(R^{\mathbf{B}})$ of the relations $R^{\mathbf{B}}$ of $\mathbf{B}$.

Lemma 8. *Let the algebras $\mathcal{B}$ and $\mathcal{B}'$, and the structures $\mathbf{B}$ and $\mathbf{B}' = \mathrm{pre}(\mathbf{B}, h)$ be as above. Then $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$ and $\mathcal{B}'$ is a reduct of $\mathrm{Al}(\mathbf{B}')$.*

Suppose $\mathcal{B}'$ is an algebra that has a subalgebra $\mathcal{A} \subseteq \mathcal{B}'$ that is a reduct of $\mathcal{B}$. Note that $A = B \subseteq B'$, i.e. the universes of $\mathcal{A}$ and $\mathcal{B}$ are the same and are a subset of the universe of $\mathcal{B}'$. We define a new structure $\mathbf{B}' = \mathrm{ext}(\mathbf{B}, B')$, the *extension* of $\mathbf{B}$, with universe B' and the same relations as $\mathbf{B}$.

Lemma 9. *Let the algebras $\mathcal{B}$ and $\mathcal{B}'$, and the structures $\mathbf{B}$ and $\mathbf{B}' = \mathrm{ext}(\mathbf{B}, B')$ be as above. Then $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$ and $\mathcal{B}'$ is a reduct of $\mathrm{Al}(\mathbf{B}')$.*

Let R be an r-ary relation on the set A^n. Then the *flattening* of R, denoted $\mathrm{fla}(R, n)$, is the rn-ary relation on A that contains all tuples $(x_1, \dots, x_{rn})$ such that $((x_1, \dots, x_n), \dots, (x_{(r-1)n+1}, \dots, x_{rn})) \in R$. Suppose $\mathcal{B}'$ is an algebra that has a direct power $\mathcal{A} = \mathcal{B}'^n$ that is a reduct of $\mathcal{B}$. Note that $A = B = B'^n$, i.e. the universes of $\mathcal{A}$ and $\mathcal{B}$ are the same and are the n-th power of the universe of $\mathcal{B}'$. We define a new structure $\mathbf{B}' = \mathrm{fla}(\mathbf{B}, n)$, the *flattening* of $\mathbf{B}$, whose universe is B and whose relations are the flattenings of the relations of $\mathbf{B}$.

Lemma 10. *Let the algebras $\mathcal{B}$ and $\mathcal{B}'$, and the structures $\mathbf{B}$ and $\mathbf{B}' = \mathrm{fla}(\mathbf{B}, n)$ be as above. Then $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$ and $\mathcal{B}'$ is a reduct of $\mathrm{Al}(\mathbf{B}')$.*

These lemmas allow us to derive the following consequence we require.

Theorem 11. *Let $\mathbf{B}$ and $\mathbf{B}'$ be finite structures and let $\mathcal{B}$ and $\mathcal{B}'$ be their respective algebras. If $\mathrm{var}(\mathcal{B}')$ contains a reduct of $\mathcal{B}$, then $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{B}')$.*

Proof: Suppose that some algebra $\mathcal{A}$ of $\mathrm{var}(\mathcal{B}')$ is a reduct of $\mathcal{B}$. By the HSP-theorem [19, Theorem 9.5] $\mathcal{A}$ is a homomorphic image of a subalgebra of a direct power of $\mathcal{B}'$. Let $\mathcal{B}_p$, $\mathcal{B}_s$, and $\mathcal{B}_h$ be the direct power, its subalgebra, and the homomorphic image, respectively. We have $\mathcal{A} = \mathcal{B}_h$. Let n be such that $\mathcal{B}_p = \mathcal{B}'^n$, an let h be a homomorphism from $\mathcal{B}_s$ to $\mathcal{B}_h$.

We use three intermediate structures $\mathbf{B}_s = \mathrm{pre}(\mathbf{B}, h)$, $\mathbf{B}_p = \mathrm{ext}(\mathbf{B}_s, B_p)$, and $\mathbf{B}_f = \mathrm{fla}(\mathbf{B}_p, n)$ that, by the definition, have the universes of the algebras $\mathcal{B}_s$, $\mathcal{B}_p$, and $\mathcal{B}'$ respectively. By Lemma 8, $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}_s)$ and $\mathcal{B}_s$ is a reduct of $\mathrm{Al}(\mathbf{B}_s)$. By Lemma 9, $\mathrm{CSP}(\mathbf{B}_s) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}_p)$ and $\mathcal{B}_p$ is a reduct of $\mathrm{Al}(\mathbf{B}_p)$. By Lemma 10, $\mathrm{CSP}(\mathbf{B}_p) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}_f)$, and $\mathcal{B}'$ is a reduct of $\mathrm{Al}(\mathbf{B}_f)$. Now, let $\mathbf{D}$ be the reduced version of $\mathbf{B}_f$. Then $\mathrm{CSP}(\mathbf{B}_f) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{D})$ by Lemma 7. We prove that $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$ and the result will follow by composing.

Let $\mathbf{D}'$ be the expansion of $\mathbf{D}$ obtained by adding all the relations of $\mathbf{B}'$. Since $\mathcal{B}'$ is a reduct of the algebra of $\mathbf{B}_f$ and $\mathbf{D}'$ is the flattening of $\mathbf{B}_f$, it is straightforward that every relation of $\mathbf{D}'$ is invariant under all polymorphisms of $\mathbf{B}'$. Moreover, the relations in $\mathbf{D}'$ that are not in $\mathbf{B}'$ are reduced, so $\mathrm{CSP}(\mathbf{D}') \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$ by Lemma 6. It is also obvious that $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{D}')$ through the mapping that sends a structure to its expansion with empty relations. Composing we get $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{B}')$. □

4.4 Reduction from the idempotent case

To every finite structure $\mathbf{B}$ we associate a new structure, the *singleton-expansion* of $\mathbf{B}$, by adding one unary relation $\{b\}$ for every $b \in B$. In other words, if $B = \{b_1, \ldots, b_n\}$, then the structure $(\mathbf{B}, \{b_1\}, \ldots, \{b_n\})$ is the singleton-expansion of $\mathbf{B}$. Note that the polymorphisms of the singleton-expansion of $\mathbf{B}$ are exactly the *idempotent* polymorphisms of $\mathbf{B}$, that is polymorphisms f satisfying the identity $f(x, \ldots, x) = x$. Indeed, every singleton set $\{b\}$ is preserved by any idempotent polymorphism of $\mathbf{B}$, and any polymorphism of $\mathbf{B}$ that preserves every singleton set $\{b\}$ must by idempotent.

Lemma 12. *Let* $\mathbf{B}$ *be a finite structure, and let* $\mathbf{D}$ *be the singleton-expansion of* $\mathbf{B}$. *Then* $\mathrm{CSP}(\mathbf{B}) \leq_{\mathrm{pqf}} \mathrm{CSP}(\mathbf{D})$ *and if* $\mathbf{B}$ *is a core with at least two points, then* $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{ep}} \mathrm{CSP}(\mathbf{B})$.

5 Omitting types and Results

Let $\mathcal{A}$ be an algebra. A *congruence* of $\mathcal{A}$ is an equivalence relation α that is invariant with respect to all operations of $\mathcal{A}$. In other words, for any (n-ary) operation f of $\mathcal{A}$ and any $a_1, \ldots, a_n, b_1, \ldots, b_n \in \mathcal{A}$ such that $(a_i, b_i) \in \alpha$ we have $(f(a_1, \ldots, a_n), f(b_1, \ldots, b_n)) \in \alpha$. The congruences of $\mathcal{A}$ form its *congruence lattice* $\mathrm{con}(\mathcal{A})$. A *prime quotient* in this lattice is a pair of congruences α, β such that $\alpha \leq \beta$, $\alpha \neq \beta$, and for any γ with $\alpha \leq \gamma \leq \beta$ we have either $\alpha = \gamma$, or $\beta = \gamma$. The fact that α, β is a prime quotient will be denoted by $\alpha \prec \beta$.

Tame congruence theory [10] allows one to assign to each prime quotient of the congruence lattice $\mathrm{con}(\mathcal{A})$ of a finite algebra $\mathcal{A}$ one of five types. The type reflects the local structure of the algebra, which can be one of the following: **1** a finite set with a group action on it (unary type), **2** a finite vector space over a finite field (affine type), **3** a two-element Boolean algebra, **4** a two-element lattice, **5** a two-element semilattice. We use tame congruence as a black box extracting properties we need from existing results, and we do not therefore need a precise definition of the types.

The type of a prime quotient $\alpha \prec \beta$ is denoted by $\mathsf{typ}(\alpha, \beta)$, while $\mathsf{typ}(\mathcal{A})$ denotes the set of types appearing as types of some prime quotient of $\mathcal{A}$. If $\mathfrak{A}$ is a class of algebras, $\mathsf{typ}(\mathfrak{A})$ denotes the set $\bigcup_{\mathcal{A} \in \mathfrak{A}} \mathsf{typ}(\mathcal{A})$. If $\mathbf{i} \notin \mathsf{typ}(\mathfrak{A})$, we say that $\mathfrak{A}$ *omits* type $\mathbf{i}$. Otherwise, we say $\mathfrak{A}$ *admits* type $\mathbf{i}$. We need the following:

Lemma 13. *Let $\mathcal{A}$ be a finite idempotent algebra. If* $\mathrm{var}(\mathcal{A})$ *admits types* **1** *or* **2** *then it contains a finite idempotent reduct of a module.*

Recall from Section 3 the definition of the structure $\mathbf{E}_{\mathcal{G},r}$ for every finite Abelian group $\mathcal{G}$ and every integer $r \geq 1$.

Lemma 14. *Let $\mathcal{M}$ be a finite module, let $\mathcal{G}$ be its underlying Abelian group, and let $\mathcal{A}$ be an idempotent reduct of $\mathcal{M}$. Then $\mathcal{A}$ is a reduct of the algebra of $\mathbf{E}_{\mathcal{G},r}$ for every $r \geq 1$.*

Bringing together these results with those of Section 4 we get:

Theorem 15. *Let* $\mathbf{B}$ *be a finite structure and let $\mathcal{B}$ be its algebra. If* $\mathrm{var}(\mathcal{B})$ *admits the unary or affine types, then there exists a non-trivial finite Abelian group $\mathcal{G}$ such that* $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},r}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{B})$ *for every $r \geq 1$.*

Proof. Since $\mathrm{CSP}(\mathbf{B}) = \mathrm{CSP}(\mathrm{core}(\mathbf{B}))$, where $\mathrm{core}(\mathbf{B})$ is the core of $\mathbf{B}$, we may assume that $\mathbf{B}$ is a core. Let $\mathbf{D}$ be the singleton-expansion of $\mathbf{B}$ and let $\mathcal{D}$ be its algebra, which is idempotent. By Lemma 12, we have $\mathrm{CSP}(\mathbf{D}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{B})$. Moreover, if $\mathrm{var}(\mathcal{B})$ admits types **1** or **2**, so does $\mathrm{var}(\mathcal{D})$ because $\mathcal{D}$ is a reduct of $\mathcal{B}$ (see [10, Chapter 5]). By Lemma 13, the variety $\mathrm{var}(\mathcal{D})$ contains a finite idempotent reduct $\mathcal{A}$ of a module. Let $\mathcal{G}$ be the Abelian group underlying the module. Then $\mathcal{G}$ is non-trivial and finite. Moreover, $\mathcal{A}$ is a reduct of the algebra of $\mathbf{E}_{\mathcal{G},r}$ for every $r \geq 1$ by Lemma 14. It follows that $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},r}) \leq_{\mathrm{datalog}} \mathrm{CSP}(\mathbf{D})$. Composing we get the result. □

We have seen in Section 3 that $\mathrm{CSP}(\mathbf{E}_{\mathcal{G},3})$ is not definable in $\mathrm{C}^{\omega}_{\infty\omega}$ when $\mathcal{G}$ is non-trivial. Since definability in $\mathrm{C}^{\omega}_{\infty\omega}$ is preserved downwards by Datalog-reductions, this yields the following corollary:

Corollary 16. *Let* $\mathbf{B}$ *be a finite structure and let $\mathcal{B}$ be its algebra. If* $\mathrm{CSP}(\mathbf{B})$ *is definable in $\mathrm{C}^{\omega}_{\infty\omega}$, then* $\mathrm{var}(\mathcal{B})$ *omits the unary and affine types.*

Corollary 16 can be seen as a strengthening of the result of Larose and Zádori [11] that if the complement of $\mathrm{CSP}(\mathbf{B})$ is definable in Datalog then $\mathrm{var}(\mathcal{B})$ omits the unary and affine types. Larose and Zádori also conjectured the converse, namely that if $\mathrm{var}(\mathcal{B})$ omits the unary and affine types then the complement of $\mathrm{CSP}(\mathbf{B})$ is definable in Datalog. By Corollary 16 this conjecture would imply that every $\mathrm{CSP}(\mathbf{B})$ is either definable in Datalog or not definable in $\mathrm{C}^{\omega}_{\infty\omega}$, which can be seen as a definability dichotomy.

Finally, we consider three decision problems. In Algebra of type 2 we are given a finite set A and operation tables of idempotent operations $f_1, \ldots, f_n$ on A, and the question is whether $\mathrm{var}(\mathcal{A})$, where $\mathcal{A} = (A; \{f_1, \ldots, f_n\})$, omits types **1** and **2**. In Relational Structure of type 2 we are given a finite relational structure $\mathbf{A}$, and the question is whether $\mathrm{var}(\mathrm{Al}(\mathbf{A}))$ omits types **1** and **2**. In Relational Structure of type $2(k)$ we are given a finite relational structure $\mathbf{A}$, $|A| \leq k$, and the question is whether $\mathrm{var}(\mathrm{Al}(\mathbf{A}))$ omits types **1** and **2**. The problems Algebra of type 2 and Relational Structure of type $2(k)$ were shown tractable in [12].

Theorem 17. Relational Structure of type 2 *is NP-complete.*

References

1. Feder, T., Vardi, M.: Computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory. SIAM Journal of Computing **28** (1998) 57–104
2. Dalmau, V.: Linear datalog and bounded path duality of relational structures. Logical Methods in Computer Science **1**(1) (2005)
3. Larose, B., Loten, C., Tardif, C.: A characterisation of first-order constraint satisfaction problems. In: LICS. (2006) 201–210
4. Bulatov, A.: Mal'tsev constraints are tractable. Technical Report PRG-RR-02-05, Computing Laboratory, University of Oxford, Oxford, UK (2002)
5. Bulatov, A.A., Dalmau, V.: A simple algorithm for mal'tsev constraints. SIAM J. Comput. **36**(1) (2006) 16–27
6. Dalmau, V., Kolaitis, P., Vardi, M.: Constraint satisfaction, bounded treewidth, and finite variable logics. In: Proceedings of the 8th International Conference on Principles and Practice of Constraint Programming, CP'02. Lecture Notes in Computer Science, Springer-Verlag (2002) 311–326
7. Kolaitis, P., Vardi, M.: A game-theoretic approach to constraint satisfaction. In: Proc. 17th National Conference on Artificial Intelligence, AAAI-2000. (2000) 175–181
8. Blass, A., Gurevich, Y., Shelah, S.: On polynomial time computation over unordered structures. J. Symbolic Logic **67**(3) (2002) 1093–1125
9. Bulatov, A., Jeavons, P., Krokhin, A.: Classifying the complexity of constraints using finite algebras. SIAM Journal on Computing **34** (2005) 720–742
10. Hobby, D., McKenzie, R.: The Structure of Finite Algebras. Volume 76 of Contemporary Mathematics. American Mathematical Society, Providence, R.I. (1988)
11. Larose, B., Zádori, L.: Bounded width problems and algebras. (Algebra Universalis, to appear)
12. Larose, B., Valeriote, M. personal communication (2006)
13. Diestel, R.: Graph Theory. Springer (1997)
14. Seymour, P., R.Thomas: Graph searching and a min-max theorem for treewidth. Journal of Combinatorial Theory, Series B **58** (1993) 22–33
15. Libkin, L.: Elements of Finite Model Theory. Springer (2004)
16. Ebbinghaus, H.D., Flum, J.: Finite Model Theory. 2 edn. Springer (1999)
17. Immerman, N.: Descriptive Complexity. Springer (1999)
18. Hella, L.: Logical hierarchies in PTIME. Info. and Comput. **129** (1996) 1–19
19. Burris, S., Sankappanavar, H.: A course in universal algebra. Volume 78 of Graduate Texts in Mathematics. Springer-Verlag, New York-Berlin (1981)