Electronic Colloquium on Computational Complexity, Report No. 24 (2005)

Quantified Constraints: The Complexity of Decision and Counting for Bounded Alternation

Michael Bauland[1], Elmar Böhler[2], Nadia Creignou[3], Steffen Reith[4], Henning Schnoor[1], and Heribert Vollmer[1]

[1] Theoretische Informatik, Universität Hannover, Appelstr. 4, 30167 Hannover, Germany. bauland|schnoor|vollmer@thi.uni-hannover.de

[2] Theoretische Informatik, Universität Würzburg, Am Hubland, 97072 Würzburg, Germany. boehler@informatik.uni-wuerzburg.de

[3] LIF (UMR 6616), Univ. Méditerranée, Marseille, France. creignou@lif.univ-mrs.fr

[4] Lengfelderstr. 35b, 97078 Würzburg, Germany. streit@streit.cc

Abstract. We consider constraint satisfaction problems parameterized by the set of allowed constraint predicates. We examine the complexity of quantified constraint satisfaction problems with a bounded number of quantifier alternations and the complexity of the associated counting problems. We obtain classification results that completely solve the Boolean case, and we show that hardness results carry over to the case of arbitrary finite domains.

1 Introduction

Constraint satisfaction is recognized as a fundamental problem in computer science, since combinatorial problems from many different application areas (artificial intelligence, databases, automated design, etc.) can be expressed in a natural way by means of constraints. Informally, an instance of a constraint satisfaction problem consists of a set of variables, a set of possible values for the variables, and a set of constraints that restrict the combinations of values that certain tuples of variables may take; the question is whether there is an assignment of values to variables that satisfies the given constraints. Usually constraints are specified by means of relations. The standard constraint satisfaction problem can therefore be parameterized by restricting the set of relations S, thus defining the so called *non-uniform constraint satisfaction problem* $\mathsf{CSP}(S)$. The problem of classifying the complexity of $\mathsf{CSP}(S)$ (and its many variants) as a function of the set S has attracted much attention, not only because constraint satisfaction problems play an important role in application areas as mentioned above, but also because these problems form the "nucleus" of many complexity classes. Therefore, "by focusing on this restricted world one can present a reasonably accurate bird's eye view of complexity theory" [CKS01] and hope to contribute to the study of complexity classes.

The Boolean case, i.e., the set S of constraint relations consists only of Boolean relations, was first investigated by Thomas Schaefer [Sch78]. Schaefer showed that here, every $\mathsf{CSP}(S)$ is either NP-complete or solvable in polynomial time, hence avoiding the infinitely many complexity degrees that exist (under the assumption $\mathrm{P} \neq \mathrm{NP}$) in between (see [Lad75]). Since then, there has been a growing body of classification results for related problems such as counting problems or optimization problems for Boolean CSPs (see the monograph [CKS01] or the recent surveys [BCRV03,BCRV04]). The simple change of allowing non-Boolean variables seems to increase the expressive power of constraint satisfaction problems considerably. Feder and Vardi [FV98] conjectured that the dichotomy exhibited in the Boolean case continues to hold in the non-Boolean case. Their conjecture, however, remains unresolved to this date. The most successful approach so far has been the algebraic approach developed in [JCG97,Jea98,BJK00], cf. also [BJK04] for a survey. It has led to many wonderful results, in particular a complete classification of CSP over the three-element domain [Bul02].

ISSN 1433-8092

Recently, *quantified constraint satisfaction problems* have raised a lot of attention. Quantified constraint satisfaction problems are in PSPACE, and the Boolean quantified constraint satisfaction problem QSAT is the prototypical PSPACE-complete problem (see [Pap94]). A dichotomy theorem for the complexity of Boolean constraint satisfaction problems $\mathsf{QCSP}(S)$ was established in [Sch78,Dal97,CKS01]. More recently researchers have embarked on an investigation of the complexity of $\mathsf{QCSP}(S)$ over an arbitrary finite domain. Specifically, Börner *et al.* [BKBJ02,BBJK03] have extended the algebraic approach to the more general framework of quantified constraint satisfaction problems. In this way they found sufficient conditions for tractability of $\mathsf{QCSP}(S)$ and obtained a trichotomy result for those sets S that include all graphs of permutations. Chen [Che04] identified further large classes of tractable quantified constraint problems.

Very recently Edith Hemaspaandra considered Boolean quantified constraint satisfaction problems in which the number of quantifier alternations is bounded, denoted by $\mathsf{QCSP}_i(S)$ (for $i-1$ alternations). These problems are prototypical for the polynomial hierarchy [MS72]; in particular, QSAT_i (i.e., QCSP_i with no restriction on the set of allowed predicates) is complete for the class Σ_iP. Hemaspaandra obtained a dichotomy result for Boolean $\mathsf{QCSP}_i(S)$, identifying conditions for S that make $\mathsf{QCSP}_i(S)$ complete for Σ_iP, and showing that the problem is tractable otherwise. The proof given in [Hem04] uses implementations in the style of Schaefer [Sch78] and the monograph [CKS01] and makes no reference to the algebraic framework at all. Feder and Kolaitis [FK05] considered quantified constraints with bounded number of quantifier alternations over arbitrary finite domains. They exhibited interesting connections to finite model theory and managed to obtain a dichotomy theorem for a variant of $\mathsf{QCSP}_i(S)$ where the application of universal quantifiers as well as the allowed predicates in S are restricted in a certain way.

The contributions of our paper are threefold:

1. Making use of the algebraic approach and the structure of Post's lattice of Boolean clones we obtain a very short re-proof of Hemaspaandra's dichotomy.

2. We generalize the hardness part of our algebraic proof for the dichotomy for Boolean $\mathsf{QCSP}_i(S)$ to arbitrary finite domains. In this way we show, e.g., that $\mathsf{QCSP}_i(S)$ is complete (under logspace many-one reductions) for Σ_iP if all closure properties of relations in S are essentially unary or constant.

3. We generalize the complexity results we obtained for $\mathsf{QCSP}_i(S)$ from decision problems to counting problems, i.e., we consider the problem $\#\mathsf{QCSP}_i(S)$, given a quantified constraint satisfaction instance with at most $i-1$ alternations of quantifiers, to determine the number of satisfying solutions it has. We introduce a new type of reductions that we call *permutative reductions.* These are a generalization of subtractive reductions introduced in [DHK00]. In the same way as subtractive reductions, our reductions have the advantage that they are strict enough to close most relevant counting classes and are wide enough to obtain hardness for a number of problems. We believe that permutative reductions will turn out useful in other contexts, in particular for problems with symmetry properties. Using these reductions, we first obtain a complete classification of the complexity of $\#\mathsf{QCSP}_i(S)$ for Boolean S: We show that these problems are either (a) solvable in polynomial time or (b) complete for Valiant's class #P or (c) complete for the class $\#\cdot\Sigma_i$P, the counting analogue of Σ_iP, and we obtain easy criteria to determine which case holds. Then we turn to the case of arbitrary finite domains and show that $\#\mathsf{QCSP}_i(S)$ is complete for $\#\cdot\Sigma_i$P if all closure properties of S are constant or essentially unary.

The organization of the paper is as follows. In Sections 2–4 we introduce the reader to constraint satisfaction problems, the algebraic framework, and quantified constraints, resp. In Sect. 5 we examine the complexity of the problems $\mathsf{QCSP}_i(S)$. This section contains our complete re-proof of Hemaspaandra's dichotomy as well as more general hardness results. In Sect. 6 we turn to the problems $\#\mathsf{QCSP}_i(S)$. We first give a general introduction to counting problems and the reductions that are useful in this context. Here, we also introduce permutative reductions. Then we obtain our trichotomy for Boolean $\#\mathsf{QCSP}_i(S)$. The maybe technically most involved proof in our paper then exhibits hardness results for $\#\mathsf{QCSP}_i(S)$ for non-Boolean relations. We conclude our paper with a summary and a prospect for further research.

2 Constraint Satisfaction Problems

Throughout the paper we use the standard correspondence between predicates and relations. We use the same symbol for a predicate and its corresponding relation, the meaning will always be clear from the context. We say that the predicate *represents* the relation. The set $\mathcal{D}$ will represent a finite domain of cardinality $m \geq 2$, $\mathcal{D} = \{0, \dots, m-1\}$. An n-ary *logical relation* R is a relation of arity n defined over $\mathcal{D}$. Let V be a set of variables. A *constraint* is an application of R to an n-tuple of variables from V, i.e., $R(x_1, \dots, x_n)$. An assignment of values to the variables $I\colon V \to \mathcal{D}$ satisfies the constraint $R(x_1, \dots, x_n)$ if $(I(x_1), \dots, I(x_n)) \in R$ holds.

Example 2.1.
- Equivalence, $=^{\mathcal{D}}$, is the binary relation defined by $\{(0,0), \dots (m-1, m-1)\}$. Similarly the disequality, $\neq^{\mathcal{D}}$, is defined by $\mathcal{D}^2 \setminus =^{\mathcal{D}}$.
- Given the ternary relation $\mathrm{NAE}^{\mathcal{D}} = \mathcal{D}^3 \setminus \{(0,0,0), \dots, (m-1, m-1, m-1)\}$, the constraint $\mathrm{NAE}^{\mathcal{D}}(x_1, x_2, x_3)$ is satisfied if and only if not all variables are assigned the same value. We write NAE^m for $\mathrm{NAE}^{\mathcal{D}}$ with $|\mathcal{D}| = m$.
- The Boolean constraint $\mathrm{R}_{n/m}(x_1, \dots, x_m)$ is satisfied if exactly n of the m variables are assigned to 1.

Let S be a non-empty finite set of relations defined over $\mathcal{D}$. An S*-formula* is a finite conjunction of S*-clauses*, $\varphi = c_1 \wedge \cdots \wedge c_k$, where each S-clause c_i is a constraint application of some logical relation $R \in S$. An assignment I satisfies φ if it satisfies all clauses c_i. We denote by $\mathsf{sat}(\varphi)$ the set of satisfying assignments of a formula φ. We denote by $\mathsf{CSP}(S)$ the satisfiability problem for S-formulas. For a relation R, we often write $\mathsf{CSP}(R)$ instead of $\mathsf{CSP}(\{R\})$.

Example 2.2.
- The well-known 3-SAT problem can be seen as the CSP problem over the set $\mathrm{S}_3 := \{(x_1 \vee x_2 \vee x_3), (\overline{x_1} \vee x_2 \vee x_3), (\overline{x_1} \vee \overline{x_2} \vee x_3), (\overline{x_1} \vee \overline{x_2} \vee \overline{x_3})\}$.
- The 3-Colorability problem can be seen as the CSP problem using only the disequality relation over the three-element domain.

Given a set S of relations, in order to study the complexity of $\mathsf{CSP}(S)$ we will be interested in the expressive power of S, which can be measured by the set $\mathsf{COQ}(S)$ of all relations that can be represented by formulas of the form

$$F(x_1, \dots, x_k) \quad = \quad \exists y_1 \exists y_2 \cdots \exists y_l \, \varphi(x_1, \dots, x_k, y_1, \dots, y_l),$$

where φ is an S-formula. Such formulas are also called *conjunctive-queries.*

3 Closure Properties

Throughout the text we refer to different types of Boolean constraint relations following Schaefer's terminology [Sch78]. We say that a Boolean relation R is *1-valid* if $(1, \dots, 1) \in R$ and it is *0-valid* if $(0, \dots, 0) \in R$, *(dual) Horn* if R can be represented by a conjunctive normal form (CNF) formula having at most one unnegated (negated) variable in each clause, *bijunctive* if it can be represented by a CNF formula having at most two variables in each clause, *affine* if it can be represented by a conjunction of linear functions, i.e., a CNF formula with $\oplus$-clauses (XOR-CNF), *complementive* if for each $(\alpha_1, \dots, \alpha_n) \in R$, also $(\neg\alpha_1, \dots, \neg\alpha_n) \in R$.

A set S of Boolean relations is called 0-valid (1-valid, Horn, dual Horn, affine, bijunctive, complementive) if every relation in S has this property. Finally a set S of Boolean relations is called *Schaefer* if it is either Horn, dual Horn, affine, or bijunctive.

Given a Boolean relation R the following well-known closure properties determine the structure of R (operations are applied coordinate-wise on vectors, maj is the ternary majority function, which yields 1 if and only if at least two of its arguments are 1).

- R is Horn if and only if $m, m' \in R$ implies $m \wedge m' \in R$.
- R is dual Horn if and only if $m, m' \in R$ implies $m \vee m' \in R$.
- R is bijunctive if and only if $m, m', m'' \in R$ implies $\mathrm{maj}(m, m', m'') \in R$.
- R is affine if and only if $m, m', m'' \in R$ implies $m \oplus m' \oplus m'' \in R$.

The notion of closure property of a relation has been defined more generally, see for instance [JCG97,Pip97]. Let $f\colon \mathcal{D}^k \to \mathcal{D}$ be a k-ary function. We say that R is *closed under* f, or that f is a *polymorphism* of R, if for any choice of k vectors $m_1, \ldots, m_k \in R$, not necessarily distinct, we have that

$$\Big(f\big(m_1[1], \ldots, m_k[1]\big),\ f\big(m_1[2], \ldots, m_k[2]\big),\ \ldots,\ f\big(m_1[n], \ldots, m_k[n]\big)\Big) \in R,$$

i.e., the vector constructed coordinate-wise from m_1, …, m_k by means of f belongs to R.

We denote by $\mathrm{Pol}(R)$ the set of all polymorphisms of R and by $\mathrm{Pol}(S)$ the set of functions that are polymorphisms of every relation in S. It turns out that $\mathrm{Pol}(S)$ is a *clone* for every set of relations S, i.e., $\mathrm{Pol}(S)$ contains all projection functions and is closed under superposition (composition of functions), see e.g. [Pip97].

An interesting Galois correspondence exists between the sets of functions $\mathrm{Pol}(S)$ and the sets of relations S. An introduction to this correspondence can be found in [Pip97,Pös01] and a comprehensive study in [PK79]. This theory helps us to get elegant and short proofs for complexity results concerning constraint satisfaction problems, see e.g. [JCG97], [BCRV04]. Indeed, it shows that the smaller the set of polymorphisms is, the more expressive the corresponding conjunctive queries are, which is the cornerstone for applying the algebraic method to complexity. The following proposition can be found, e.g., in [Dal00].

Proposition 3.1. *Let S_1 and S_2 be sets of relations defined over $\mathcal{D}$. If the inclusion* $\mathrm{Pol}(S_2) \subseteq \mathrm{Pol}(S_1)$ *holds, then* $\mathsf{COQ}(S_1) \subseteq \mathsf{COQ}(S_2 \cup \{=^{\mathcal{D}}\})$.

This result was used in [JCG97] to obtain the following complexity result.

Theorem 3.2. *Let S_1 and S_2 be sets of relations defined over $\mathcal{D}$ such that S_1 is finite. If the inclusion* $\mathrm{Pol}(S_2) \subseteq \mathrm{Pol}(S_1)$ *holds, then* $\mathsf{CSP}(S_1)$ *is polynomial-time reducible to* $\mathsf{CSP}(S_2)$.

A number of results on the complexity of CSP have been obtained via this approach (see e.g., [JCG97,Bul02]). In particular, the well-known Schaefer's dichotomy theorem can be proved in this way by using *Post's lattice* (see e.g., [BCRV04]).

Theorem 3.3. [Sch78] *Let S be a finite set of Boolean relations. If S is Schaefer, or* 0- *or* 1-*valid, then* $\mathsf{CSP}(S)$ *is in* P, *otherwise* $\mathsf{CSP}(S)$ *is* NP-*complete.*

4 Quantified Problems

In this paper we consider the more general framework of quantified constraint satisfaction problems, which are defined as follows.

Let S be a finite set of relations defined over the domain $\mathcal{D}$. An instance of $\mathsf{QCSP}(S)$ is a closed formula of the form $Q_1x_1Q_2x_2 \ldots Q_nx_n\phi$, where $Q_1, \ldots, Q_n$ are arbitrary quantifiers and ϕ is an S-formula. The question is whether the sentence is true.

One can use an exhaustive algorithm to show that $\mathsf{QCSP}(S)$ is always in PSPACE. The problem of deciding, whether a given closed quantified Boolean formula is true, is PSPACE-complete [SM73]. This problem remains PSPACE-complete if we restrict the formulas to 3-CNF [Sto77]. It is worth noticing that the Boolean case still displays a dichotomy for quantified satisfiability.

Theorem 4.1. [Sch78,Dal97,CKS01] *Let S be a finite set of Boolean relations. If S is Schaefer, then* $\mathsf{QCSP}(S)$ *is in* P*, otherwise* $\mathsf{QCSP}(S)$ *is* PSPACE*-complete.*

In this paper, we are interested in quantified constraint satisfaction problems in which the number of quantifier alternations is bounded. These problems are prototypical for the *polynomial-time hierarchy* (PH for short), which was defined by Meyer and Stockmeyer [MS72]. Following the notation of [Pap94], $\Sigma_0\mathrm{P} = \Pi_0\mathrm{P} = \mathrm{P}$ and for all $i \geq 0$, $\Sigma_{i+1}\mathrm{P} = \mathrm{NP}^{\Sigma_i\mathrm{P}}$ and $\Pi_{i+1}\mathrm{P} = \mathrm{coNP}^{\Sigma_i\mathrm{P}}$. The set QSAT_i is the set of all closed, true Boolean formulas with $i-1$ quantifier alternations, starting with an $\exists$-quantifier. For all $i \geq 1$, QSAT_i is complete for $\Sigma_i\mathrm{P}$. This problem remains $\Sigma_i\mathrm{P}$-complete if we restrict the Boolean formula to be 3-CNF for i odd, and 3-DNF for i even [Wra77]. To generalize QSAT_i to arbitrary sets of constraints S, and to arbitrary finite domains, we adopt the following definition for $\mathsf{QCSP}_i(S)$ from [Hem04]:

Let S be set a of relations over the domain $\mathcal{D}$ and let $i \geq 1$.

- A $\Sigma_i(S)$-formula is a closed formula of the form $\Phi = \exists X_1 \forall X_2 \ldots Q_i X_i \psi$,
- a $\Pi_i(S)$-formula is a closed formula of the form $\Phi = \forall X_1 \exists X_2 \ldots Q_i X_i \psi$,

where the X_j, $j = 1, \ldots, i$, are disjoint sets of variables and ψ is a quantifier-free S-formula defined on $\bigcup_j X_j$, and is called the matrix of Φ. For i odd, a $\mathsf{QCSP}_i(S)$-formula is a $\Sigma_i(S)$-formula, for i even, a $\mathsf{QCSP}_i(S)$-formula is a $\Pi_i(S)$-formula. For i odd (even), $\mathsf{QCSP}_i(S)$ is the problem of deciding whether a given $\mathrm{QCSP}_i(S)$-formula is true (false).

Note that $\mathsf{QCSP}_i(S)$ belongs to $\Sigma_i\mathrm{P}$ for each $i \geq 1$. Moreover, according to Wrathall's result [Wra77], $\mathsf{QCSP}_i(\mathrm{S}_3)$ (see Example 2.2) is $\Sigma_i\mathrm{P}$-complete.

The following proposition states that the Galois connection between sets of relations and their closure properties still applies to quantified problems with bounded alternations.

Proposition 4.2. *Let S_1 and S_2 be two sets of relations over the same finite domain $\mathcal{D}$ such that S_1 is finite, and let $i \geq 1$. If the inclusion* $\mathrm{Pol}(S_2) \subseteq \mathrm{Pol}(S_1)$ *holds, then* $\mathsf{QCSP}_i(S_1)$ *is logspace many-one reducible to* $\mathsf{QCSP}_i(S_2)$.

Proof. If $\mathrm{Pol}(S_2) \subseteq \mathrm{Pol}(S_1)$, then due to Proposition 3.1 one can express every relation from S_1 by an existential $S_2 \cup \{=^{\mathcal{D}}\}$-formula. We locally replace every S_1-constraint by its equivalent $S_2 \cup \{=^{\mathcal{D}}\}$-formula and move the additional existential variables to the right end of the quantifier sequence. Since in every $\mathsf{QCSP}_i(S)$-formula the last quantifier is $\exists$, we end with a $\mathsf{QCSP}_i(S_2 \cup \{=^{\mathcal{D}}\})$-formula equivalent to the original formula.

We now remove the equality constraints. We check if there are variables x and y such that y is $\forall$-quantified after x is quantified with an $=$-path from x to y. In this case, the formula is false. Otherwise, all $=$-cliques of variables consist of variables of which at most the first one, x is universally quantified. We can rename all these variables to x and delete the existential quantifiers. The complexity of this procedure is dominated by undirected graph accessibility, which is in logspace due to [Rei04].

Contrary to [BBJK03, Theorem 4], we cannot restrict our attention to surjective polymorphisms, because in the context of bounded quantifier alternation, this does not yield sharp reductions.

5 Complexity Results in the Polynomial Hierarchy

Let us start with a completeness result in the Boolean domain.

Lemma 5.1. $\mathsf{QCSP}_i(\mathrm{NAE}^2)$ *is* $\Sigma_i\mathrm{P}$*-complete under logspace reductions.*

Proof. Since $\text{Pol}(\text{R}_{1/3}) = \text{I}_2 \subseteq \text{Pol}(\text{S}_3)$, Proposition 4.2 states that $\mathsf{QCSP}_i(\text{S}_3)$ is logspace many-one reducible to $\mathsf{QCSP}_i(\text{R}_{1/3})$. Since $\mathsf{QCSP}_i(\text{S}_3)$ is complete for $\Sigma_i\text{P}$, it suffices to show $\mathsf{QCSP}_i(\text{R}_{1/3}) \leq \mathsf{QCSP}_i(\text{NAE}^2)$.

Let φ be a $\mathsf{QCSP}_i(\text{R}_{1/3})$-formula, $\varphi = Q_1X_1 \ldots \exists X_i \bigwedge_{j=1}^{p} \text{R}_{1/3}(x_{j_1}, x_{j_2}, x_{j_3})$. For each constraint $\text{R}_{1/3}(x_{j_1}, x_{j_2}, x_{j_3})$, introduce the following conjunction of NAE^2 constraints:

$$\text{R}_{2/4}(x_{j_1}, x_{j_2}, x_{j_3}, t) = \bigwedge_{j \neq k \in \{j_1, j_2, j_3\}} \text{NAE}^2(x_j, x_k, t) \wedge \text{NAE}^2(x_{j_1}, x_{j_2}, x_{j_3}).$$

Let $\varphi' = Q_1 t, X_1 \ldots \exists X_i \bigwedge_{j=1}^{p} \text{R}_{2/4}(x_{j_1}, x_{j_2}, x_{j_3}, t)$. Since $\text{R}_{1/3}(x, y, z) = \text{R}_{2/4}(x, y, z, 1)$, $\varphi'[t/1]$ is true iff φ is true. Since $\text{R}_{1/3}(\bar{x}, \bar{y}, \bar{z}) = \text{R}_{2/4}(x, y, z, 0)$, $\varphi'[t/0]$ is true iff $\text{Ren}(\varphi)$ is true, where $\text{Ren}(\varphi)$ is obtained from φ by renaming all variables x by their negation $\bar{x}$. Finally, since $\text{Ren}(\varphi)$ is true iff φ is true, we proved that φ is true if and only if φ' is true.

More generally, it can be shown that the completeness result for NAE holds for any finite domain:

Lemma 5.2. $\mathsf{QCSP}_i(\text{NAE}^m)$ *is complete for* $\Sigma_i\text{P}$ *under logspace reductions.*

Proof. The proof is nearly identical to the proof for Lemma 6.8 below.

This result allows to identify a larger class of $\Sigma_i\text{P}$-complete problems over finite domains, namely the ones for which the set of polymorphisms consists only of constants or essentially unary functions. A k-ary function $f\colon \mathcal{D}^k \to \mathcal{D}$ is *essentially unary* if there is a non-constant unary function $g\colon \mathcal{D} \to \mathcal{D}$ and some $1 \leq i \leq k$ such that $f(v_1, \ldots, v_k) = g(v_i)$ for all $v_1, \ldots, v_k \in \mathcal{D}$.

Lemma 5.3. *For every finite domain* $\mathcal{D}$*, there exists a relation* R_0 *defined over* $\mathcal{D}$ *such that* $\text{Pol}(R_0)$ *contains all essentially unary functions and all constants, and such that* $\mathsf{QCSP}_i(R_0)$ *is* $\Sigma_i\text{P}$*-complete under logspace reductions.*

Proof. Let $\mathcal{D}$ be a finite domain of size m. Let R_0 be the $(m+3)$-ary relation

$$R_0 = \big\{(t_1, \ldots, t_m, x_1, x_2, x_3) \big| \, |\{t_1, \ldots, t_m\}| \leq m - 1 \text{ or } \text{NAE}^m(x_1, x_2, x_3)\big\}.$$

It is clear that $\text{Pol}(R_0)$ contains all the constants. It is also easy to see that R_0 is closed under unary functions g (if g is injective, then the NAE-property is invariant under g, and if g is not injective, then $|\{g(t_1), \ldots, g(t_m)\}| \leq m - 1$), and therefore R_0 is closed under essentially unary functions. Now we prove that $\mathsf{QCSP}_i(\text{NAE}^m)$, which is complete for $\Sigma_i\text{P}$, can be reduced to $\mathsf{QCSP}_i(R_0)$ in logarithmic space.

Let $\varphi = Q_1X_1 \ldots \exists X_i \bigwedge_{j=1}^{p} \text{NAE}^m(x_{j_1}, x_{j_2}, x_{j_3})$ be an instance of $\mathsf{QCSP}_i(\text{NAE}^m)$. Let $\varphi' = Q_1X_1 \ldots \forall X_{i-1} \forall t_1 \ldots \forall t_m \exists X_i \bigwedge_{j=1}^{p} R(t_1, \ldots, t_m, x_{j_1}, x_{j_2}, x_{j_3})$. It is clear that φ is true if and only if φ' is true, concluding the proof of the lemma.

This lemma yields the following completeness result.

Theorem 5.4. *Let* S *be a set of relations over a finite domain* $\mathcal{D}$*. If* $\text{Pol}(S)$ *consists only of essentially unary functions and constants, then* $\mathsf{QCSP}_i(S)$ *is* $\Sigma_i\text{P}$*-complete under logspace reductions.*

Proof. We have $\text{Pol}(S) \subseteq \text{Pol}(R_0)$, where R_0 is the relation exhibited in Lemma 5.3. Hence, the conclusion follows from Proposition 4.2.

Theorem 5.4 settles the Boolean case completely, thus reproving via the algebraic approach a result first obtained by E. Hemaspaandra [Hem04].

Theorem 5.5. *Let S be a set of Boolean relations. If S is Schaefer, then $\mathsf{QCSP}_i(S)$ is in P, otherwise $\mathsf{QCSP}_i(S)$ is Σ_iP-complete under logspace reductions.*

Proof. The polynomial cases follow from Theorem 4.1. According to the closure properties of a non-Schaefer set (see [BCRV04], Section 2), the case $\mathrm{Pol}(S) = \mathrm{N}$ remains. Since N is the set of all essentially unary Boolean functions and constants, the theorem follows from Proposition 5.4.

6 Complexity of Counting Problems

6.1 Introduction to Counting Problems

Let Σ, Γ be alphabets and let $R \subseteq \Sigma^* \times \Gamma^*$ be a binary relation between strings such that, for each $x \in \Sigma^*$, the set $R(x) = \{y \in \Gamma^* \mid R(x, y)\}$ is finite. We write $\#R$ to denote the following counting problem: Given a string $x \in \Sigma^*$, find the cardinality $|R(x)|$ of the set $R(x)$ associated with x.

Valiant [Val79a,Val79b] was the first to investigate the computational complexity of counting problems. To this end, he introduced the class #P of counting functions that count the number of accepting paths of nondeterministic polynomial-time Turing machines. Toda [Tod91] has introduced higher complexity counting classes using a predicate-based framework that focuses on the complexity of membership in the witness sets. Specifically, if $\mathcal{C}$ is a complexity class of decision problems, then $\#\cdot\mathcal{C}$ is the class of all counting problems whose witness relation R satisfies the following conditions:

1. There is a polynomial $p(n)$ such that for every x and every y with $R(x, y)$, we have that $|y| \leq p(|x|)$, where $|x|$ is the length of x and $|y|$ is the length of y.
2. The witness recognition problem "given x and y, does $R(x, y)$ hold?" is in $\mathcal{C}$.

Following Toda [Tod91], $\#\cdot\Sigma_k\mathrm{P} \subseteq \#\cdot\Pi_k\mathrm{P} = \#\mathrm{P}^{\Sigma_k\mathrm{P}} \subseteq \#\cdot\Sigma_{k+1}\mathrm{P}$ holds for each k.

Several notions of reducibilities for counting problems have been defined. The strongest is the one of *parsimonious* reduction [Val79a], which is a polynomial-time many-one reduction preserving the number of witnesses. The aforementioned counting classes are closed under this reduction, but it does not allow to prove completeness of many known #P-complete problems. Valiant [Val79b] used so called counting reductions (essentially Turing-reductions with one oracle query) in his #P-completeness proofs, but the aforementioned counting classes are not closed under these reductions [TW92]. In fact, the closure of #P under counting reductions gives already $\#\cdot\mathrm{PH}$ ($\mathrm{PH} = \cup_i \Sigma_i\mathrm{P}$).

In [BCC+04] the notion of *complementive reduction* appeared to be useful for Boolean constraint satisfaction problems involving complementive relations. More generally such reductions naturally appear when the set of relations is invariant under permutations of the domain. Therefore we generalize the notion of complementive reduction introduced in [BCC+04] to the one of *permutative reduction.* On the one hand it will allow us to get completeness results, on the other hand it has the advantage that #P and all higher complexity classes $\#\cdot\Pi_k\mathrm{P}$, $k \geq 1$ are still closed under permutative reductions (see [BCC+04]). Thus, to use permutative reductions for hardness results in the hierarchy of the classes $\#\cdot\Sigma_i\mathrm{P}$ makes perfect sense. A problem complete for $\#\cdot\Sigma_{i+1}\mathrm{P}$ cannot be in $\#\cdot\Sigma_i\mathrm{P}$, unless the polynomial-time hierarchy collapses to $\Sigma_i\mathrm{P}$.

Before we can state the definition of permutative reductions, we need some additional notions. We enlarge every permutation π on Γ to the strings in Γ^* by means of $\pi(x_1 \cdots x_k) = \pi(x_1) \cdots \pi(x_k)$ for each string $x_1 \cdots x_k \in \Gamma^*$. A set of strings $E \subseteq \Gamma^*$ over an alphabet Γ is called *permutative* if for all permutations Π on Γ it holds that $x \in E$ implies $\Pi(x) \in E$. Given two alphabets Σ, Γ, a binary relation $B \subseteq \Sigma^* \times \Gamma^*$ is said to be *permutative* if the sets $B(x)$ for each string $x \in \Sigma^*$ are permutative.

Definition 6.1. *Let Σ, Γ be two alphabets, $m = |\Gamma|$, and let $\#A$ and $\#B$ be two counting problems determined by the binary relations A and B between the strings from Σ and Γ, where B is permutative.*

- *We say that $\#A$ reduces to $\#B$ via a* ***strong permutative reduction****, if there exist $f, g \in \mathrm{FP}$ such that for every string $x \in \Sigma^*$:*
  - $B(g(x)) \subseteq B(f(x))$
  - $m! \cdot |A(x)| = |B(f(x))| - |B(g(x))|$.
- *A* ***permutative reduction*** *$\#A \leq_{pm} \#B$ is the transitive closure of strong permutative and parsimonious reductions.*

Since permutative reductions are in a sense a generalization of complementive reductions from the Boolean to the general case, it is an easy observation that all hardness results we give below for Boolean $\#\mathsf{QCSP}_i(S)$ hold for complementive reductions.

In the Boolean case Creignou and Hermann [CH96] proved that the complexity of the counting problem $\#\mathsf{CSP}(S)$ of S-formulas is dichotomous: $\#\mathsf{CSP}(S)$ is in FP, if S is a set of affine relations, otherwise $\#\mathsf{CSP}(S)$ is #P-complete. Bauland *et al.* [BCC+04] exhibited a trichotomy result, FP, #P-complete and #·NP-complete, for the counting problem associated with conjunctive queries, i.e., existentially quantified formulas.

6.2 Counting Problems Associated with Quantified Formulas

We are interested in the counting problem associated with $\mathsf{QCSP}_i(S)$-formulas. Therefore we consider quantified formulas φ with free variables Y, $\varphi(Y) = \exists X_1 \forall X_2 \ldots \exists X_i \psi(Y, X_1, \ldots, X_i)$, where ψ is quantifier-free. We are interested in the number of assignments for Y such that $\varphi(Y)$ holds, we denote by $\#\mathsf{sat}(\varphi)$ this number (and by $\#\mathsf{unsat}(\varphi)$ the number of assignments for Y such that $\varphi(Y)$ does not hold). Let us denote by $\#\mathsf{QSAT}_i$ the problem of counting the satisfying assignments of a quantified Boolean formula with free variables and $i-1$ quantifiers alternations starting with an $\exists$-quantifier. This problem is prototypical for $\#\cdot\Sigma_i$P-complete problems under parsimonious reductions. It remains $\#\cdot\Sigma_i$P-complete when the formula is restricted to be 3-CNF for i odd, and 3-DNF for i even [DHK00]. Therefore, it is natural to define the counting problem associated with $\mathsf{QCSP}_i(S)$ as follows.

Definition 6.2. *Let S be a set of relations. Then, $\#\mathsf{QCSP}_i(S)$ is the counting problem to determine, for a $\mathsf{QCSP}_i(S)$-formula φ with free variables, $\#\mathsf{sat}(\varphi)$ for i odd, and $\#\mathsf{unsat}(\varphi)$ for i even.*

Observe that $\#\mathsf{QCSP}_1(S)$ is the same as the problem $\#\mathsf{SAT\text{-}COQ}(S)$ studied in [BCC+04]. Note that $\#\mathsf{QCSP}_i(S) \in \#\cdot\Sigma_i\mathrm{P}$, and that according to the remark above $\#\mathsf{QCSP}_i(S_3)$ is $\#\cdot\Sigma_i$P-complete. Our goal is to study the complexity of $\#\mathsf{QCSP}_i(S)$ for all possible sets S. A central result for our development is the following easy consequence of Proposition 3.1.

Proposition 6.3. *Let S_1 and S_2 be two sets of relations over the same finite domain $\mathcal{D}$, such that S_1 is finite. If the inclusion $\mathrm{Pol}(S_2) \subseteq \mathrm{Pol}(S_1)$ holds, then there exists a parsimonious reduction from $\#\mathsf{QCSP}_i(S_1)$ to $\#\mathsf{QCSP}_i(S_2)$.*

Proof. This is trivial, since the last quantifier in our formulas is always $\exists$, and therefore we can just use the co-clone closure properties.

Our work will essentially follow the same line as the one for the corresponding decision problems in the previous section.

Proposition 6.4. $\#\mathsf{QCSP}_i(\mathrm{R}_{1/3})$ *is $\#\cdot\Sigma_i$P-complete under parsimonious reductions.*

Proof. $\#\mathsf{QCSP}_i(\mathrm{S}_3)$ is $\#{\cdot}\Sigma_i\mathrm{P}$-complete under parsimonious reductions. Now apply Proposition 6.3 (remember $\mathrm{Inv}(\mathrm{R}_{1/3}) = \mathrm{I}_2$).

Proposition 6.5. *$\#\mathsf{QCSP}_i(\mathrm{NAE}^2)$ is $\#{\cdot}\Sigma_i\mathrm{P}$-complete under permutative reductions.*

Proof. We show that $\#\mathsf{QCSP}_i(\mathrm{R}_{1/3})$ can be reduced to $\#\mathsf{QCSP}_i(\mathrm{NAE}^2)$. The construction is very similar to the one in the proof of Proposition 5.1.

Let $\varphi(Y)$ be a $\mathsf{QCSP}_i(\mathrm{R}_{1/3})$-formula with free variables Y (suppose $Y = y_1, \ldots, y_n$) and $\varphi(Y) = Q_1X_1 \ldots \forall X_{i-1}\exists X_i C_1 \wedge \cdots \wedge C_m$ such that each C_j is of the form $C_j = \mathrm{R}_{1/3}(v_{j_1}, v_{j_2}, v_{j_3})$ for some $v_{j_1}, v_{j_2}, v_{j_3} \in Y \cup X_1 \cup \cdots \cup X_i$. Consider now the formula

$$\varphi_1(Y, u, v) = Q_1X_1 \ldots \forall X_{i-1}\exists X_i \bigwedge_{j=1}^{m} C_j \wedge \mathrm{R}_{1/3}(u, u, v),$$

where u and v are two new variables. Observe that $\#\mathsf{sat}(\varphi_1) = \#\mathsf{sat}(\varphi)$ and $\#\mathsf{unsat}(\varphi_1) = 2^{n+2} - \#\mathsf{sat}(\varphi)$. Now, let t be an additional new variable, and construct the formula $\varphi_2(Y, u, v) = Q_1X_i \ldots \forall X_{i-1}\exists X_i \exists t \bigwedge_{j=1}^{m} \mathrm{R}_{2/4}(v_{j_1}, v_{j_2}, v_{j_3}, t) \wedge \mathrm{R}_{2/4}(u, u, v, t)$, where each relation $\mathrm{R}_{2/4}(a, b, c, d)$ stands for the equivalent conjunction of NAE^2-clauses. We get a $\mathsf{QCSP}_i(\mathrm{NAE}^2)$-formula $\varphi_2(Y, u, v)$ such that $\#\mathsf{sat}(\varphi_2) = 2\#\mathsf{sat}(\varphi)$ and $\#\mathsf{unsat}(\varphi_2) = 2^{n+2} - 2\#\mathsf{sat}(\varphi)$.

Now consider the formula $\varphi_3(Y, u, v) = \mathrm{NAE}^2(u, u, v) \wedge \bigwedge_{j=1}^{n} \mathrm{NAE}^2(u, v, y_j)$. Observe that $\mathsf{unsat}(\varphi_3) \subseteq \mathsf{unsat}(\varphi_2)$, and that $\#\mathsf{unsat}(\varphi_3) = 2^{n+1}$. For i odd, we construct $\varphi_2(Y, u, v)$ from $\varphi(Y)$. This is a $\mathsf{QCSP}_i(\mathrm{NAE}^2)$-formula which verifies $\#\mathsf{sat}(\varphi) = \#\mathsf{sat}(\varphi_2)/2$. For i even we construct the pair $(\varphi_2(Y, u, v), \varphi_3(Y, u, v))$ of $\mathsf{QCSP}_i(\mathrm{NAE}^2)$-formulas, which verify $\mathsf{unsat}(\varphi_3) \subseteq \mathsf{unsat}(\varphi_2)$ and $\#\mathsf{unsat}(\varphi) = \frac{\#\mathsf{unsat}(\varphi_2) - \#\mathsf{unsat}(\varphi_3)}{2}$. Thus, in both cases we have a permutative reduction from $\#\mathsf{QCSP}_i(\mathrm{R}_{1/3})$ to $\#\mathsf{QCSP}_i(\mathrm{NAE}^2)$.

Lemma 6.6. *There exists a Boolean relation R_0 such that $\mathrm{N} \subseteq \mathrm{Pol}(R_0)$ and $\#\mathsf{QCSP}_i(R_0)$ is $\#{\cdot}\Sigma_i\mathrm{P}$-complete under polynomial-time reductions.*

Proof. Observe that the reduction provided in the proof of Lemma 5.3 is parsimonious. Thus, the conclusion follows from Proposition 6.5.

We are now in a position to prove the following complexity classification, which completely classifies the $\#\mathsf{QCSP}_i(S)$ problem for the Boolean case.

Theorem 6.7. *Let S be a non-empty finite set of Boolean relations and $i \geq 1$.*

- *If S is affine, then $\#\mathsf{QCSP}_i(S)$ is in* FP.
- *Else if S is bijunctive, or Horn, or dual Horn, then $\#\mathsf{QCSP}_i(S)$ is $\#\mathrm{P}$-complete under counting reductions.*
- *Otherwise, $\#\mathsf{QCSP}_i(S)$ is $\#{\cdot}\Sigma_i\mathrm{P}$-complete under permutative reductions.*

Proof. If S is affine, then the Gaussian elimination algorithm given in [CH96] for $\#\mathsf{CSP}(S)$ can also be used to construct a corresponding polynomial-time algorithm for $\#\mathsf{QCSP}_i(S)$.

If S is Horn, dual Horn, or bijunctive, then $\mathsf{QCSP}(S)$ (and a fortiori $\mathsf{QCSP}_i(S)$) is in P (see Theorem 4.1) and therefore $\#\mathsf{QCSP}_i(S)$ is in $\#\mathrm{P}$. Moreover, we know from [CH96] that in this case $\#\mathsf{CSP}(S)$ is $\#\mathrm{P}$-hard. Hence, the trivial reduction from $\#\mathsf{CSP}(S)$ to $\#\mathsf{QCSP}_i(S)$ shows that $\#\mathsf{QCSP}_i(S)$ is $\#\mathrm{P}$-complete.

The case $\mathrm{Pol}(S) = \mathrm{N}$ follows from Lemma 6.6 and Proposition 6.3.

We have seen that the clone containing all essentially unary or constant functions gives rise to hard constraint satisfaction problems in the decision problem, and in the Boolean counting problem. We now show this hardness result also holds for arbitrary finite domains.

Lemma 6.8. $\#\mathsf{QCSP}_i(\mathrm{NAE}^m)$ *is complete for* $\#{\cdot}\Sigma_i\mathrm{P}$ *under permutative reductions.*

Proof. We use ideas from the proof for Proposition 4.1 in [BKBJ02]. Observe that $x \neq y$ can be expressed as $\mathrm{NAE}(x, x, y)$ over any domain. The proof is by induction. The case $m = 2$ follows from Proposition 6.5. We now show $\#\mathsf{QCSP}_i(\mathrm{NAE}^m) \leq_{\mathrm{pm}} \#\mathsf{QCSP}_i(\mathrm{NAE}^{m+1})$. Let φ be a NAE^m-formula with n free variables $X = \{x_1, \ldots, x_n\}$, existentially quantified variables $Y = \{y_1, \ldots, y_{n_y}\}$ and universally quantified variables $Z = \{z_1, \ldots, z_{n_z}\}$. We add free variables $x_{n+1}, \ldots, x_{n+m}$ with disequality constraints between any two of them. We call the result φ_m, and observe that $\#\mathsf{sat}(\varphi_m) = m! \cdot \#\mathsf{sat}(\varphi)$. We construct a formula φ_{m+1} as follows:

- Copy the formula φ_m and replace every relation symbol NAE^m with NAE^{m+1}, and add a new free variable w.
- For each free or $\exists$-quantified variable $v \in X \cup \{x_{n+1}, \ldots, x_{n+m}\} \cup Y$, add a constraint $v \neq w$.
- For each universally quantified variable z_i, change $\forall z_i$ to $\forall z'_i$, and add $\exists t_{i,1}, \ldots, t_{i,m-1}, z_i$ to the next $\exists$- block, add disequality constraints $t_{i,j} \neq t_{i,k}$ for $j \neq k$ and $(z_i \neq t_{i,j}) \wedge (z'_i \neq t_{i,j}) \wedge (w \neq t_{i,j})$ for all $j \in \{1, \ldots, m-1\}$.

Note that the set of satisfying assignments to these formulas is closed under permutations of the domain, and every solution assigns exactly m different values to the free variables $x_1, \ldots, x_{n+m}$. We call two assignments I_0 and I_1 equivalent if there is a permutation Π of the domain such that $I_0(x_i) = \Pi(I_1(x_i))$ for all i. For each equivalence class I, let I^0 be one canonical representative that does not use the value m, for example the one of minimal lexicographic order. It can be verified that for these assignments $I^0 \models \varphi_m$ holds if and only if $(I^0 \cup \{w = m\}) \models \varphi_{m+1}$ holds.

Since each I^0 represents $m!$ (resp. $(m+1)!$) satisfying assignments of φ_m (resp. φ_{m+1}) - one for each permutation of the domain (note that the value for w is fully determined by the values for $x_{n+1}, \ldots, x_{n+m}$ and therefore the additional variable w does not add another factor), we have

$$\#\mathsf{sat}(\varphi_{m+1}) = (m+1) \cdot \#\mathsf{sat}(\varphi_m) = (m+1)! \cdot \#\mathsf{sat}(\varphi).$$

Also observe that this gives us a parsimonious reduction, if we consider the number of satisfying (or unsatisfying) assignments up to permutations of the domain.

Now we make a case distinction between i odd and i even:

i **odd** In this case, we are interested in the number of satisfying assignments. Let $g := x_1 \neq x_1$, then obviously g is unsatisfiable, and the following holds:

- $(m+1)! \cdot \#\mathsf{sat}(\varphi) = \#\mathsf{sat}(\varphi_{m+1}) - \#\mathsf{sat}(g)$
- $\mathsf{sat}(g) \subseteq \mathsf{sat}(\varphi)$

i **even** Here we count the number of unsatisfying assignments.

Let $g := \displaystyle\bigwedge_{n+1 \leq i < j \leq n+m} x_i \neq x_j \wedge (x_1 \neq w) \wedge \cdots \wedge (x_{m+n} \neq w).$

Since all clauses of g appear in φ_{m+1}, we have $\mathsf{unsat}(g) \subseteq \mathsf{unsat}(\varphi_{m+1})$. It holds that $\#\mathsf{sat}(g) = (m+1)! \cdot m^n$: There are $(m+1)!$ possibilities for the variables $x_{n+1}, \ldots, x_{n+m}$, which leaves only one value for w, and the variables $x_1, \ldots, x_n$ can take any combination of values from $\{0, \ldots, m\} \setminus \{w\}$.

Now, let $d := \#\mathsf{sat}(\varphi)$, and observe that the following holds:

$$\begin{aligned}
\#\mathsf{unsat}(\varphi) &= m^n - \#\mathsf{sat}(\varphi) = m^n - d \\
\#\mathsf{unsat}(\varphi_{m+1}) &= (m+1)^{m+n+1} - \#\mathsf{sat}(\varphi_{m+1}) = (m+1)^{m+n+1} - (m+1)! \cdot d \\
\#\mathsf{unsat}(g) &= (m+1)^{m+n+1} - \#\mathsf{sat}(g) = (m+1)^{m+n+1} - (m+1)! \cdot m^n.
\end{aligned}$$

This implies $\#\mathsf{unsat}(\varphi_{m+1}) - \#\mathsf{unsat}(g) = (m+1)! \cdot \#\mathsf{unsat}(\varphi_m)$.

Therefore, in both cases we have a permutative reduction.

Corollary 6.9. *Let S be a finite set of relations such that* $\mathrm{Pol}(S)$ *only contains constants and essentially unary functions. Then* $\#\mathsf{QCSP}_i(S)$ *is complete for* $\#\cdot\Sigma_i\mathrm{P}$ *under permutative reductions.*

Proof. $\#\mathsf{QCSP}_i(\mathrm{NAE}^m)$ is complete for $\#\cdot\Sigma_i\mathrm{P}$ due to Lemma 6.8. Now use the same reduction as in Lemma 5.3, which is parsimonious.

7 Conclusion

We examined the complexity of the problems $\mathsf{QCSP}_i(S)$ and $\#\mathsf{QCSP}_i(S)$. For sets S of relations over the Boolean universe we presented complete classifications. For non-Boolean universes we obtained a number of quite general hardness results. Contrasting our results with the tractable cases presented in [BBJK03,Che04], an already quite detailed picture of the complexity of quantified constraints emerges.

Our trichotomy for Boolean $\#\mathsf{QCSP}_i(S)$ can easily be generalized to the case of an unbounded number of alternations. Denoting this problem by $\#\mathsf{QCSP}(S)$, a classification completely analogous to Theorem 6.7, but replacing $\#\cdot\Sigma_i\mathrm{P}$ by #PSPACE, is obtained. Here, #PSPACE in the sense of Valiant [Val79a] denotes $\#\mathrm{P}^{\mathrm{PSPACE}}$. It is easy to observe that #PSPACE coincides with Ladner's class ♮PSPACE [Lad89]. (*Caveat:* What Ladner denotes by #PSPACE is a *different class.*) Ladner proves that #PSPACE = ♮PSPACE additionally coincides with FPSPACE(poly), the class of all polynomially length-bounded functions computable in polynomial space, and he observes that #QSAT is complete in this class under parsimonious reductions.

The main open problem is certainly to obtain a finer or even complete classification for the problems $\#\mathsf{QCSP}_i(S)$ and $\#\mathsf{QCSP}(S)$ for non-Boolean universes. Bulatov's results [Bul02] may be a hint that in the case of a 3-element universe this is no hopeless pursuit.

References

[BBJK03] F. Börner, A. Bulatov, P. Jeavons, and A. Krokhin. Quantified constraints: algorithms and complexity. In *Proceedings 17th International Workshop on Computer Science Logic*, volume 2803 of *Lecture Notes in Computer Science*, Berlin Heidelberg, 2003. Springer Verlag.

[BCC+04] M. Bauland, P. Chapdelaine, N. Creignou, M. Hermann, and H. Vollmer. An algebraic approach to the complexity of generalized conjunctive queries. In *Proceedings 7th International Conference on Theory and Applications of Satisfiability Testing*, pages 181–190, 2004.

[BCRV03] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with Boolean blocks, part I: Post's lattice with applications to complexity theory. *SIGACT News*, 34(4):38–52, 2003.

[BCRV04] E. Böhler, N. Creignou, S. Reith, and H. Vollmer. Playing with Boolean blocks, part II: Constraint satisfaction problems. *SIGACT News*, 35(1):22–35, 2004.

[BJK00] A. Bulatov, P. G. Jeavons, and A. A. Krokhin. Constraint satisfaction problems and finite algebras. In *Proceedings 27th International Colloqium on Automata, Languages and Programming*, volume 1853 of *Lecture Notes in Computer Science*, pages 272–282, Berlin Heidelberg, 2000. Springer Verlag.

[BJK04] A. Bulativ, P. Jeavons, and A. Krokhin. Classifying the complexity of constraints using finite algebrais. URL: http://web.comlab.ox.ac.uk/oucl/work/andrei.bulatov/finalg.ps, 2004.

[BKBJ02] F. Börner, A. Krokhin, A. Bulatov, and P. Jeavons. Quantified constraints and surjective polymorphisms. Technical Report PRG-RR-02-11, Computing Laboratory, University of Oxford, UK, 2002.

[Bul02] A. Bulatov. A dichotomy theorem for constraints on a three-element set. In *Proceedings 43rd Symposium on Foundations of Computer Science*, pages 649–658. IEEE Computer Society Press, 2002.

[CH96] N. Creignou and M. Hermann. Complexity of generalized satisfiability counting problems. *Information and Computation*, 125:1–12, 1996.

[Che04] H. Chen. *The computational complexity of quantified constraint satisfaction.* PhD thesis, Cornell Universtiy, 2004.

[CKS01] N. Creignou, S. Khanna, and M. Sudan. *Complexity Classifications of Boolean Constraint Satisfaction Problems.* Monographs on Discrete Applied Mathematics. SIAM, 2001.

[Dal97] V. Dalmau. Some dichotomy theorems on constant-free quantified boolean formulas, 1997.

[Dal00] V. Dalmau. *Computational complexity of problems over generalized formulas.* PhD thesis, Department de Llenguatges i Sistemes Informàtica, Universitat Politécnica de Catalunya, 2000.

[DHK00] A. Durand, M. Hermann, and P. G. Kolaitis. Subtractive reductions and complete problems for counting complexity classes. In *25th International Symposium on Mathematical Foundations of Computer Science*, volume 1893 of *Lecture Notes in Computer Science*, pages 323–332. Springer-Verlag, 2000.

[FK05] T. Feder and P. G. Kolaitis. Closures and dichotomies for quantified constraints. URL: http://theory.stanford.edu/∼tomas/clodi.ps, 2005.

[FV98] T. Feder and M. Y. Vardi. The computational structure of monotone monadis SNP and constraint satisfaction: a study through Datalog and group theory. *SIAM Journal on Computing*, 28(1):57–104, 1998.

[Hem04] E. Hemaspaandra. Dichotomy theorems for alternation-bounded quantified boolean formulas. *CoRR*, cs.CC/0406006, 2004.

[JCG97] P. G. Jeavons, D. A. Cohen, and M. Gyssens. Closure properties of constraints. *Journal of the ACM*, 44(4):527–548, 1997.

[Jea98] P. G. Jeavons. On the algebraic structure of combinatorial problems. *Theoretical Computer Science*, 200:185–204, 1998.

[Lad75] R. Ladner. On the structure of polynomial-time reducibility. *Journal of the ACM*, 22:155–171, 1975.

[Lad89] R. Ladner. Polynomial space counting problems. *SIAM Journal on Computing*, 18(6):1087–1097, 1989.

[MS72] A. R. Meyer and L. J. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential time. In *Proceedings 13th Symposium on Switching and Automata Theory*, pages 125–129. IEEE Computer Society Press, 1972.

[Pap94] C. H. Papadimitriou. *Computational Complexity.* Addison-Wesley, Reading, MA, 1994.

[Pip97] N. Pippenger. *Theories of Computability.* Cambridge University Press, Cambridge, 1997.

[PK79] R. Pöschel and L. A. Kalužnin. *Funktionen- und Relationenalgebren.* Deutscher Verlag der Wissenschaften, Berlin, 1979.

[Pös01] R. Pöschel. Galois connection for operations and relations. Technical Report MATH-LA-8-2001, Technische Universität Dresden, 2001.

[Rei04] O. Reingold. Undirected st-connectivity in log-space. Technical Report TR04-094, ECCC Reports, 2004.

[Sch78] T. J. Schaefer. The complexity of satisfiability problems. In *Proccedings 10th Symposium on Theory of Computing*, pages 216–226. ACM Press, 1978.

[SM73] L. J. Stockmeyer and A. R. Meyer. Word problems requiring exponential time. In *Proceedings 5th ACM Symposium on the Theory of Computing*, pages 1–9. ACM Press, 1973.

[Sto77] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1977.

[Tod91] S. Toda. *Computational Complexity of Counting Complexity Classes.* PhD thesis, Tokyo Institute of Technology, Department of Computer Science, Tokyo, 1991.

[TW92] S. Toda and O. Watanabe. Polynomial time 1-Turing reductions from #PH to #P. *Theoretical Computer Science*, 100:205–221, 1992.

[Val79a] L. G. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.

[Val79b] L. G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal of Computing*, 8(3):411–421, 1979.

[Wra77] C. Wrathall. Complete sets and the polynomial-time hierarchy. *Theoretical Computer Science*, 3:23–33, 1977.

ECCC ISSN 1433-8092
http://www.eccc.uni-trier.de/eccc
ftp://ftp.eccc.uni-trier.de/pub/eccc
ftpmail@ftp.eccc.uni-trier.de, subject 'help eccc'