Precise Upper and Lower bounds for the Monotone Constraint Satisfaction Problem

Victor Lagerkvist[1]

Department of Computer and Information Science, Linköping University, Sweden
victor.lagerkvist@liu.se

Abstract. The *monotone constraint satisfaction problem* (MCSP) is the problem of, given an existentially quantified positive formula, decide whether this formula has a model. This problem is a natural generalization of the constraint satisfaction problem, which can be seen as the problem of determining whether a conjunctive formula has a model. In this paper we study the worst-case time complexity, measured with respect to the number of variables, n, of the MCSP problem parameterized by a constraint language Γ (MCSP(Γ)). We prove that the complexity of the NP-complete MCSP(Γ) problems on a given finite domain D falls into exactly $|D| - 1$ cases and ranges from $O(2^n)$ to $O(|D|^n)$. We give strong lower bounds and prove that MCSP(Γ), for any constraint language Γ over any finite domain, is solvable in $O(|D'|^n)$ time, where D' is the domain of the core of Γ, but not solvable in $O(|D'|^{\delta n})$ time for any $\delta < 1$, unless the strong exponential-time hypothesis fails. Hence, we obtain a complete understanding of the worst-case time complexity of MCSP(Γ) for constraint languages over arbitrary finite domains.

1 Introduction

The *constraint satisfaction problem* over a constraint language Γ (CSP(Γ)) is a widely studied computational problem which can be described as the problem of, given a conjunctive formula over Γ, verify whether there exists a model of this formula. In general the CSP(Γ) problem is NP-complete, and much research has been made to separate tractable from NP-hard cases [3,13]. A related question to establishing dichotomies between tractable and intractable cases is to study the complexity differences between NP-complete CSP problems. Let n denote the number of variables in a given CSP(Γ) instance. Is it then, for example, possible to characterize all constraint languages Γ such that CSP(Γ) is solvable in $O(c^n)$ time for some $c \in \mathbb{R}$? Ultimately, one would like to have a table, which for every constraint language Γ contains a constant $c \in \mathbb{R}$ such that CSP(Γ) is solvable in $O(c^n)$ time but not in $O(d^n)$ time for any $d < c$. Clearly, even assuming P $\neq$ NP, such a table would be very difficult, if not impossible, to obtain. A more feasible approach is to order all NP-complete CSP problems by their relative worst-case time complexity, i.e., CSP(Γ) lies below CSP(Δ) in this ordering if CSP(Γ) is solvable in $O(c^n)$ time whenever CSP(Δ) is solvable in $O(c^n)$ time. Jonsson et al. [9] studied the structure of this ordering for Boolean

CSP problems and proved that it had a minimal element. With this "easiest" Boolean CSP problem they obtained lower bounds for all NP-complete Boolean CSP problems and proved that there does not exist any NP-complete Boolean CSP(Γ) problem solvable in subexponential time, unless the exponential-time hypothesis fails. A similar study was later conducted for Boolean optimization problems where analogous results and lower bounds were obtained [10].

In this paper we continue this line of research in the context of the *monotone constraint satisfaction problem* parameterized by a constraint language Γ (MCSP(Γ)). This problem can be viewed as a generalization of the CSP problem where the objective is to determine whether an existentially quantified first-order formula without negation over Γ has a model. This problem has been studied by Hermann and Richoux [6], who gave a dichotomy theorem for arbitrary finite domains, separating tractable from NP-complete cases. This result was later extended to also cover the case of infinite domains [1]. Closure properties of disjunctive logic formulas was investigated by Fargier and Marquis [5] but with regards to knowledge representation problems. We are now interested in the aforementioned questions regarding the worst-case time complexity of MCSP(Γ), and in particular how well the ordering between the complexity of NP-complete MCSP(Γ) problems can be approximated. As a tool to compare and relate worst-case running times between NP-complete MCSP(Γ) problems, we utilize a restricted form of polynomial-time reductions, which only increases the number of variables within instances by a constant, *constant variable reductions* (CV-reductions). It is readily verified that if MCSP(Γ) is CV-reducible to MCSP(Δ) then MCSP(Γ) is solvable in $O(c^n)$ time whenever MCSP(Δ) is solvable in $O(c^n)$ time (where n denotes the number of variables in a given instance). We begin in Section 3.1 by first giving a straightforward condition to check whether MCSP(Γ) is CV-reducible to MCSP(Δ). This proof makes use of the *algebraic approach* to constraint satisfaction problems [8] and the Galois connection between strong partial endomorphism monoids and weak Krasner algebras without existential quantification [2]. In Section 3.2 we use this result to obtain a full understanding of the applicability of CV-reductions to the MCSP(Γ) problem. Let Γ be an arbitrary constraint language over a finite domain D and let D' denote the domain of the core of Γ (the reader unfamiliar with this model theoretical concept is advised to quickly consult Section 2.2). We prove that MCSP(Γ) is exactly as hard as the full MCSP over D', i.e. the MCSP problem where all relations over D' are allowed to occur in constraints. This furthermore implies that there are at most $|D| - 1$ possible cases for the worst-case complexity of MCSP(Γ).

With the help of the results from Section 3 we in Section 4 turn to the problem of determining lower bounds for MCSP(Γ) . To prove these lower bounds we relate the complexity of MCSP to the *strong exponential-time hypothesis* (SETH), i.e. the conjecture that the Boolean satisfiability problem is not solvable in $O(2^{\delta n})$ time for any $\delta < 1$ [4,7]. We prove the following: if the SETH holds then MCSP(Γ) is solvable in $O(c^n)$ time but not in $O(c^{\delta n})$ time for any $\delta < 1$, where c is the size of the domain of the core of Γ. As a side result we prove an analogous result also hold for the the CSP problem: if the Boolean CSP problem is not solvable in $O(2^{\delta n})$ for any $\delta < 1$ then the CSP problem over any finite domain D is not solvable in $O(|D|^{\delta n})$ for any $\delta < 1$. Hence, for any finite domain D and any $c \in \{1, \ldots, |D|\}$, we obtain a complete classification of the MCSP problems solvable in $O(c^n)$ time but not in $O(c^{\delta n})$ time for any $\delta < 1$. In contrast to the CSP problem we can therefore not only approximate the ordering between the complexity of NP-complete MCSP problems, but actually obtain a complete understanding. While these results do not directly carry over to the CSP problem, we still believe that some of the involved techniques could be useful when studying the time complexity of other problems parameterized by constraint languages.

2 Preliminaries

In this section we introduce constraint languages, the monotone constraint satisfaction problem, and give a brief introduction to the necessary algebraic concepts required in the subsequent treatment.

2.1 Constraint Languages and Functions

Let $D \subset \mathbb{N}$ be a finite domain. Let Rel_D denote the set of all finitary relations over D. For a relation $R \in \mathrm{Rel}_D$ we let $\mathrm{ar}(R)$ denote its arity. For each $i \in D$ let c_i denote the constant relation $\{(i)\}$. Let $\mathrm{id}_D(i) = i$ be the identity function over D, $\mathrm{Eq}_D = \{(i, i) \mid i \in D\}$ the equality relation, and $\mathrm{Neq}_D = \{(i, j) \in D^2 \mid i \neq j\}$ the inequality relation. Given a function f on D we let $\mathrm{img}(f)$ denote its image. If the domain is clear from the context we simply write id, Eq and Neq, respectively. A *constraint language* Γ over D is a finite or infinite set of relations $\Gamma \subseteq \mathrm{Rel}_D$ such that $\mathrm{Eq}_D \in \Gamma$. Hence, whenever we speak of a constraint language we assume that this language contains the equality relation.

We usually represent relations as logical formulas, and use the notation $R(x_1, \ldots, x_n) \equiv \phi$ to denote the n-ary relation $R = \{(f(x_1), \ldots, f(x_n)) \mid f$ is a model of $\phi\}$. A *monotone formula* over a constraint language Γ is logical formula with free variables $x_1, \ldots, x_n$ of the form $\exists y_1, \ldots, y_n.\phi$, where ϕ is a quantifier-free formula over $x_1, \ldots, x_n, y_1, \ldots, y_m$ consisting of disjunction, conjunction, over positive literals of the form $R_i(\mathbf{x_i})$, where $R_i \in \Gamma$, and $\mathbf{x_i}$ is a tuple of variables over $x_1, \ldots, x_n, y_1, \ldots, y_m$. A *quantifier-free monotone formula* is a monotone formula without existential quantification. These restricted classes of logical formulas will be important in Section 2.3 when we define the monotone constraint satisfaction problem, and in Sections 2.4 and 2.5 where we discuss closure properties of relations.

2.2 Cores of Constraint Languages

A constraint language $\{R'_1, \ldots, R'_m\}$ over $D' \subseteq D$ is a *substructure* of a constraint language $\{R_1, \ldots, R_m\}$ over D if each $R'_i \subseteq R_i$. A *homomorphism* h of a constraint language $\{R_1, \ldots, R_m\}$ to a constraint language $\{R'_1, \ldots, R'_m\}$ over D' is a function $h : D \rightarrow D'$ such that $(h(a_1), \ldots, h(a_{\mathrm{ar}(R_i)})) \in R'_i$ for every i and every $(a_1, \ldots, a_{\mathrm{ar}(R_i)}) \in R_i$. Here we tacitly assume that constraint languages can be viewed as relational structures, i.e. that the relations are given as an ordered sequence and have an associated signature. A constraint language Γ is a *core* if there does not exist a homomorphism to a proper substructure of Γ, and we say that Γ' is a core of Γ if Γ' is a core and there exists a homomorphism from Γ to Γ'. Since the cores of a constraint language Γ are equivalent up to isomorphism we typically speak of *the core* of Γ and let $Core(\Gamma)$ denote this constraint language. The *core-size* of Γ is the size of the domain of $Core(\Gamma)$.

2.3 The Monotone Constraint Satisfaction Problem

Let Γ be a constraint language over D. Recall that a constraint language in our notation always includes the equality relation over D. The *monotone constraint satisfaction problem* over Γ ($\mathrm{MCSP}(\Gamma)$) is defined as follows.

INSTANCE: A tuple (V, ϕ) where V is a set of variables and ϕ a quantifier-free monotone formula over Γ and V.
QUESTION: Is there a function $f : V \rightarrow D$ such that f is a model of ϕ?

Even if $\mathrm{Eq} \notin \Gamma$ we typically write $\mathrm{MCSP}(\Gamma)$ instead of $\mathrm{MCSP}(\Gamma \cup \{\mathrm{Eq}\})$. Given an instance $I = (V, \phi)$ of $\mathrm{MCSP}(\Gamma)$ we let $||I||$ denote the number of bits required to represent I and $\mathrm{Constraints}(I) = \{R_i(\mathbf{x_i}) \mid R_i(\mathbf{x_i})$ is a constraint application in $\phi\}$, where each $R_i \in \Gamma$ and $\mathbf{x_i}$ is a tuple of variables over V of length $\mathrm{ar}(R_i)$.

Example 1. Consider the problem $\mathrm{MCSP}(\{c_1, c_2\})$ over the Boolean domain $\{1, 2\}$, where $c_1 = \{(1)\}$ and $c_2 = \{(2)\}$. Then $\mathrm{MCSP}(\{c_1, c_2\})$ can be seen as a variant of the Boolean satisfiability problem, with the distinction that instances are not necessarily in conjunctive normal form.

Hermann and Richoux [6] classified the complexity of $\mathrm{MCSP}(\Gamma)$ with respect to polynomial-time many-one reductions. Since we are interested in a more fine-grained analysis of the complexity of $\mathrm{MCSP}(\Gamma)$ we introduce a restricted form of reduction which only increases the number of variables within instances by an additive constant.

Definition 2. *Let Γ and Δ be two constraint languages. A* constant variable-reduction *(CV-reduction) from* $\mathrm{MCSP}(\Gamma)$ *to* $\mathrm{MCSP}(\Delta)$ *is a computable function f from the instances of* $\mathrm{MCSP}(\Gamma)$ *to the instances of* $\mathrm{MCSP}(\Delta)$ *such that for every instance (V, ϕ) of* $\mathrm{MCSP}(\Gamma)$*:*

- *$f((V, \phi))$ can be computed in $O(poly(||(V, \phi)||))$ time,*
- *(V, ϕ) is satisfiable if and only if $f((V, \phi))$ is satisfiable,*
- *$f((V, \phi)) = (V', \phi')$ where $|V'| = |V| + O(1)$.*

We write $\mathrm{MCSP}(\Gamma) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Delta)$ as a shorthand for this reduction. In other words a CV-reduction is a polynomial-time many one reduction which only increases the number of variables by a constant. The utility of these reductions stems from the fact that if $\mathrm{MCSP}(\Gamma)$ is solvable in $O(c^n \cdot poly(||I||))$ time for some constant $c \geq 1$, and if $\mathrm{MCSP}(\Delta) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma)$, then $\mathrm{MCSP}(\Delta)$ is also solvable in $O(c^n \cdot poly(||I||))$ time.

2.4 Closure Operators on Functions and Relations

Let R be a k-ary relation over a finite domain D. A unary function e over D is said to be an *endomorphism* of R if $(e(a_1), \ldots, e(a_k)) \in R$ for every $(a_1, \ldots, a_k) \in R$. In this case we also say that e *preserves* R or that R is *invariant* under e. This notion is extended to constraint languages in the obvious way. Given a constraint language Γ we let $\mathrm{End}_D(\Gamma)$ denote the set of all endomorphisms over D of Γ. Similarly, given a set of unary functions E over D we let $\mathrm{Inv}_D(E)$ denote the set of all relations over D that are invariant under E. Since the domain is typically clear from the context we usually just write $\mathrm{End}(\Gamma)$ and $\mathrm{Inv}(E)$. Sets of the form $\mathrm{End}(\Gamma)$ and $\mathrm{Inv}(E)$ are known as *endomorphism monoids* and *weak Krasner algebras*, respectively. Despite these rather enigmatic names they are in fact quite easy to grasp: $\mathrm{End}(\Gamma)$ is a set of unary functions containing the identity function id which is closed under functional composition; $\mathrm{Inv}(E)$ is a set of relations closed under monotone formulas [2]. The latter means that whenever $\Gamma \subseteq \mathrm{Inv}(E)$ then $\mathrm{Inv}(E)$ also contains all relations of the form $R(x_1, \ldots, x_n) \equiv \phi$, where ϕ is a monotone formula over Γ. If we let $\langle \Gamma \rangle = \mathrm{Inv}(\mathrm{End}(\Gamma))$ we obtain the following Galois connection between weak Krasner algebras and endomorphism monoids.

Theorem 3 ([2]). *Let Γ and Δ be two constraint languages. Then $\Gamma \subseteq \langle \Delta \rangle$ if and only if* $\mathrm{End}(\Delta) \subseteq \mathrm{End}(\Gamma)$.

Using this Galois connection, Hermann and Richoux [6] proved that the complexity of $\mathrm{MCSP}(\Gamma)$, up to polynomial-time many one reductions, is determined by the endomorphisms of Γ.

Theorem 4 ([6]). *Let Γ and Δ be two finite constraint languages. If* $\mathrm{End}(\Gamma) \subseteq \mathrm{End}(\Delta)$ *then* $\mathrm{MCSP}(\Delta)$ *is polynomial-time many-ony reducible to* $\mathrm{MCSP}(\Gamma)$.

With this result they obtained a dichotomy theorem for $\mathrm{MCSP}(\Gamma)$ for constraint languages Γ over arbitrary finite domains, proving that $\mathrm{MCSP}(\Gamma)$ is NP-complete if and only if $\mathrm{End}(\Gamma)$ does not contain a constant endomorphism, i.e. an endomorphism e which for some $j \in D$ satisfies $e(i) = j$ for all $i \in D$.

Theorem 5 ([6]). *Let Γ be constraint language. Then* $\mathrm{MCSP}(\Gamma)$ *is NP-complete if and only if* $\mathrm{End}(\Gamma)$ *does not contain a constant endomorphism.*

2.5 Restricted Closure Operators on Functions and Relations

We are now interested in closure operators based on quantifier-free monotone formulas. To get a similar Galois connection as in Theorem 6 we need a slight modification to the $\mathrm{End}(\cdot)$ operator. An n-ary *partial function* f over D is a map $f : X \to D$ where $X \subseteq D^n$. Let $\mathrm{dom}(f) = X$. If f and g are two partial functions then g is a *subfunction* of f if $\mathrm{dom}(g) \subseteq \mathrm{dom}(f)$ and $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ for all $(x_1, \ldots, x_n) \in \mathrm{dom}(g)$. A set F of partial functions is *strong*, if, whenever $f \in F$, then F also contains all subfunctions of f. Given a set of partial functions F we let $\mathrm{Strong}(F)$ denote the smallest strong set of partial functions containing F. A unary partial function e is said to be a *partial endomorphism* of a k-ary relation R if $(e(a_1), \ldots, e(a_k)) \in R$ for all $(a_1, \ldots, a_k) \in R$ such that $(a_1), \ldots, (a_k) \in \mathrm{dom}(e)$. Again, this notion easily generalizes to constraint languages. Let $\mathrm{pEnd}_D(\Gamma)$ denote the set of all partial endomorphisms over D to a constraint language Γ over D. As usual we omit the domain D when it is clear from the context. A set of the form $\mathrm{pEnd}(\Gamma)$ is known as a *strong partial endomorphism monoid* [2] and is a strong, composition-closed set of unary partial functions containing the identity function. The utility of these definitions stems from the following: if E is a set of unary partial functions and if $\Gamma \subseteq \mathrm{Inv}(E)$, then $\mathrm{Inv}(E)$ also contains all relations $R(x_1, \ldots, x_n) \equiv \phi$, where ϕ is a quantifier-free monotone formula over Γ [2]. For a constraint language Γ let $\langle \Gamma \rangle_{\nexists} = \mathrm{Inv}(\mathrm{pEnd}(\Gamma))$. We have the following Galois connection.

Theorem 6 ([2]). *Let Γ and Δ be two constraint languages. Then $\Gamma \subseteq \langle \Delta \rangle_{\nexists}$ if and only if* $\mathrm{pEnd}(\Delta) \subseteq \mathrm{pEnd}(\Gamma)$.

As will be made clear in Section 3, this Galois connection will allow us to obtain a corresponding result to Theorem 4, where we prove that the partial endomorphims of a constraint language Γ determines the complexity of $\mathrm{MCSP}(\Gamma)$ up to $O(c^{|V|})$ time complexity.

3 The Complexity of Monotone Constraint Satisfaction

By Theorem 5 we can for every $\Gamma \subseteq \mathrm{Rel}_D$ easily determine whether $\mathrm{MCSP}(\Gamma)$ is NP-complete or in P. We are interested in a more fine-grained analysis of the NP-complete MCSP problems with respect to CV-reductions. We first (in Section 3.1) prove that the partial endomorphisms of a constraint language determines the complexity of the MCSP problem with respect to CV-reductions. In Section 3.2 we then use partial endomorphism to obtain a full understanding of the applicability of CV-reductions for the MCSP problem.

3.1 Partial Endomorphisms and CV-reductions

We first prove an easy, but very important, theorem which gives a conditition for obtaining a CV-reduction from one MCSP problem to another.

Theorem 7. *Let Γ and Δ be two finite constraint languages. If* $\mathrm{pEnd}(\Gamma) \subseteq \mathrm{pEnd}(\Delta)$ *then* $\mathrm{MCSP}(\Delta)$ *is CV-reducible to* $\mathrm{MCSP}(\Gamma)$.

Proof. Since $\mathrm{pEnd}(\Gamma) \subseteq \mathrm{pEnd}(\Delta)$ we can exploit the Galois connection to infer that $\Delta \subseteq \langle \Gamma \rangle_{\nexists}$. This furthermore implies that every $R \in \Delta$ can be expressed as a quantifier-free monotone formula over Γ. Since both languages are finite we can easily find all such definitions in constant time, with respect to the size of Γ and Δ. Now let $I = (V, \phi)$ be an instance of $\mathrm{MCSP}(\Delta)$. For each constraint $R_i(\mathbf{x_i}) \in \mathrm{Constraints}(I)$ replace it by its equivalent quantifier-free monotone formula over Γ. Let $I' = (V, \phi')$ be the resulting instance. Then I' is satisfiable if and only if I is satisfiable. Since we do not introduce any fresh variables, and since the reduction runs in $O(poly(||I||))$ time, it follows that the reduction is a CV-reduction. $\square$

Since we are working over arbitrary finite domains we are interested in simplyfing things whenever possible. The following theorem offers such a simplification whenever $\mathrm{End}(\Gamma)$ contains a non-injective endomorphism, i.e., when the core-size of Γ is strictly smaller than $|D|$. The proof is simple and is therefore omitted.

Theorem 8. *Let Γ be a finite constraint language. Then (1)* $\mathrm{MCSP}(\Gamma) \leq_{\mathrm{CV}} \mathrm{MCSP}(Core(\Gamma))$ *and (2)* $\mathrm{MCSP}(Core(\Gamma)) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma)$.

3.2 Intervals of Strong Partial Endomorphism Monoids

By Theorem 7 we now have a relatively simple property for determining whether $\mathrm{MCSP}(\Gamma)$ is CV-reducible to $\mathrm{MCSP}(\Delta)$. This condition is sufficient to guarantee the existence of a CV-reduction, but as we will see in this section, there are many cases that are not covered. Assume e.g. that $\mathrm{pEnd}(\Gamma) \subset \mathrm{pEnd}(\Delta)$. Could it then still be the case that $\mathrm{MCSP}(\Delta)$ is CV-reducible to $\mathrm{MCSP}(\Gamma)$? In this section we obtain a complete understanding of when such CV-reductions are possible. Our main technical tool for accomplishing this is to study *intervals* of strong partial endomorphism monoids.

Definition 9. *Let* $\mathrm{End}(\Gamma)$ *be an endomorphism monoid over D. The* strong partial monoid interval of $\mathrm{End}(\Gamma)$ *is the set*

$$\mathcal{I}(\mathrm{End}(\Gamma)) = \{\mathrm{pEnd}(\Delta) \mid \mathrm{End}(\Delta) = \mathrm{End}(\Gamma)\}.$$

The smallest element in this set is given by $\mathrm{Strong}(\mathrm{End}(\Gamma))$ and the largest element by $\bigcup \mathcal{I}(\mathrm{End}(\Gamma)) = \bigcup_{\mathrm{pEnd}(\Delta) \in \mathcal{I}(\mathrm{End}(\Gamma))} \mathrm{pEnd}(\Delta)$. Hence, this set can indeed be viewed as a bounded interval. We illustrate this definition by an example.

Example 10. Consider the Boolean domain $D = \{1, 2\}$ and let $E = \{\mathrm{id}\}$, i.e. the smallest Boolean endomorphism monoid consisting only of the unary projection function. Recall that $c_1 = \{(1)\}$, $c_2 = \{(2)\}$, let $c_{(1,2)} = \{(1, 2)\}$, and let e_1 and e_2 be the two partial functions $e_1(2) = 1$, $e_2(1) = 2$, which are undefined otherwise. Define $\mathrm{pEnd}(\Gamma_1), \ldots, \mathrm{pEnd}(\Gamma_4)$ as:

– $\mathrm{pEnd}(\Gamma_1) = \mathrm{Strong}(\{\mathrm{id}\})$, $\Gamma_1 = \{c_1, c_2\}$,
– $\mathrm{pEnd}(\Gamma_2) = \mathrm{pEnd}(\Gamma_1) \cup \{e_1\}$, $\Gamma_2 = \{c_1, c_{(1,2)}\}$
– $\mathrm{pEnd}(\Gamma_3) = \mathrm{pEnd}(\Gamma_1) \cup \{e_2\}$, $\Gamma_3 = \{c_2, c_{(1,2)}\}$, and
– $\mathrm{pEnd}(\Gamma_4) = \mathrm{pEnd}(\Gamma_1) \cup \mathrm{pEnd}(\Gamma_2) \cup \mathrm{pEnd}(\Gamma_3)$, $\Gamma_4 = \{c_{(1,2)}\}$.

Then one can prove that $\mathcal{I}(E) = \{\mathrm{pEnd}(\Gamma_1), \mathrm{pEnd}(\Gamma_2), \mathrm{pEnd}(\Gamma_3), \mathrm{pEnd}(\Gamma_4)\}$, and it is readily verified that the inclusions $\mathrm{pEnd}(\Gamma_4) \supset \mathrm{pEnd}(\Gamma_3) \supset \mathrm{pEnd}(\Gamma_1)$ and $\mathrm{pEnd}(\Gamma_4) \supset \mathrm{pEnd}(\Gamma_2) \supset \mathrm{pEnd}(\Gamma_1)$ hold.

In Example 10 one can also prove that $\mathrm{MCSP}(\Gamma_1) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma_4)$. Due to the inclusion structure between these strong partial endomorphism monoids this furthermore implies that $\mathrm{MCSP}(\Gamma_1)$, ..., $\mathrm{MCSP}(\Gamma_4)$ are all CV-reducible to each other, and hence solvable within exactly the same $O(c^{|V|})$ running time. We are now interested in whether this holds when considering strong partial endomorphism monoid intervals over arbitrary finite domains. To accomplish this we first need a better characterization of the largest element $\bigcup \mathcal{I}(\mathrm{End}(\Gamma))$.

Definition 11. *Let $E = \mathrm{End}(\Gamma)$ be an endomorphism monoid over $D = \{1, \ldots, k\}$. The relation $E(D)$ is defined as $E(D) = \{(e(1), \ldots, e(k)) \mid e \in E\}$.*

The notation $E(D)$ is a mnemonic with the intended meaning that we are constructing a relation that is closed under every endomorphism in E.

Theorem 12. *Let $E = \mathrm{End}(\Gamma)$ be an endomorphism monoid over D. Then $\mathrm{pEnd}(\{E(D)\}) = \bigcup \mathcal{I}(\mathrm{End}(\Gamma))$.*

Proof. First, we prove that $\mathrm{pEnd}(\{E(D)\}) \in \mathcal{I}(E)$, i.e. that $\mathrm{End}(\{E(D)\}) = E$. By definition $E(D)$ is closed under every function in E, so the inclusion $E \subseteq \mathrm{End}(\{E(D)\})$ holds. Let $e \in \mathrm{End}(\{E(D)\})$ and let $(e(1), \ldots, e(k)) = (a_1, \ldots, a_k)$. Observe that $(a_1, \ldots, a_k) \in E(D)$ since e preserves $E(D)$. By definition of $E(D)$ there then exists $e' \in E$ such that $(e'(1), \ldots, e'(k)) = (e(1), \ldots, e(k)) = (a_1, \ldots, a_k)$. Hence, $e \in E$.

Second, we prove that $\mathrm{pEnd}(E(E)) \supseteq \mathrm{pEnd}(\Delta)$ for any $\mathrm{pEnd}(\Delta) \in \mathcal{I}(E)$. Let $e \in \mathrm{pEnd}(\Delta)$. Assume towards contradiction that $e \notin \mathrm{pEnd}(\{E(D)\})$. Then there exists $(b_1, \ldots, b_k) \in E(D)$ such that $(e(b_1), \ldots, e(b_k)) \notin E(D)$. Let $e' \in \mathrm{End}(\{E(D)\}) = \mathrm{End}(\Delta)$ be the total function satisfying $(e'(1), \ldots, e'(k)) = (a_1, \ldots, a_k)$. Observe that such a function must exist according to the definition of $E(D)$. Then define the unary function g as $g(i) = e(e'(i))$ for every $i \in D$. Clearly, g is a total function which does not preserve $E(D)$ since e is defined on $\mathrm{img}(e')$, but this is a contradiction since $g \in \mathrm{End}(\Delta) = \mathrm{End}(\{E(D)\})$. □

By combining Theorem 7 and Theorem 12 we obtain the following lemma.

Lemma 13. *Let Γ be a finite constraint language over D and let $E = \mathrm{End}(\Gamma)$. Then $\mathrm{MCSP}(\{E(D)\}) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma)$.*

We now have all the machinery in place to characterize the complexity of $\mathrm{MCSP}(\Gamma)$. In particular, we want to prove the converse of Lemma 13, i.e. that $\mathrm{MCSP}(\Gamma)$ is CV-reducible to $\mathrm{MCSP}(\{E(D)\})$. To prove this we first investigate the expressive power of the relation $E(D)$. Recall that Neq_D denotes the binary inequality relation over D.

Theorem 14. *Let Γ be a finite constraint language over D and let $D' \subseteq D$ be the domain of $Core(\Gamma)$. Then* $\mathrm{MCSP}(\{\mathrm{Neq}_{D'}\}) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma)$.

Proof. Let $k' = |D'|$, $k = |D|$, and let e be the homomorphism from Γ to $Core(\Gamma)$. Observe that $e \in \mathrm{End}(\Gamma)$, $\mathrm{img}(e) = D'$, and that there does not exist any $e' \in \mathrm{End}(\Gamma)$ such that $|\mathrm{img}(e')| < |\mathrm{img}(e)|$. We prove that $\mathrm{MCSP}(\{\mathrm{Neq}_{D'}\})$ is CV-reducible to $\mathrm{MCSP}(\{E(D)\})$, which according to Lemma 13 is sufficient to prove the claim. Let $t = (e(1), \ldots, e(k)) \in E(D)$. Assume without loss of generality that $\{e(1), \ldots, e(k')\} = D'$, i.e., that e is injective on the k' first arguments. Since $\mathrm{img}(e) = D'$ this means that for every $i > k'$ there exists a $j \leq k'$ such that $t[i] = t[j]$. Let $h : \{k'+1, \ldots, k\} \to \{1, \ldots, k'\}$ be a function satisfying $t[i] = t(h(i))$ for every $i \in \{k'+1, \ldots, k\}$. Let $R(x_1, \ldots, x_{k'}) \equiv E(D)(x_1, \ldots, x_{k'}, x_{h(k'+1)}, \ldots, x_{h(k)})$, i.e. k'-ary relation obtained by identifying all arguments that are equal in the tuple t. We now claim that if $i \neq j$ then $t'[i] \neq t'[j]$ for every $t' \in R$. Assume to the contrary that there exists $t' \in R$ such that $t[i] = t[j]$ for some $i \neq j$. According to the definition of R this implies that there exists a tuple $t'' \in E(D)$ such that $t''[i] = t''[j]$, and, furthermore, that $t''[h(j')] = t''(j')$ for every $j' \in \{1, \ldots, k'\}$. By letting $e'(1) = t''[1], \ldots, e'(k) = t''[k]$, we see that $\mathrm{img}(e') \subset \mathrm{img}(e)$, a contradiction. Hence, all elements are distinct in every tuple $t' \in R$.

For the reduction, let $I = (V, \phi)$ be an instance of $\mathrm{MCSP}(\{\mathrm{Neq}_{D'}\})$. We introduce k' fresh variables $y_1, \ldots, y_{k'}$ and introduce the constraint $R(y_1, \ldots, y_{k'})$. For each constraint $\mathrm{Neq}(x_i, x_j) \in \mathrm{Constraints}(I)$ replace it by $(\mathrm{Eq}(x_i, y_1) \wedge (\mathrm{Eq}(x_j, y_2) \vee \ldots \vee \mathrm{Eq}(x_j, y_{k'}))) \vee \ldots \vee (\mathrm{Eq}(x_i, y_{k'}) \wedge (\mathrm{Eq}(x_j, y_1) \vee \ldots \vee \mathrm{Eq}(x_j, y_{k'-1})))$. Let I' be the resulting instance over the variables $V \cup \{y_1, \ldots, y_{k'}\}$. Clearly, if I is satisfiable then it is easy to find a satisfying assignment to I'. Similarly, if I' is satisfiable then one can apply e to the satisfying assignment to get an assignment over D'. It follows that the reduction is a CV-reduction since k' is a constant depending only on D and Γ. $\square$

This shows that $\mathrm{MCSP}(\Gamma)$ is at least as hard as $\mathrm{MCSP}(\{\mathrm{Neq}_{D'}\})$ where D' is the domain of $Core(\Gamma)$. One might now wonder exactly how powerful the relation Neq_D is. Due to space constraints, we omit the proof, but it is in fact not difficult to see that whenever we have access to this appearingly simple relation, then $\mathrm{MCSP}(\{\mathrm{Neq}_D\})$ is as hard as $\mathrm{MCSP}(\mathrm{Rel}_D)$.

Theorem 15. *Let D be a finite domain. Then* $\mathrm{MCSP}(\mathrm{Rel}_D) \leq_{\mathrm{CV}} \mathrm{MCSP}(\{\mathrm{Neq}_D\})$.

Put together, Theorems 14 and 15 imply that $\mathrm{MCSP}(\Gamma)$ is always CV-reducible to $\mathrm{MCSP}(\{\mathrm{End}(\Gamma)(D)\})$, which results in the following corollary. The proof is straightforward and therefore omitted.

Corollary 16. *Let Γ and Δ be two finite constraint languages over D, with core-size c and d, respectively. If $d \leq c$ then* $\mathrm{MCSP}(\Delta) \leq_{\mathrm{CV}} \mathrm{MCSP}(\Gamma)$.

4 Upper and Lower Bounds for the Complexity of Monotone Constraint Satisfaction

With Corollary 16 in Section 3 we now have a powerful condition for verifying whether MCSP(Γ) is CV-reducible to MCSP(Δ). Moreover, since MCSP(Γ) is solvable in $O(d^{|V|} \cdot poly(||I||))$, where d is the core-size of Γ, we have an obvious upper bound on the complexity of MCSP(Γ) for all finite constraint languages. Proving lower bounds, i.e. the problem of proving that a problem is not solvable in $O(c^{\delta|V|})$ time for any $\delta < 1$, is much more challenging and usually requires stronger complexity theoretical assumptions than P $\neq$ NP. Let SAT denote the Boolean satisfiability problem where instances are given as a tuple (V, ϕ), where V is a set of variables and ϕ a conjunctive formula where each clause is a disjunction of positive and negative literals over V. The *strong exponential-time hypothesis* (SETH) is the conjecture that SAT is not solvable in $O(2^{\delta|V|})$ time for any $\delta < 1$ [4,7]. Using the SETH we can not only prove that MCSP(Rel_D) is not solvable in $O(2^{\delta|V|})$ time for any $\delta < 1$, unless the SETH fails, but that MCSP(Rel_D) is not solvable in $O(|D|^{\delta|V|})$ for any $\delta < 1$.

Theorem 17. *Let D be a finite domain. If the SETH holds then* MCSP(Rel_D) *is not solvable in $O(|D|^{\delta|V|})$ time for any $\delta < 1$.*

Proof. Assume that MCSP(Rel_D) is solvable in $O(|D|^{\delta|V|})$ time for some $\delta < 1$. Let $I = (V, \phi)$ be an instance of SAT over the variables $V = \{x_1, \ldots, x_n\}$ and the formula ϕ. Let $K \geq 1$ and $L = \lceil \frac{K}{\log_2(|D|)} \rceil$. The exact value of K, which is a constant depending on δ and D, will be determined later. Assume without loss of generality that $n \equiv 0 \pmod{K}$. We will partition V into subsets of size K and show that every such subset can be represented by L variables over D. Hence, let $V_1, \ldots, V_{\frac{n}{K}}$ be such a partition of V. Let $f : \{1, \ldots, n\} \rightarrow \{1, \ldots, \frac{n}{K}\}$ be a function satisyfing $f(i) = j$ if and only if $x_i \in V_j$. For every V_i introduce L fresh variables $y_{i_1}, \ldots, y_{i_L}$, and observe that we in total require $\frac{n \cdot L}{K}$ new variables. Let $h : \{x_1, \ldots, x_n\} \rightarrow \{1, \ldots, K\}$ be a function which is injective on every V_i, i.e., every variable in a subset V_i of V is assigned a unique index from 1 to K.

Let b_D be an injective function from $\{0,1\}^K$ to D^L. Such a function exists since by definition $2^K \leq |D|^L$. The purpose of b_D is to convert a K-ary Boolean sequence to an L-ary tuple over D. For each $i \in \{1, \ldots, L\}$ define the L-ary relation $R_i^+ = \{b_D(x_1, \ldots, x_K) \mid (x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_K) \in \{0,1\}^K\}$, and the L-ary relation $R_i^- = \{b_D(x_1, \ldots, x_K) \mid (x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_K) \in \{0,1\}^K\}$.

Let $(\ell_{i_1} \vee \ldots \vee \ell_{i_k})$, $\ell_{i_j} = x_{i_j}$ or $\ell_{i_j} = \neg x_{i_j}$, be a clause in ϕ. Let $z_{i_1}, \ldots, z_{i_k}$ such that $z_{i_j} = +$ if $\ell_{i_j} = x_{i_j}$ and $z_{i_j} = -$ if $\ell_{i_j} = \neg x_{i_j}$. For each literal ℓ_{i_j} let $V_{f(i_j)}$ be the partition corresponding to x_{i_j} and let $y_{f(i_j)_1}, \ldots, y_{f(i_j)_L}$ be the corresponding variables over D. Then replace the clause $(\ell_{i_1} \vee \ldots \vee \ell_{i_k})$ with

$$R^{z_{i_1}}_{h(x_{i_1})}(y_{f(i_1)_1}, \ldots, y_{f(i_1)_L}) \vee \ldots \vee R^{z_{i_j}}_{h(x_{i_j})}(y_{f(i_j)_1}, \ldots, y_{f(i_j)_L}).$$

This reduction might appear to be complicated but essentially just follows the intuition that we can replace every variable set V_i with the corresponding variables over D. Let I' be the resulting instance of $\mathrm{MCSP}(\mathrm{Rel}_D)$ over $\frac{n \cdot L}{K}$ variables. It is now easy to verify that I' is satisfiable if and only if I is satisfiable.

Hence, SAT can be solved in $O(|D|^{\delta \cdot \frac{n}{K} \cdot L})$ time. Now observe that $O(|D|^{\delta \cdot \frac{n}{K} \cdot L}) \subseteq O(|D|^{\delta \cdot \frac{n}{K} \cdot (\frac{K}{\log_2(|D|)}+1)}) = O(|D|^{\delta \cdot (\frac{n}{\log_2(|D|)}+\frac{n}{K})}) = O((|D|^{\delta \cdot (n+\frac{n \cdot \log_2(|D|)}{K})})^{\frac{1}{\log_2(|D|)}}) = O(2^{\delta \cdot (n+\frac{n \cdot \log_2(|D|)}{K})}) = O(2^{n \cdot (\delta+\frac{\delta \cdot \log_2(|D|)}{K})})$. Since δ and $\log_2(|D|)$ are constants depending only on the domain D, SAT is solvable in $O(2^{n \cdot (\delta+\epsilon)})$ time for every $\epsilon > 0$, by choosing a large enough value of K. Hence, we can find an $\epsilon > 0$ such that SAT is solvable in $O(2^{n\delta'})$ for some $\delta' = \delta + \epsilon < 1$. This contradicts the SETH, and the result follows. □

Let Γ be a constraint language over D and let D' be the domain of $Core(\Gamma)$. By Theorem 15 we know that $\mathrm{MCSP}(\mathrm{Rel}_{D'})$ is CV-reducible to $\mathrm{MCSP}(\Gamma)$. Hence, we get the following corollary of Theorem 17.

Corollary 18. *Let Γ be a constraint language over a finite domain and let d be the core-size of Γ. If MCSP(Γ) is NP-complete then* $\mathrm{MCSP}(\Gamma)$ *is solvable in $O(d^{|V|})$ time but not in $O(d^{\delta|V|})$ for any $\delta < 1$, unless the SETH fails.*

We can also obtain lower bounds for $\mathrm{CSP}(\mathrm{Rel}_D)$ by using a similar proof strategy as in Theorem 17, but there is a caveat: relations in CSP instances are usually represented as list of tuples. Hence, if we replace a clause $(x_1 \vee \ldots \vee x_n)$ by its corresponding constraint, then the resulting instance might be exponentially larger than the original SAT instance, since a clause of the form $(x_1 \vee \ldots \vee x_n)$ can be compactly represented by a single tuple. However, we can obtain lower bounds for $\mathrm{CSP}(\mathrm{Rel}_D)$ vis-à-vis the complexity of $\mathrm{CSP}(\mathrm{Rel}_{\{1,2\}})$.

Corollary 19. *Assume that CSP*$(\mathrm{Rel}_{\{1,2\}})$ *is not solvable in $O(2^{\delta|V|})$ time for any $\delta < 1$. Then, for any finite domain D, CSP*(Rel_D) *is not solvable in $O(|D|^{\delta|V|})$ time for any $\delta < 1$.*

5 Concluding Remarks

In this paper we have obtained a complete classification of the worst-case time complexity of $\mathrm{MCSP}(\Gamma)$. Most of the proofs make heavy use of the Galois connection between strong partial endomorphism monoids and weak Krasner algebras without existential quantification. Obtaining similar results for the CSP problem would likely be extremely difficult since one in this case needs to consider partial functions of arbitrary high arity [9,11]. However, recent research in partial clone theory suggests that arity bounded sets of partial functions in many cases are expressive enough to characterize partial functions of arbitrary arity [12]. Hence, an interesting continuation of research would e.g. be to consider only binary or ternary partial functions, and investigate if such a restriction could be used increase our understanding of the worst-case time complexity of CSP and related problems.

Acknowledgments

The author is grateful towards Peter Jonsson for several helpful discussions regarding the content of this paper, and to the anonymous reviewers for many suggestions for improvement.

References

1. M. Bodirsky, M. Hermann, and F. Richoux. Complexity of existential positive first-order logic. In K. Ambos-Spies, B. Löwe, and W. Merkle, editors, *Mathematical Theory and Computational Practice*, volume 5635 of *Lecture Notes in Computer Science*, pages 31–36. Springer Berlin Heidelberg, 2009.
2. F. Börner. Basics of galois connections. In N. Creignou, P.G. Kolaitis, and H. Vollmer, editors, *Complexity of Constraints*, volume 5250 of *Lecture Notes in Computer Science*, pages 38–67. Springer Berlin Heidelberg, 2008.
3. A. Bulatov. A dichotomy theorem for constraint satisfaction problems on a 3-element set. *J. ACM*, 53(1):66–120, January 2006.
4. C. Calabro, R. Impagliazzo, and R. Paturi. The complexity of satisfiability of small depth circuits. In J. Chen and F. V. Fomin, editors, *Parameterized and Exact Computation*, volume 5917 of *Lecture Notes in Computer Science*, pages 75–85. Springer Berlin Heidelberg, 2009.
5. H. Fargier and P. Marquis. Disjunctive closures for knowledge compilation. *Artificial Intelligence*, 216(0):129 – 162, 2014.
6. M. Hermann and F. Richoux. On the computational complexity of monotone constraint satisfaction problems. In Sandip Das and Ryuhei Uehara, editors, *WALCOM: Algorithms and Computation*, volume 5431 of *Lecture Notes in Computer Science*, pages 286–297. Springer Berlin Heidelberg, 2009.
7. R. Impagliazzo and R. Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367 – 375, 2001.
8. P. Jeavons. On the algebraic structure of combinatorial problems. *Theoretical Computer Science*, 200:185–204, 1998.
9. P. Jonsson, V. Lagerkvist, G. Nordh, and B. Zanuttini. Complexity of SAT problems, clone theory and the exponential time hypothesis. In *In Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-2013)*, pages 1264–1277, 2013.
10. P. Jonsson, V. Lagerkvist, J. Schmidt, and H. Uppman. Relating the time complexity of optimization problems in light of the exponential-time hypothesis. In *Proceedings of the 39th International Symposium on Mathematical Foundations of Computer Science*, MFCS'14, pages 408–419, Berlin, Heidelberg, 2014. Springer-Verlag.
11. V. Lagerkist and M. Wahlström. Polynomially closed co-clones. In *Proceedings of the 44th International Symposium on Multiple-Valued Logic (ISMVL-2014)*, pages 85 – 90, 2014.
12. V. Lagerkist, M. Wahlström, and B. Zanuttini. Bounded bases of strong partial clones. In *Proceedings of the 45th International Symposium on Multiple-Valued Logic (ISMVL-2015)*, 2015. To appear.
13. T. Schaefer. The complexity of satisfiability problems. In *In Proceedings of the 10th Annual ACM Symposium on Theory Of Computing (STOC-78)*, pages 216–226. ACM Press, 1978.