A Survey on the Fine-grained Complexity of Constraint Satisfaction Problems Based on Partial Polymorphisms

Dedicated to the memory of Professor Ivo Rosenberg

Miguel Couceiro*[1], Lucien Haddad†[2], and Victor Lagerkvist‡[3]

[1]Université de Lorraine, CNRS, Inria Nancy G.E., LORIA, F-54000 Nancy, France
[2]Department of Mathematics & Computer Science, Royal Military College of Canada, Kingston, Ontario, Canada
[3]Department of Computer and Information Science, Linköping University, Linköping, Sweden

Abstract

Constraint satisfaction problems (CSPs) are combinatorial problems with strong ties to universal algebra and clone theory. The recently proved CSP dichotomy theorem states that each finite-domain CSP is either solvable in polynomial time, or that it is NP-complete. However, among the intractable CSPs there is a seemingly large variance in how fast they can be solved by exponential-time algorithms, which cannot be explained by the classical algebraic approach based on polymorphisms. In this contribution we will survey an alternative approach based on partial polymorphisms, which is useful for studying the fine-grained complexity of NP-complete CSPs. Moreover, we will state and discuss some challenging open problems in this research field.

1 Algebraic Background

We begin by providing a self-contained introduction to the underlying algebraic approach. The reader familiar with universal algebra and clone theory can safely skim the two following subsections.

*miguel.couceiro@{loria,Inria}.fr
†haddad-l@rmc.ca
‡victor.lagerkvist@liu.se

1.1 Partial Operations

Let $k \geq 2$ be an integer and let $\mathbf{k}$ denote a k-element set. Without loss of generality we assume that $\mathbf{k} := \{0, \ldots, k-1\}$. For a positive integer n, an *n-ary partial operation* on $\mathbf{k}$ is a map $f\colon \mathrm{dom}(f) \to \mathbf{k}$ where $\mathrm{dom}(f)$ is a subset of $\mathbf{k}^n$, called the *domain* of f. Let $\mathrm{Par}^{(n)}(\mathbf{k})$ denote the set of all n-ary partial operations on $\mathbf{k}$ and let

$$\mathrm{Par}(\mathbf{k}) := \bigcup_{n \geq 1} \mathrm{Par}^{(n)}(\mathbf{k}).$$

An n-ary partial operation g is said to be a *total operation* if $\mathrm{dom}(g) = \mathbf{k}^n$, and we let $\mathrm{Op}^{(n)}(\mathbf{k})$ be the set of all n-ary total operations on $\mathbf{k}$ and $\mathrm{Op}(\mathbf{k}) := \bigcup_{n \geq 1} \mathrm{Op}^{(n)}(\mathbf{k})$. For every positive integer n and each $1 \leq i \leq n$, let e_i^n denote the *n-ary i-th projection* defined by $e_i^n(a_1, \ldots, a_n) = a_i$ for all $(a_1, \ldots, a_n) \in \mathbf{k}^n$. Furthermore, let $J_{\mathbf{k}} := \{e_i^n \mid 1 \leq i \leq n, n \in \mathbb{N} \setminus \{0\}\}$ be the set of all (total) projections. Partial operations on $\mathbf{k}$ are composed in a natural way. For additional details we refer the reader to Lau [49].

Definition 1. *A clone is a composition closed subset of* $\mathrm{Op}(\mathbf{k})$ *containing $J_{\mathbf{k}}$, and a partial clone on $\mathbf{k}$ is a composition closed subset of* $\mathrm{Par}(\mathbf{k})$ *containing $J_{\mathbf{k}}$. A partial clone is said to be strong if it is closed under taking suboperations*[1].

It is well known that a partial clone C is strong if and only if $\mathrm{Str}(J_{\mathbf{k}}) \subseteq C$ (see, e.g., Lemma 2.11 in Haddad and Börner [11]). The set of (partial) clones on $\mathbf{k}$ forms a lattice $\mathcal{L}_{\mathrm{Op}(\mathbf{k})}$ ($\mathcal{L}_{\mathrm{Par}(\mathbf{k})}$) under inclusion, in which the infimum is the set-theoretical intersection. It is then known that the cardinality of $\mathcal{L}_{\mathrm{Op}(\mathbf{k})}$ ($\mathcal{L}_{\mathrm{Par}(\mathbf{k})}$) equals the continuum for $k \geq 3$ ($k \geq 2$), but that $\mathcal{L}_{\mathrm{Str}(\mathrm{Op}(\mathbf{2}))}$, *Post's lattice*, is countably infinite [52]. Similarly, the set of strong partial clones on $\mathbf{k}$ also forms a lattice $\mathcal{L}_{\mathrm{Str}(\mathrm{Par}(\mathbf{k}))}$, which is a sublattice of $\mathcal{L}_{\mathrm{Par}(\mathbf{k})}$ whose cardinality also equals the continuum for each $k \geq 2$. By definition, J_k and $\mathrm{Str}(J_k)$ are the least elements of $\mathcal{L}_{\mathrm{Par}(\mathbf{k})}$ and $\mathcal{L}_{\mathrm{Str}(\mathrm{Par}(\mathbf{k}))}$, respectively. For further background see, e.g., [11, 21, 23]. For $F \subseteq \mathrm{Par}(\mathbf{k})$, let $[F]_s$ denote the intersection of all strong partial clones on $\mathbf{k}$ containing F. Similarly, for $F \subseteq \mathrm{Op}(\mathbf{k})$, let $[F]$ be the intersection of all clones on $\mathbf{k}$ containing F, and in both cases we write $[f]$ or $[f]_s$ when $F = \{f\}$ is singleton. Say that a strong partial clone C over $\mathbf{k}$ is *finitely generated* if there exists a finite set $F \subseteq \mathrm{Par}(\mathbf{k})$ such that $[F]_s = C$, and is said to be *infinitely generated otherwise*.

[1] For $f, g \in \mathrm{Par}(\mathbf{k})$, g is a *suboperation of* f, $g \leq f$, if $g = f|_{\mathrm{dom}(g)}$. We denote the closure of $F \subseteq \mathrm{Par}(\mathbf{k})$ under taking suboperations by $\mathrm{Str}(F)$.

1.2 Relations

An h-ary *relation* R over $\mathbf{k}$ is a subset of $\mathbf{k}^h$, and we write $\mathrm{ar}(R) = h$ to denote its arity, and $\mathrm{Rel}_{\mathbf{k}}$ for the set of all relations over $\mathbf{k}$. It is well known that strong partial clones are exactly those partial clones that are *determined by relations* in the following way. Let $h, n \geq 1$ be integers, and let R be an h-ary relation on $\mathbf{k}$. An n-ary partial operation f on $\mathbf{k}$ is said to *preserve* R if for every $h \times n$ matrix $M = [M_{ij}]$ whose columns $M_{*j} \in R$, and whose rows $M_{i*} \in \mathrm{dom}(f)$, the h-tuple $(f(M_{1*}), \ldots, f(M_{h*})) \in R$. Note that if there is no $h \times n$ matrix $M = [M_{ij}]$ whose columns $M_{*j} \in R$ and whose rows $M_{i*} \in \mathrm{dom}(f)$, then f preserves R. It is not difficult to see that

$$\mathrm{pPol}(R) := \{f \in \mathrm{Par}(\mathbf{k}) \mid f \text{ preserves } R\}$$

is a strong partial clone, called the *partial clone determined by the relation* R. Similarly, if Γ is a set of relations over $\mathbf{k}$ we write $\mathrm{pPol}(\Gamma)$ for the set $\bigcap_{R \in \Gamma} \mathrm{pPol}(R)$. In the total case we similarly write $\mathrm{Pol}(R)$ for the set of total polymorphisms of R and $\mathrm{Pol}(\Gamma)$ if Γ is a set of relations.

The fact that (strong partial) clones can be defined exclusively via relations suggests a deeper relationship between operations and relations. In fact, for each clone $\mathrm{Pol}(\Gamma)$ (respectively, strong partial clone $\mathrm{pPol}(\Gamma)$) there exists a corresponding set of relations that can be defined through Γ by a suitable closure operator. First, say that an n-ary relation R has a *primitive positive definition* (pp-definition) over $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ if R is the set of models of a first-order formula (with equality) $\varphi(x_1, \ldots, x_n)$ consisting only of existential quantification and conjunction over positive atoms from Γ. In symbols we denote such a definition by

$$R(x_1, \ldots, x_n) \equiv \varphi(x_1, \ldots, x_n),$$

where $\varphi(x_1, \ldots, x_n)$ is of the form

$$\exists x_{n+1}, \ldots, x_{n+n'} : R_1(\mathbf{x}_1) \wedge \ldots \wedge R_m(\mathbf{x}_m)$$

and where each $\mathbf{x}_i$ is a tuple of variables over $x_1, \ldots, x_{n+n'}$, and each $R_i \in \Gamma \cup \{(x, x) \mid x \in \mathbf{k}\}$. In addition, we say that R has a *quantifier-free primitive positive definition* (qfpp-definition) over Γ if R has a pp-definition over Γ where $n' = 0$, i.e., a pp-definition without any existentially quantified variables. These two definitions naturally induce two closure operators over relations, in the following sense.

Definition 2. *A set* $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ *is said to be a* relational clone*, or a* co-clone*, if*

1) $R \in \Gamma$ *for each* R *pp-definable over* Γ*, and*
2) $\emptyset \in \Gamma$.

Similarly, a set $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ *is called a* weak co-clone*, or a* weak system*, if*

1) $R \in \Gamma$ *for each* R *qfpp-definable over* Γ*, and*
2) $\emptyset \in \Gamma$.

1.3 Galois Connections

Clones and strong partial clones are related to co-clones and weak systems, respectively, in following way. The first important observation is that the set $\mathrm{Inv}(F)$ of all relations preserved by each (partial) operation in $F \subseteq \mathrm{Par}(\mathbf{k})$ is (1) a co-clone if each operation in F is total, and (2) a weak system otherwise. Moreover, it is well known that $\mathrm{Inv}(\mathrm{Pol}(\Gamma))$, (respectively $\mathrm{Inv}(\mathrm{pPol}(\Gamma))$) is the smallest co-clone (respectively weak system) over $\mathbf{k}$ containing Γ. Thus, the operators $\mathrm{Inv}(\cdot)$ and $\mathrm{Pol}(\cdot)$ constitute a *Galois connection* between clones and co-clones, whereas $\mathrm{Inv}(\cdot)$ and $\mathrm{pPol}(\cdot)$ constitute a Galois connection between strong partial clones and weak systems.

Theorem 3. *[9, 10, 34, 54] Let* $\Gamma, \Delta \subseteq \mathrm{Rel}_k$ *be two sets of relations. Then (1)* $\Gamma \subseteq \mathrm{Inv}(\mathrm{Pol}(\Delta))$ *if and only if* $\mathrm{Pol}(\Delta) \subseteq \mathrm{Pol}(\Gamma)$*, and (2)* $\Gamma \subseteq \mathrm{Inv}(\mathrm{pPol}(\Gamma))$ *if and only if* $\mathrm{pPol}(\Delta) \subseteq \mathrm{pPol}(\Gamma)$*.*

One practical consequence of Theorem 3 is that properties of clones can be translated into properties of co-clones, and vice versa. Moreover, due to the antitone nature of Galois connections, one of these viewpoints may be substantially simpler than the other one.

2 Constraint Satisfaction Problems

In a *constraint satisfaction problem* (CSP) the objective is to assign values to variables subjected to a set of constraints deciding admissible assignments. Typically, a CSP is formulated as the decision problem of determining whether there exists an assignment respecting all constraints. For the sake of self-containment, we follow the predominant definition of CSPs in computer science literature [55].

Definition 4. *A* constraint satisfaction problem *(CSP) over a set* $\mathbf{k}$ *is defined as a decision problem of the following form.*
Instance: *A tuple* (V, C) *where* V *is a finite set, and* C *a finite set of the form* (R_i, t_i)*,* $i \in I$*, where* $R_i \in \mathrm{Rel}_{\mathbf{k}}$ *and* $t_i \in V^{\mathrm{ar}(R_i)}$*.*
Question: *Is there a function* $f\colon V \to \mathbf{k}$ *such that* $(f(x_i^1), \ldots, f(x_i^{\mathrm{ar}(R_i)})) \in R_i$ *for each* $(R_i, (x_i^1, \ldots, x_i^{\mathrm{ar}(R_i)})) \in C$*?*

The set $\mathbf{k}$ *is called the* domain *of the CSP (not to be confused with the domain of a partial operation). If* $k = 2$*, then* $\mathbf{k}$ *is said to be* Boolean*. The members of* V *are referred to as* variables *and are usually* *denoted by x, v, or, if necessary, by using suitable subscripts. A tuple $(R_i, t_i) \in C$ is called a* constraint*, and we typically write $R(t_i)$ instead of (R_i, t_i). The function f, if it exists, is called a* solution*, a* model*, or a* satisfying assignment*.*

CSPs can be further specified by fixing a set of relations Γ, called a *constraint language*. This class of problems is then denoted by $\mathrm{CSP}(\Gamma)$ and it is restricted to instances (V, C) where $R_i \in \Gamma$ for each constraint $(R_i, t_i) \in C$. If Γ contains only Boolean relations (and thus $k = 2$), then $\mathrm{CSP}(\Gamma)$ can be viewed as a class of *satisfiability* problems, and it is usually denoted by $\mathrm{SAT}(\Gamma)$. Note that we have not yet specified how instances of $\mathrm{CSP}(\Gamma)$ are represented. If Γ is finite then the particular representation is not important, but if Γ is infinite the precise representation may become relevant. Here, we take a simple approach and assume that each relation is represented by a list of tuples. This is certainly not the only possible choice, and there exist languages where this representation scheme can be exponentially larger than a simpler encoding. For example, the relation corresponding to a clause $(x_1 \vee \ldots \vee x_n)$ of length $n \geq 1$ can naively be represented as a list of $2^n - 1$ tuples, but can succinctly be represented by a single tuple encoding the forbidden truth assignment.

Observe that if we associate a constraint language Γ over a domain D to a relational signature τ, then Γ can be thought of as a relational structure Γ^τ. In this way, an instance $(\{v_1, \ldots, v_n\}, C)$ of $\mathrm{CSP}(\Gamma)$ can be viewed as an existentially quantified τ-formula

$$\exists v_1, \ldots, v_n \colon \bigwedge_{(R_i, t_i) \in C} R_i(t_i),$$

and the question is then whether this τ-formula has a model.

It is also possible to reformulate $\mathrm{CSP}(\Gamma)$ as a homomorphism problem since an instance I of $\mathrm{CSP}(\Gamma)$ can be seen as a τ-structure $\mathcal{I}$, and where the question is then to decide whether there exists a homomorphism between $\mathcal{I}$ and Γ^τ.

Example 1. *Let $R_{1/3} = \{(0,0,1), (0,1,0), (1,0,0)\}$. Then $\mathrm{SAT}(\{R_{1/3}\})$ can be seen as an alternative formulation of the monotone 1-in-3-SAT problem which is well-known to be NP-complete. By choosing a suitable Boolean Γ, a large range of satisfiability problems can be represented as a $\mathrm{CSP}(\Gamma)$ problem. For example, for each $k \geq 1$ let Γ^k_{SAT} be the set of relations of the form $\{0,1\}^k \setminus \{t\}$ for a single k-ary Boolean tuple t. Then $\mathrm{SAT}(\Gamma^k_{\mathrm{SAT}})$ can be verified to be an alternative formulation of k-SAT which is NP-complete for $k \geq 3$. It may also be interesting to note that if we let $\Gamma_{\mathrm{SAT}} = \bigcup_{k \geq 1} \Gamma^k_{\mathrm{SAT}}$ then the only difference between $\mathrm{SAT}(\Gamma_{\mathrm{SAT}})$ and the satisfiability problem in conjunctive normal form,* CNF-SAT*, is the preferred representation scheme, since a clause in* *the latter problem is traditionally represented by a single falsifying truth assignment, rather than by the list of all satisfying truth assignments.*

Example 2. *Let us also consider a few non-Boolean examples. One of the prototypical examples of a CSP is the k-*COLOURING *problem: given an undirected graph (V, E), can (V, E) be coloured using at most k colours? To formulate this problem as a CSP we take the relation $R_{\neq_k} = \{(x, y) \in \mathbf{k}^2 \mid x \neq y\}$ and for each $(x, y) \in E$ introduce a constraint $R_{\neq_k}(x, y)$. It is also easy to find examples of tractable CSPs, i.e., CSPs solvable in polynomial time. One such example is systems of linear equations $x_1 + \ldots + x_n = 0 \,(\mathrm{mod}\, k)$ which can be solved in polynomial time using Gaussian elimination. As we will see in Section 3 this discrepancy in complexity between tractable and NP-complete CSPs can be explained using algebraic methods.*

Although this survey mainly focuses on finite-domain CSPs, a substantial amount of research is dedicated towards infinite-domain CSPs. This is especially true in artificial intelligence where many classical problems are intrinsically linked to constraints over infinite domains. Some examples include spatial and temporal reasoning problems such as ALLEN'S INTERVAL ALGEBRA, THE REGION CONNECTION CALCULUS, and the RECTANGLE ALGEBRA (cf. the surveys [7, 32]).

3 Polymorphisms and the Complexity of CSPs

Feder & Vardi conjectured that $\mathrm{CSP}(\Gamma)$ is either tractable or NP-complete [33]. This conjecture is usually referred to as the CSP *dichotomy conjecture*. It was then realized that several classical algorithms that run in polynomial time, e.g., Gaussian elimination and k-consistency, in a uniform manner could be explained by the presence of certain polymorphisms of Γ [38]. More generally, Jeavons proved the following reducibility result, usually interpreted as "the polymorphisms of Γ determine the complexity of $\mathrm{CSP}(\Gamma)$ up to polynomial time reductions".

Theorem 5 ([37])**.** *Let Γ and Δ be two finite constraint languages over $\mathbf{k}$. If $\mathrm{Pol}(\Gamma) \subseteq \mathrm{Pol}(\Delta)$, then $\mathrm{CSP}(\Delta)$ is polynomial-time many-one reducible to $\mathrm{CSP}(\Gamma)$.*

Proof. From Theorem 3 the condition $\mathrm{Pol}(\Gamma) \subseteq \mathrm{Pol}(\Delta)$ is equivalent to the condition that $\Delta \subseteq \mathrm{Inv}(\mathrm{Pol}(\Gamma))$. Hence, each relation in Δ is pp-definable over Γ. The reduction from $\mathrm{CSP}(\Delta)$ to $\mathrm{CSP}(\Gamma)$ then follows as a classical "gadget reduction" where each constraint in an instance (V, C) of $\mathrm{CSP}(\Delta)$ is replaced by the set of constraints corresponding to a pp-definition over Γ, and any two variables occuring in an equality constraint are identified. This can be accomplished in polynomial time with respect to $|C|$ and $|V|$ since

1) each pp-definition of $R \in \Delta$ can be precomputed and stored in a table whose size depends only on the two finite sets Γ and Δ,
2) the identification of variables is a special case of ST-CONNECTIVITY which is solvable using only logarithmic space [53], and
3) Δ is finite and thus $|C|$ is polynomially bounded in $|V|$.

The proof can then be completed by observing that the resulting instance of $\mathrm{CSP}(\Gamma)$ may contain up to $|C| \cdot m$ fresh variables, for a constant m depending on Γ and Δ, since existentially quantified variables in pp-definitions correspond to the introduction of fresh variables. $\square$

Now, to obtain a dichotomy for $\mathrm{CSP}(\Gamma)$ over $\mathbf{k}$ one would, in principle, need to consider all operations over $\mathbf{k}$ and to determine which combinations of operations that result in tractable CSPs. However, such a process turns out to be unecessary, since the classical complexity of $\mathrm{CSP}(\Gamma)$ only depends on the *identities* or the *strong Maltsev conditions*, satisfied by the polymorphisms of Γ [18, 2]. In technical terms, this means that the complexity of $\mathrm{CSP}(\Gamma)$ depends only on the *variety* to which $\mathrm{Pol}(\Gamma)$ belongs to. For example, if $\mathrm{Pol}(\Gamma)$ contains a *Maltsev operation* satisfying the identities $m(x, x, y) \approx y, m(x, y, y) \approx x$, then $\mathrm{CSP}(\Gamma)$ is tractable since it can be solved by the simple algorithm for Maltsev constraints [17]. The main advantage of this observation is that it suffices to describe all identities resulting in tractable CSPs rather than all concrete operations. This approach recently culminated in the following dichotomy theorem.

Theorem 6 ([16, 63])**.** *Let Γ be a constraint language over $\mathbf{k}$. Then $\mathrm{CSP}(\Gamma)$ is either tractable or NP-complete.*

Although simple to state, Theorem 6 is the result of decades of intense research, and was known to hold for several important, special cases [56, 13, 14] before the main result could be proven. For additional details concerning the classification project of CSP and the algebraic approach based on strong Maltsev conditions, see e.g. the survey by Barto [1].

4 Partial Polymorphisms and the Fine-Grained Complexity of CSPs

We begin this section by outlining how partial polymorphisms can be useful for (classical) complexity classifications where the standard algebraic approach based on polymorphisms falls short. We then discuss the rather vague term "fine-grained complexity" in relationship to CSPs in Section 4.2, and in Section 4.3 describe how the algebraic approach based on partial polymorphisms can be used to study this question in greater detail.

4.1 Weak Bases and Classical Complexity

Before we describe how partial polymorphisms can be used to study the fine-grained complexity of CSPs, we take a slight detour in order to outline a related application, which preceded fine-grained complexity in time. To understand this motivation it is important to realise that many CSP-related problems have been classified during the last decades as well, and in almost all cases using a very similar algebraic toolbox. Some promiment examples are *counting CSPs* [15, 27], *min-ones* [41], and *propositional abduction* [30]. For further details and additional examples, see e.g. the survey by Creignou & Vollmer [29].

In short, such complexity dichotomies are usually proved by first establishing a counter part to Theorem 5, and for a set of operations F either (1) prove that $\mathrm{Inv}(F)$ results in a tractable problem, or (2) show that there exists $\Gamma \subseteq \mathrm{Inv}(F)$ resulting in an intractable problem (typically NP-hard or co-NP-hard). Hence, instead of considering arbitrary constraint languages we for each clone only have to consider a fixed constraint language. Informally, this strategy works for all problems parameterised by constraint languages where the introduction of fresh variables (stemming from existentially quantified variables in pp-definitions) does not affect the existence of a solution. However, *what if this is not the case?* This question motivated Schnoor & Schnoor [58] to investigate a connection between partial polymorphisms and the complexity of CSP-related problems which had been difficult to classify by existing tools.

Example 3. $\mathrm{CSP}(\Gamma)$ *is sometimes said to be* a priori *compatible with polymorphisms due to Theorem 5. In contrast, there exist problems proven to be* a posteriori *compatible with polymorphisms, in the sense that* $\mathrm{Pol}(\Gamma)$ *determines whether the problem is tractable or intractable, but where an analogue of Theorem 5 cannot be proven. One such example is the problem of finding a surjective model of a* $\mathrm{SAT}(\Gamma)$ *instance (*$\mathrm{SUR\text{-}SAT}(\Gamma)$*), which is NP-complete if* $\mathrm{Pol}(\Gamma)$ *is essentially unary*[2] *and tractable otherwise [28]. Curiously, almost all CSP-like problems studied in the literature turn out to be either a priori or a posteriori compatible with polymorphisms, and only a handful of concrete counter examples exist, e.g., enumerating models of* $\mathrm{CSP}(\Gamma)$ *with polynomial delay [57], the inverse satisfiability problem over infinite constraint languages [44], and the maximum satisfiability problem [26].*

[2] A clone C is essentially unary if $C = [F]$ for a set of unary operations F.

Problems that are not a priori compatible with polymorphisms may instead be compatible with partial polymorphisms. It is, for example, straightforward to prove that if $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(\Delta)$ then SUR-SAT(Δ) is polynomial-time many-one reducible to SUR-SAT(Γ). Unfortunately, the usefulness of this observation remains limited because the lattice of Boolean strong partial clones $\mathcal{L}_{\mathrm{Str}(\mathrm{Op}(\mathbf{2}))}$ is still not fully known. However, Schnoor & Schnoor [58] realized that for many classification purposes, there is typically no need to consider the whole lattice $\mathcal{L}_{\mathrm{Str}(\mathrm{Op}(\mathbf{2}))}$, but only a small fragment corresponding to *weak bases.*

Definition 7. *[58] Let $C = \mathrm{Pol}(\Gamma)$ be a clone over $\mathbf{k}$ where Γ is finite. A set of relations $\Gamma_w \subseteq \mathrm{Rel}_{\mathbf{k}}$ is said to be a* weak base *of* $\mathrm{Inv}(C)$ *if (1)* $\mathrm{Pol}(\Gamma_w) = C$ *and (2)* $\mathrm{pPol}(\Delta) \subseteq \mathrm{pPol}(\Gamma_w)$ *for each set* $\Delta \subseteq \mathrm{Rel}_{\mathbf{k}}$ *such that* $\mathrm{Pol}(\Delta) = C$.

Example 4. *Let us again consider* SUR-SAT(Γ) *and assume that we are given a weak base Γ_w of a co-clone* $\mathrm{Inv}(C)$*. If we can prove that* SUR-SAT(Γ_w) *is NP-complete, then NP-completeness also carries over to every Γ such that* $\mathrm{Pol}(\Gamma) = C$*. Hence, equipped with a weak base of each Boolean co-clone, we in practice only need to consider Post's lattice [52] rather than* $\mathcal{L}_{\mathrm{Str}(\mathrm{Op}(\mathbf{2}))}$.

Schnoor & Schnoor [58] also described a procedure for constructing weak bases for co-clones satisfying the preconditions in Definition 7, which was leveraged by Lagerkvist to provide a list of weak bases for all Boolean co-clones [42], whose inclusion structure was later completely determined [45]. We will not describe the method for constructing weak bases in detail, but remark that it is based on the observation that the algebra whose universe consists of all n-ary operations in C can be viewed as a relation R, with the property that any partial operation not preserving R can be extended to a total operation. This construction has been referred to as the *n-generated free algebra* [1], or the *n-th graphic* [51]. Using a similar strategy to that used in Example 4, weak bases have been used to obtain complexity dichotomies for several classes of Boolean CSP-like problems incompatible with polymorphisms [3, 4, 44, 58, 59].

Example 5. *Behrisch et al. [5] considered several problems, e.g.,* nearest solution *(*NSOL*),* nearest other solution *(*NOSOL*), and* minimum solution distance *(*MSD*), all parameterised by Boolean constraint languages. The optimisation variants of these problems may be defined as follows.*

1) NSOL(Γ)*: given a* SAT(Γ) *instance I and a function $f\colon V \to \mathbf{2}$, compute a satisfying assignment to I with minimal Hamming distance from f.*

2) NOSol(Γ): *given a* SAT(Γ) *instance* I *and a satisfying assignment to* I*, compute a satisfying assignment to* I *with minimal Hamming distance from* f*.*

3) MSD(Γ): *given a* SAT(Γ) *instance* I*, compute two satisfying assignments to* I *with minimal Hamming distance.*

Among these problems only NSol *is a priori compatible with polymorphisms in the sense discussed in Example 3, but with a non-trivial reduction, while* NoSol *and* MSD *can be studied with partial polymorphisms via the weak bases approach. For instance, if* $\mathrm{Pol}(\Gamma) = [\{f_\neg\}]$*, where* $f_\neg(x) = 1 - x$*, then it is sufficient to show that* $\textsc{NoSol}(\{\mathrm{R}\})$ *does not admit a polynomial-time approximation scheme (unless P = NP) for the weak base* R *of* $\mathrm{Inv}(\{f_\neg\})$ *provided by Lagerkvist [42], instead of having to prove this for every possible choice of* $\Gamma \subseteq \mathrm{Rel}_{\mathbf{2}}$ *where* $\mathrm{Pol}(\Gamma) = [\{f_\neg\}]$.

4.2 Fine-Grained Complexity

Recall from Section 3 that polymorphisms are useful for studying the classical complexity of CSPs up to polynomial-time reductions. However, there are reasons to believe that, in practice, even NP-complete problems can exhibit a striking difference in complexity, and that it may be disadvantageous to group them together under the guise of polynomial-time reductions. For example, $\mathrm{SAT}(\{R_{1/3}\})$ from Example 1, is known to be solvable in $O(1.0984^n)$ time, where n denotes the number of variables [62], whereas it is not known whether CNF-SAT is solvable in $O(c^n)$ time for $c < 2$. This phenomena is not restricted to CSPs: for example, van Rooij *et al.* [8] proved that the Partition Into Triangles problem restricted to graphs of maximum degree 4 can be solved in $O(1.0222^n)$ time despite being NP-complete.

Our main concern in this survey paper is thus to study the complexity of NP-complete CSPs with regards to $O(c^n)$ time complexity. To make this question more precise we begin with the following definition.

Definition 8. *Let* $k \geq 2$*. For* $\Gamma \subseteq \mathrm{Rel}_k$*, set*

$$\mathsf{T}(\Gamma) = \inf\{c \mid \mathrm{CSP}(\Gamma) \textit{ is solvable in time } 2^{cn}\},$$

where n *is the number of variables in an instance of* CSP(Γ)*.*

Note that CSP(Γ) may be solvable in $O(2^{(c+\varepsilon)n})$ time for every $\varepsilon > 0$ despite not being solvable in $O(2^{cn})$ time, thus showing that the use of infimum in Definition 8 is necessary. If $\mathsf{T}(\Gamma) = 0$, then CSP(Γ) is said to be solvable in *subexponential time.* It is important to observe that no concrete value of $\mathsf{T}(\Gamma)$ is known when CSP(Γ) is NP-complete, but that a large number of upper bounds of the form $\mathsf{T}(\Gamma) \leq c$ are known for concrete constraint languages Γ. For example, as already mentioned, $\mathsf{T}(\{R_{1/3}\}) \leq \log_2(1.0984)$ since $\mathrm{SAT}(\{R_{1/3}\})$ is solvable in $O(1.0984^n)$ time, and if we take the relation $R_{\neq_k}$ from Example 2 then $\mathsf{T}(\{R_{\neq_2}\} = 0$ (since 2-COLOURING is in P), and for each $k \geq 3$ it is known that $\mathsf{T}(\{R_{\neq_k}\}) \leq 1$ [6].

To study the function T and its connection to partial polymorphisms, we will make use the following conjecture, which is of central importance in current research on fine-grained complexity and lower bounds.

Definition 9. *The* exponential-time hypothesis *(ETH) [35] conjectures that* $\mathsf{T}(\Gamma^3_{\mathrm{SAT}}) > 0$.

In other words, the ETH states that there exists a $c > 0$ such that 3-SAT is not solvable in $O(2^{cn})$ time, i.e., not in subexponential time. Although not immediate from Definition 9, the ETH is also known to imply that the sequence $\mathsf{T}(\Gamma^3_{\mathrm{SAT}}), \mathsf{T}(\Gamma^4_{\mathrm{SAT}}), \ldots,$ increases infinitely often, i.e., that for every k there exists $k' > k$ such that $\mathsf{T}(\Gamma^k_{\mathrm{SAT}}) < \mathsf{T}(\Gamma^{k'}_{\mathrm{SAT}})$ [35]. It is tempting to also conjecture that the limit of the sequence $\mathsf{T}(\Gamma^3_{\mathrm{SAT}}), \mathsf{T}(\Gamma^4_{\mathrm{SAT}}), \ldots$ equals 1. This conjecture is known as the *strong exponential-time hypothesis* (SETH) [19, 35]. Under this conjecture, the unrestricted SAT problem cannot be solved in $O(2^{cn})$ time for *any* $c < 1$.

The ETH and the SETH are important conjectures also when studying fine-grained complexity from an algebraic point of view, since they represent the best possible bounds that one should realistically aim for. This is similar to how one should not hope to achieve a polynomial-time algorithm for an NP-hard problems if P $\neq$ NP. Indeed, it is then known that we cannot achieve subexponential-time algorithms for NP-complete finite-domain CSPs without violating the ETH.

Theorem 10. *([40]) Let* $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ *such that* $\mathrm{CSP}(\Gamma)$ *is NP-complete. Then* $\mathsf{T}(\Gamma) > 0$ *unless the ETH fails.*

Subexponentiality can also be ruled out for certain classes of *structurally restricted* CSPs [31], but we refrain from formally stating these results since the current focus is on constraint language restrictions. Let us also remark that $\mathrm{CSP}(\Gamma)$ for $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ is always solvable in $O(k^n)$ time by simply enumerating all possible assignments over $\mathbf{k}$. Hence, $\mathsf{T}(\Gamma) \leq \log_2(k)$ for every $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$. It is also known that if $\Gamma \subset \mathrm{Rel}_{\mathbf{k}}$ is finite then $\mathrm{CSP}(\Gamma)$ is solvable in $O(c^n)$ time for some $c < k$ [61], implying that $\mathsf{T}(\Gamma) < \log_2(k)$.

4.3 An Algebraic Approach Based on Partial Polymorphisms

We are now ready to present the link between partial polymorphisms and the function T, which allows us to study the fine-grained complexity of CSPs using partial polymorphisms.

Theorem 11 ([39])**.** *Let* Γ *and* Δ *be two finite sets of relations over* $\mathbf{k}$*. If* $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(\Delta)$*, then* $\mathsf{T}(\Delta) \leq \mathsf{T}(\Gamma)$*.*

Proof. By Theorem 3, this result can be proved rather explicitly: given an instance (V, C) of $\mathrm{CSP}(\Delta)$ each constraint in C can be rewritten as a set of constraints over $\Gamma \cup \{\{(x, x) \mid x \in \mathbf{k}\}\}$ without introducing any fresh variables, and the same techniques that were used in the proof of Theorem 5 can then be employed to complete the reduction in polynomial time. Hence, Theorem 11 can be restated in a slightly stronger version without making use of the function T, but for our purposes the above statement is sufficient. Also note that Theorem 11 is valid even if $\mathrm{CSP}(\Gamma)$ and $\mathrm{CSP}(\Delta)$ are both solvable in polynomial time since in this case we have that $\mathsf{T}(\Gamma) = \mathsf{T}(\Delta) = 0$. $\square$

Now, let C be a clone such that $\mathrm{Pol}(\Gamma) = C$, and suppose that $\mathrm{CSP}(\Gamma)$ is NP-complete. Theorem 11 then offers an algebraic method to analyse $\mathsf{T}(\Gamma)$ by studying the properties of $\mathcal{I}_{\mathrm{Str}}(C) := \{\mathrm{pPol}(\Gamma) \mid \mathrm{Pol}(\Gamma) = C\}$. For example, if $\mathcal{I}_{\mathrm{Str}}(C)$ is finite, then the fine-grained complexity of $\mathrm{CSP}(\Gamma)$ would fall into a finite number of cases. Hence, as a rough approximation, we would like to know the cardinality of $\mathcal{I}_{\mathrm{Str}}(\mathrm{Pol}(\Gamma))$ when $\mathrm{CSP}(\Gamma)$ is NP-complete. A dichotomy has been proved for Boolean clones, with the surprising implication that these sets are always either finite or equal to the continuum.

Theorem 12 ([25, 60])**.** *Let* C *be a Boolean clone. Then* $\mathcal{I}_{\mathrm{Str}}(C)$ *is finite if*

$$C \supseteq \mathrm{Pol}(\{\{(0,1),(1,0)\},\{(0,1)\}\})$$

or

$$C \supseteq \mathrm{Pol}(\{\{(0,0),(0,1),(1,1)\},\{(0,1)\}\}),$$

and is of continuum cardinality otherwise.

By inspecting Post's lattice of Boolean clones [52] one can then verify that the finite cases of Theorem 12 only hold for 10 clones. Also, it is known that $\mathrm{SAT}(\Gamma)$ is NP-complete if and only if $\mathrm{Pol}(\Gamma) = [f_\neg]$ or $\mathrm{Pol}(\Gamma) = J_{\{0,1\}}$, where $f_\neg(x) = 1 - x$ [56], implying that $\mathcal{I}_{\mathrm{Str}}(\mathrm{Pol}(\Gamma))$ is of continuum cardinality, whenever $\mathrm{SAT}(\Gamma)$ is NP-complete.

However, the fact that $\mathcal{I}_{\mathrm{Str}}(\mathrm{Pol}(\Gamma))$ is of continuum cardinality in these cases says very little of their actual complexity, and it suggests that one needs a different technique that does not rely on a classification akin to Post's lattice. For certain classes of clones C we may immediately observe yet another striking difference between $\mathcal{I}_{\mathrm{Str}}(C)$ and Post's lattice.

Theorem 13 ([46])**.** *Let* $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ *be a finite set of relations such that* $\mathrm{Pol}(\Gamma)$ *is essentially unary. Then* $\mathrm{pPol}(\Gamma)$ *is infinitely generated.*

In particular, Theorem 13 holds when $\mathrm{SAT}(\Gamma)$ is NP-complete, which suggests that a full description of $\mathrm{pPol}(\Gamma)$ (that correlates to fine-grained complexity via Theorem 11) is a complicated task for finite constraint languages. To illustrate, let us for the moment concentrate on Boolean constraint languages Γ such that $\mathrm{Pol}(\Gamma) = J_{\{0,1\}}$, which subsume the examples 1-IN-3-SAT and k-SAT from Example 1. Even though the full description of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ does not seem realistic by Theorem 12 and Theorem 13, there are plenty of questions relevant to the study of the fine-grained complexity of $\mathrm{SAT}(\Gamma)$. To illustrate, we list two below.

- Does $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ admit a greatest element, and if this is the case, is it then possible to describe the maximal elements?
- Is it possible to describe the minimal strong partial clones of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ — provided they exist (note that a unique least element trivially exists, namely $\mathrm{Str}(J_{\{0,1\}})$)[3].

These questions are pertaining to the study of fine-grained complexity since, by Theorem 11, "small" members of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ correspond to SAT problems with high time complexity, and "large" members of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ correspond to SAT problems of low time complexity.

It is worth observing that one of these questions can be answered immediately, by making use of the concept of a weak base R of a co-clone $\mathrm{Inv}(C)$ recalled from Section 4.1. Indeed, $\mathrm{pPol}(R) \supseteq \mathrm{pPol}(\Gamma)$ for each $\mathrm{pPol}(\Gamma) \in \mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ implies that $\mathrm{pPol}(R)$ is the greatest element. Furthermore, $\mathrm{Inv}(J_{\{0,1\}})$ is known to admit a particularly simple weak base $R^{\neq\neq\neq 01}_{1/3} = \{(0,0,1,1,1,0,0,1),(0,1,0,1,0,1,0,1),(1,0,0,0,1,1,0,1)\}$ [42]. This observation was then leveraged by Jonsson et al. [39] to show that $\mathrm{SAT}(\{R^{\neq\neq\neq 01}_{1/3}\})$ constitutes the "easiest NP-complete SAT problem", in the following sense.

Theorem 14 ([39]). $\mathrm{SAT}(\{R^{\neq\neq\neq 01}_{1/3}\})$ *is NP-complete and* $\mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}\}) \leq \mathsf{T}(\Gamma)$ *for* any *Boolean constraint language* Γ *such that* $\mathrm{SAT}(\Gamma)$ *is NP-complete.*

Proof. We give a short sketch of the most important ideas. If $\mathrm{SAT}(\Gamma)$ is NP-complete then by Schaefer's dichotomy theorem [56] either $\mathrm{Pol}(\Gamma) = [f_{\neg}]$ or $\mathrm{Pol}(\Gamma) = J_{\{0,1\}}$. It is also known [42] that the relation $R = R^{\neq\neq\neq 01}_{1/3} \cup \{(f_{\neg}(t) \mid t \in R^{\neq\neq\neq 01}_{1/3})\}$ is a weak base of $\mathrm{Inv}(\{f_{\neg}\})$, and from Theorem 11 it then follows that $\mathsf{T}(\{R\}) \leq \mathsf{T}(\Gamma)$ or $\mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}\}) \leq \mathsf{T}(\Gamma)$, since $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(R)$ or $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(R^{\neq\neq\neq 01}_{1/3})$. Hence, it is sufficient to prove that $\mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}\}) \leq \mathsf{T}(\{R\})$, which can be accomplished by a polynomial-time many-one reduction only introducing

[3] We follow the standard terminology where minimal/maximal clones are those directly above/below the greatest/least element in the clone lattice.

a constant number of fresh variables (see Lemma 19 in Jonsson el al. [39]). □

Jonsson *et al.* [39] also conjectured that the strong partial clones between $\mathrm{pPol}(R_{1/3})$ and $\mathrm{pPol}(R^{\neq\neq\neq 01}_{1/3})$ had a simple structure consisting of only three elements $\mathrm{pPol}(R^{01}_{1/3})$, $\mathrm{pPol}(R^{\neq 01}_{1/3})$, $\mathrm{pPol}(R^{\neq\neq 01}_{1/3})$, such that

$$\mathrm{pPol}(R_{1/3}) \subset \mathrm{pPol}(R^{01}_{1/3}) \subset \mathrm{pPol}(R^{\neq 01}_{1/3}) \subset \mathrm{pPol}(R^{\neq\neq 01}_{1/3}) \subset \mathrm{pPol}(R^{\neq\neq\neq 01}_{1/3}).$$

However, this conjecture turned out to be incorrect: Lagerkvist & Roy showed the existence of (countably) infinitely many strong partial clones between $\mathrm{pPol}(R^{01}_{1/3})$ and $\mathrm{pPol}(R^{\neq 01}_{1/3})$, $\mathrm{pPol}(R^{\neq 01}_{1/3})$ and $\mathrm{pPol}(R^{\neq\neq 01}_{1/3})$, and $\mathrm{pPol}(R^{\neq\neq 01}_{1/3})$ and $\mathrm{pPol}(R^{\neq\neq\neq 01}_{1/3})$ [43]. This was later refined by Couceiro *et al.* [22] that constructed families of strong partial clones of continuum size between each of these pairs of partial clones.

It is also noteworthy to remark that Theorem 14 was extended to a broad class of finite-domain CSPs, the so-called *ultraconservative* CSPs, which can be defined as $\mathrm{CSP}(\Gamma)$ problems where Γ contains all unary relations over the domain. Here, the term ultraconservative is used instead of the more familiar "conservative" since it is actually required that the unary relations are included in the constraint language, and not only that they are primitive positive definable.

Theorem 15 ([40]). *For each* $\mathbf{k}$ *there exists a relation* $R_{\mathbf{k}} \in \mathrm{Rel}_{\mathbf{k}}$ *such that (1)* $\mathrm{CSP}(\{R_{\mathbf{k}}\})$ *is NP-complete, and (2)* $\mathsf{T}(\{R_{\mathbf{k}}\}) \leq \mathsf{T}(\Gamma)$ *for* any *ultraconservative* $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$ *such that* $\mathrm{CSP}(\Gamma)$ *is NP-complete.*

Proof. Assume that $\mathrm{CSP}(\Gamma)$ is NP-complete for an ultraconservative $\Gamma \subseteq \mathrm{Rel}_{\mathbf{k}}$. In this case almost nothing is known concerning the precise structure of $\mathrm{Pol}(\Gamma)$, making it difficult to construct a weak base of $\mathrm{InvPol}(\Gamma))$. However, it is known that Theorem 6 in this case implies that $\mathrm{Pol}(\Gamma)$ does not contain an operation satisfying a strong Maltsev condition, which in turn is known to imply that Γ *primitively positively interprets* (pp-interprets) $\Gamma^{3}_{\mathrm{SAT}}$. We refrain from defining pp-interpretations formally but remark that they may be viewed as a relational counterpart to varieties, and may be used to compare the expressive strength of constraint languages which are incomparable with respect to pp-definitions. From this assumption one can then prove that Γ can pp-define a relation R with only three tuples, and this pp-definition can be transformed into a qfpp-definition of a similar relation R_{Γ}, also of cardinality three. Among all these relations it is then possible to isolate a unique relation $R_{\mathbf{k}}$ with the property that $\mathsf{T}(\{R_{\mathbf{k}}\}) \leq \mathsf{T}(\{R_{\Gamma}\})$ for each ultraconservative Γ where $\mathrm{CSP}(\Gamma)$ is NP-complete. Hence, this proof strategy does not explicitly use weak bases, due to the largely unexplored clone lattice over $\mathbf{k}$, but it completely relies on qfpp-definitions. □

4.4 The Non-Existence of Minimal Strong Partial Clones

We now turn to the question of minimal strong partial clones in $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$, i.e., partial clones $\mathrm{pPol}(\Gamma) \in \mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ such that $\mathrm{pPol}(\Gamma) \supset \mathrm{Str}(J_{\{0,1\}})$ but for which there is no $\mathrm{pPol}(\Delta) \in \mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ such that $\mathrm{pPol}(\Gamma) \supset \mathrm{pPol}(\Delta) \supset \mathrm{Str}(J_{\{0,1\}})$. The existence of a minimal element $\mathrm{pPol}(\Gamma)$ would have interesting consequences in the light of the SETH, in particular, if $\mathsf{T}(\Gamma) < 1$, since $\mathrm{SAT}(\Gamma)$ would then belong to the class of "hardest" NP-complete SAT problems that are still easier than the unrestricted SAT problem.

However, this question has a surprisingly straightforward resolution: as proven by Couceiro et al. [24], there are no minimal strong partial clones. More specifically, for each $\mathbf{k}$ ($k > 1$) it was proved that if $f \notin \mathrm{Str}(J_{\mathbf{k}})$, then the strong partial clone $[f]_s$ contains a family of strong partial subclones of continuum cardinality. Two slightly different constructions were given to prove this result, depending on whether f is constant (i.e., there exists $x \in \mathbf{k}$ such that $f(\alpha_i) = x$ for all $\alpha_i \in \mathrm{dom}(f)$) or not. Here, we provide a sketch of the latter construction.

Let f be an n-ary partial operation not in $\mathrm{Str}(J_{\mathbf{k}})$ and not constant. In the sequel we assume that the partial operation f is n-ary and with domain $\mathrm{dom}(f) = \{\alpha^1, \dots, \alpha^m\} \subseteq \mathbf{k}^n$, where $\alpha^i := (a^i_1, \dots, a^i_n)$. Let A be the $m \times n$ matrix whose rows are $\alpha^1, \dots, \alpha^m$, and consider the following construction.

Let $\mathrm{Col}(A)$ be the set of columns of A, and $\mathbf{v}_f = f(A) \in \mathbf{k}^m$. For $\mathbf{x} := (x_1, \dots, x_h) \in \mathbf{k}^h$ and $\ell \geq 1$, let $\mathbf{x}^{\times \ell} \in \mathbf{k}^{h\ell}$ be

$$\mathbf{x}^{\times \ell} = (\underbrace{x_1, \dots, x_1}_{\ell \text{ times}}, \underbrace{x_2, \dots, x_2}_{\ell \text{ times}}, \dots, \underbrace{x_h, \dots, x_h}_{\ell \text{ times}}),$$

and let $[\mathbf{x}] = \{x_1, \dots, x_h\}$. For a set $X \subseteq \mathbf{k}$ with

$$X = \{x_1 < \cdots < x_{|X|}\}$$

and $a \in X$, let $\mathrm{next}_X(a) \in X$ be defined by

$$\mathrm{next}_X(a) := \begin{cases} x_{i+1} & \text{if } a = x_i \text{ and } i < |X|, \\ x_1 & \text{if } a = x_{|X|}. \end{cases}$$

Furthermore, for $\mathbf{x} = (x_1, \dots, x_h) \in [\mathbf{v}_f]^h$ and $1 \leq i \leq h$, let $c_i(\mathbf{x})$ be the tuple

$$c_i(\mathbf{x}) := (x_1, \dots, x_{i-1}, \mathrm{next}_{[\mathbf{v}_f]}(x_i), x_{i+1}, \dots, x_h).$$

Since the partial operation f is non-constant, the set $[\mathbf{v}_f]$ contains at least two different elements, and so $c_i(\mathbf{x}) \neq \mathbf{x}$ for all $\mathbf{x} \in [\mathbf{v}_f]^h$ and all

$i = 1, \dots, h$. Let $t \geq 0$ be the number of columns $\underset{\sim}{u}$ in the matrix A that satisfy $[\underset{\sim}{u}] = [\mathbf{v}_f]$. Without loss of generality, assume that all those t columns (if any) are the first columns to the left of A.

For each $\ell \geq 1$, define the relation ρ_ℓ of arity ℓd_f by

$$\rho_\ell := \{c_i(\mathbf{v}_f^{\times \ell}) \mid 1 \leq i \leq \ell d_f\} \cup \{\underset{\sim}{u}^{\times \ell} \mid \underset{\sim}{u} \in \mathrm{Col}(A)\}.$$

Notice that $|\rho_\ell| = \ell d_f + n$.

Let M_ℓ be the matrix with ℓd_f rows, whose $(\ell d_f + n)$ columns are the tuples of ρ_ℓ in the following order:

$$c_1(\mathbf{v}_f^{\times \ell}), \dots, c_{\ell d_f}(\mathbf{v}_f^{\times \ell}), \underset{\sim}{u}_1{}^{\times \ell}, \dots, \underset{\sim}{u}_n{}^{\times \ell},$$

where $\underset{\sim}{u}_1, \dots, \underset{\sim}{u}_n$ are the columns of A written in the same order as they appear in A.

Now let $f^{\times \ell}$ be the $(\ell d_f + n)$-ary partial operation whose domain is the set of all rows of M_ℓ and defined by

$$f^{\times \ell}(M_\ell) = \mathbf{v}_f^{\times \ell}.$$

Notice that $x_1, \dots, x_{\ell d_f + t} \in [\mathbf{v}_f]$ whenever $\mathbf{x} = (x_1, \dots, x_{\ell d_f + n}) \in \mathrm{dom}(f^{\times \ell})$.

Example 6. *Let* $\mathbf{k} = \{0, 1, 2\}$, $\ell = 3$ *and*

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

Then $\mathbf{v}_f = (0, 0, 1)$, $\mathbf{v}_f^{\times 3} = (0, 0, 0, 0, 0, 0, 1, 1, 1)$,

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

$\mathrm{Col}(A) = \{(0, 1, 0)^T, (0, 0, 0)^T, (0, 1, 2)^T\}$, *and* $f^{\times 3}(M_3) =$

$$f^{\times 3} \left(\begin{array}{ccccccccc|ccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 2 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 2 \end{array} \right) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}.$$

It is not difficult to verify that this construction yields the following result.

Lemma 16 ([24])**.** *For every $\ell \geq 1$, $f^{\times \ell} \in [f]_s$. Moreover, for $\ell' \geq 1$, $f^{\times \ell} \in \mathrm{pPol}(\rho_{\ell'})$ iff $\ell \neq \ell'$.*

As an immediate consequence, we thus have the desired corollary.

Corollary 17. *Let C be a strong partial clone on $\mathbf{k}$ and suppose that C contains a partial operation $f \notin \mathrm{Str}(J_{\mathbf{k}})$ that is not a constant operation. Then the set of strong partial clones contained in C is of continuum cardinality.*

5 Open Questions and Related Work

The study of fine-grained complexity is still in its infancy and we have only concentrated a handful of results pertaining to partial polymorphisms. We now present a few concrete questions arising from the results presented thus far, and discuss related research directions.

On the non-existence of minimal strong partial clones

We provided a sketch of one of the constructions in [24], which shows that for any non-constant $f \notin \mathrm{Str}(J_{\{0,1\}})$ there exists $g \notin \mathrm{Str}(J_{\{0,1\}})$ such that $[g]_s \subset [f]_s$. Assuming that $\mathsf{T}(\mathrm{Inv}(\{f\})) < 1$, can we use a similar construction to find a g such that $\mathsf{T}(\mathrm{Inv}(\{f\})) < \mathsf{T}(\mathrm{Inv}(\{g\}))$?

Maximal elements of $\mathcal{I}_{\mathrm{Str}}(J_{\mathbf{k}})$

We have seen that $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ has a largest element $\mathrm{pPol}(R^{\neq\neq\neq 01}_{1/3})$, that constitutes the "easiest NP-complete SAT problem" $\mathrm{SAT}(\{R^{\neq\neq\neq 01}_{1/3}\})$. Given the non-existence of minimal strong partial clones one may be sceptical about the existence of maximal elements of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$. However, such elements do in fact exist. For instance, one can prove that $\mathrm{pPol}(\{R^{\neq\neq\neq 01}_{1/3}, \{(0)\}\})$ and $\mathrm{pPol}(\{R^{\neq\neq\neq 01}_{1/3}, \{(1)\}\})$ are both maximal elements. The caveat here is that $T(\{R^{\neq\neq\neq 01}_{1/3}\}) = \mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}, \{(0)\}\}) = \mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}, \{(1)\}\})$, implying that these elements are not interesting from a complexity theoretical point of view. This raises the question of whether there exists a maximal element $\mathrm{pPol}(\Gamma)$ of $\mathcal{I}_{\mathrm{Str}}(J_{\{0,1\}})$ such that $\mathsf{T}(\{R^{\neq\neq\neq 01}_{1/3}\}) < \mathsf{T}(\Gamma)$.

Strong Maltsev conditions and partial polymorphisms

Lagerkvist & Wahlström [47] propose a usage of partial polymorphisms which is similar to how strong Maltsev conditions are used to characterize the classical complexity of CSPs. For example, given a k and the identities defining a Maltsev operation $m(x, x, y) \approx y, m(x, y, y) \approx x$, we can define a partial operation f such that $\text{dom}(f) = \{(x, x, y), (x, y, y) \mid x, y \in \mathbf{k}\}$ and such that $f(x, x, y) = y$ and $f(x, y, y) = x$. Such a partial operation f is then called a *partial Maltsev operation*.

Now given an operation thus constructed, the goal is then to devise an algorithm for $\text{CSP}(\text{Inv}(f))$ with a running time better than $O(k^n)$. Surprisingly, this is indeed possible for the partial Maltsev operation, where one obtains the upper bound $O(k^{\frac{n}{2}})$. An interesting continuation to this line of research is to consider the identities defining near unanimity operations and edge operations, and investigate if similar improved bounds can be obtained for the corresponding partial operations.

A related application of partial polymorphisms in the realm of exponential-time algorithms was recently demonstrated by Brakensiek & Guruswami [12]. They prove that if Γ is preserved by an infinite family of *partial threshold polymorphisms* then $\text{SAT}(\Gamma)$ can be solved by a fast exponential-time algorithm based on linear programming. For example, this holds for $R_{1/3}$ and for Γ^k_{SAT} for $k \geq 3$. While these two problems are known to admit even faster specialised algorithms [50, 62], the linear programming framework of Brakensiek & Guruswami also provides a clear explanation of *why* these problems admit an exponentially improved algorithm, which demonstrates a distinct advantage of studying fine-grained complexity of such problems in a more abstract, algebraic setting.

Sparsification via partial polymorphisms

There exists many computational properties with a similar flavour as fine-grained complexity. One example from parameterized complexity is *sparsification*: given an instance of a computational problem, associated with a parameter $k \geq 0$, is it possible to compute (in polynomial time) an equivalent instance whose size is bounded by a fixed function in k? For example, in the case of $\text{CSP}(\Gamma)$ we could be interested in reducing the number of constraints with respect to the number of variables in the instance, and ask whether it is possible to reduce the number of constraints in an instance (V, C) to $O(|V|)$ or $O(|V|^2)$.

Sparsification of SAT problems were studied by Jansen & Pieterse [36] who observed that in many interesting cases this question could be translated into properties of (low-degree) polynomials. This same idea was generalised by Lagerkvist & Wahlström [48] who studied this question by embedding CSPs into CSPs preserved by a Maltsev operation and, more generally, by embedding an NP-complete CSP problem into a tractable CSP over a larger domain. The property of admitting "embeddings" of this form could then be witnessed by partial operations closely linked to strong Maltsev conditions. However, the question of whether this algebraic framework could give a complete dichotomy for CSPs admitting linear sparsifications was left open. Similar conditions with closely matching results were later also obtained by Chen et al. [20].

Acknowledgements

We thank the anonymous reviewer for several helpful comments. The third author is partially supported by the Swedish Research Council (VR) under grant 2019-03690.

References

[1] L. Barto, A. A. Krokhin, and Ross Willard. Polymorphisms, and how to use them. In Andrei A. Krokhin and Stanislav Zivný, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 1–44. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017.

[2] L. Barto, J. Oprvsal, and M. Pinsker. The wonderland of reflections. *Israel Journal of Mathematics*, 223(1):363–398, Feb 2018.

[3] M. Behrisch, M. Hermann, S. Mengel, and G. Salzer. Give me another one! In *Proceedings of the 26th International Symposium on Algorithms and Computation (ISAAC-2015)*, pages 664–676, 2015.

[4] M. Behrisch, M. Hermann, S. Mengel, and G. Salzer. As close as it gets. In *Proceedings of the 10th International Workshop on Algorithms and Computation (WALCOM-2016)*, pages 222–235, 2016.

[5] M. Behrisch, M. Hermann, S. Mengel, and G. Salzer. Minimal distance of propositional models. *Theory of Computing Systems*, 63(6):1131–1184, Aug 2019.

[6] A. Björklund, T. Husfeldt, and M. Koivisto. Set partitioning via inclusion-exclusion. *SIAM Journal on Computing*, 39(2):546–563, July 2009.

[7] M. Bodirsky and P. Jonsson. A model-theoretic view on qualitative constraint reasoning. *Journal of Artificial Intelligence Research*, 58:339–385, 2017.

[8] H. Bodlaender, J. van Rooij, and M. van Kooten Niekerk. Partition into triangles on bounded degree graphs. *Theory of Computing Systems*, 52(4):687–718, May 2013.

[9] V. G. Bodnarchuk, L. A. Kaluzhnin, V. N. Kotov, and B. A. Romov. Galois theory for Post algebras. I. *Cybernetics*, 5:243–252, 1969.

[10] V. G. Bodnarchuk, L. A. Kaluzhnin, V. N. Kotov, and B. A. Romov. Galois theory for Post algebras. II. *Cybernetics*, 5:531–539, 1969.

[11] F. Börner and L. Haddad. Maximal partial clones with no finite basis. *Algebra Universalis*, 40(4):453–476, 1998.

[12] J. Brakensiek and V. Guruswami. Bridging between 0/1 and linear programming via random walks. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC-2019)*, pages 568–577, New York, NY, USA, 2019. ACM.

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

[14] A. Bulatov. Complexity of conservative constraint satisfaction problems. *ACM Transactions on Computational Logic*, 12(4):24:1–24:66, July 2011.

[15] A. Bulatov. The complexity of the counting constraint satisfaction problem. *Journal of the ACM*, 60(5):34:1–34:41, 2013.

[16] A. Bulatov. A dichotomy theorem for nonuniform CSPs. In *Proceedings of the 58th Annual Symposium on Foundations of Computer Science (FOCS-2017)*. IEEE Computer Society, 2017.

[17] A. Bulatov and V. Dalmau. A simple algorithm for Mal'tsev constraints. *SIAM Journal On Computing*, 36(1):16–27, 2006.

[18] A. Bulatov, P. Jeavons, and A. Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM Journal on Computing*, 34(3):720–742, March 2005.

[19] 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.

[20] H. Chen, B. M. P. Jansen, and A. Pieterse. Best-case and worst-case sparsifiability of boolean CSPs. In *Proceedings of the 13th International Symposium on Parameterized and Exact Computation (IPEC-2018)*, volume 115 of *LIPIcs*, pages 15:1–15:13. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018.

[21] M. Couceiro and L. Haddad. Intersections of finitely generated maximal partial clones. *Multiple-Valued Logic and Soft Computing*, 19(1-3):85–94, 2012.

[22] M. Couceiro, L. Haddad, V. Lagerkvist, and B. Roy. On the interval of Boolean strong partial clones containing only projections as total operations. In *Proceedings of the 47th International Symposium on Multiple-Valued Logic (ISMVL-2017)*, pages 88–93. IEEE Computer Society, 2017.

[23] M. Couceiro, L. Haddad, and I. G. Rosenberg. Partial clones containing all boolean monotone self-dual partial functions. *Multiple-Valued Logic and Soft Computing*, 27(2-3):183–192, 2016.

[24] M. Couceiro, L. Haddad, and K. Schölzel. On the lower part of the lattice of partial clones. *Multiple-Valued Logic and Soft Computing*, 33(3):177–196, 2019.

[25] M. Couceiro, L. Haddad, K. Schölzel, and T. Waldhauser. A solution to a problem of D. Lau: Complete classification of intervals in the lattice of partial boolean clones. *Multiple-Valued Logic and Soft Computing*, 28(1):47–58, 2017.

[26] N. Creignou. A dichotomy theorem for maximum generalized satisfiability problems. *Journal of Computer and System Sciences*, 51(3):511 – 522, 1995.

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

[28] N. Creignou, S. Khanna, and M. Sudan. *Complexity classifications of Boolean constraint satisfaction problems*. SIAM Monographs on Discrete Mathematics and Applications, 2001.

[29] N. Creignou and H. Vollmer. Boolean constraint satisfaction problems: When does Post's lattice help? In N. Creignou, P. G. Kolaitis, and H. Vollmer, editors, *Complexity of Constraints*, volume 5250 of *Lecture Notes in Computer Science*, pages 3–37. Springer Berlin Heidelberg, 2008.

[30] N. Creignou and B. Zanuttini. A complete classification of the complexity of propositional abduction. *SIAM Journal on Computing*, 36(1):207–229, 2006.

[31] R. de Haan, I. A. Kanj, and S. Szeider. On the subexponential-time complexity of CSP. *Journal of Artificial Intelligence Research (JAIR)*, 52:203–234, 2015.

[32] F. Dylla, J. H. Lee, T. Mossakowski, T. Schneider, A. Van Delden, J. Van De Ven, and D. Wolter. A survey of qualitative spatial and temporal calculi: Algebraic and computational properties. *ACM Computing Surveys*, 50(1):7:1–7:39, April 2017.

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

[34] D. Geiger. Closed systems of functions and predicates. *Pacific Journal of Mathematics*, 27(1):95–100, 1968.

[35] R. Impagliazzo and R. Paturi. On the complexity of k-SAT. *Journal of Computer and System Sciences*, 62(2):367 – 375, 2001.

[36] B. M. P. Jansen and A. Pieterse. Optimal sparsification for some binary CSPs using low-degree polynomials. In *Proceedings of the 41st International Symposium on Mathematical Foundations of Computer Science (MFCS-2016)*, volume 58, pages 71:1–71:14, 2016.

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

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

[39] P. Jonsson, V. Lagerkvist, G. Nordh, and B. Zanuttini. Strong partial clones and the time complexity of SAT problems. *Journal of Computer and System Sciences*, 84:52 – 78, 2017.

[40] P. Jonsson, V. Lagerkvist, and B. Roy. Time complexity of constraint satisfaction via universal algebra. In *Proceedings of the 42nd International Symposium on Mathematical Foundations of Computer Science (MFCS-2017)*, pages 17:1–17:15, 2017.

[41] S. Khanna, M. Sudan, L. Trevisan, and D. P. Williamson. The approximability of constraint satisfaction problems. *SIAM Journal on Computing*, 30(6):1863–1920, 2000.

[42] V. Lager kvist. Weak bases of Boolean co-clones. *Information Processing Letters*, 114(9):462–468, 2014.

[43] V. Lagerkvist and B. Roy. A Preliminary Investigation of Satisfiability Problems Not Harder than 1-in-3-SAT. In *Proceedings of the 41st International Symposium on Mathematical Foundations of Computer Science (MFCS-2016)*, volume 58, pages 64:1–64:14, 2016.

[44] V. Lagerkvist and B. Roy. A dichotomy theorem for the inverse satisfiability problem. In *Proceedings of the 37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS-2017)*, volume 93, pages 39:39–39:14, 2018.

[45] V. Lagerkvist and B. Roy. The inclusion structure of Boolean weak bases. In *Proceedings of the 49th IEEEInternational Symposium on Multiple-Valued Logic (ISMVL-2019)*, pages 31–36. IEEE Computer Society, 2019.

[46] V. Lagerkvist and M. Wahlström. The power of primitive positive definitions with polynomially many variables. *Journal of Logic and Computation*, 27(5):1465–1488, 2017.

[47] V. Lagerkvist and M. Wahlström. Which NP-hard SAT and CSP problems admit exponentially improved algorithms? *CoRR*, abs/1801.09488, 2018.

[48] V. Lagerkvist and M. Wahlström. Sparsification of SAT and CSP problems via tractable extensions. *ACM Transactions on Compututation Theory*, 12(2):13:1–13:29, 2020.

[49] D. Lau. *Function Algebras on Finite Sets: Basic Course on Many-Valued Logic and Clone Theory (Springer Monographs in Mathematics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[50] R. Paturi, P. Pudlák, M. E. Saks, and F. Zane. An improved exponential-time algorithm for k-sat. *Journal of the ACM*, 52(3):337–364, May 2005.

[51] R. Pöschel and L. Kaluznin. *Funktionen- und Relationenalgebren*. DVW, Berlin, 1979.

[52] E. Post. The two-valued iterative systems of mathematical logic. *Annals of Mathematical Studies*, 5:1–122, 1941.

[53] O. Reingold. Undirected connectivity in log-space. *Journal of the ACM*, 55(4):17:1–17:24, September 2008.

[54] B. A. Romov. The algebras of partial functions and their invariants. *Cybernetics and Systems Analysis*, 17(2):157–167, 1981.

[55] F. Rossi, P. van Beek, and T. Walsh, editors. *Handbook of Constraint Programming*, volume 2 of *Foundations of Artificial Intelligence*. Elsevier, 2006.

[56] T. Schaefer. The complexity of satisfiability problems. In *Proceedings of the 10th Annual ACM Symposium on Theory Of Computing (STOC-78)*, pages 216–226. ACM Press, 1978.

[57] H. Schnoor and I. Schnoor. Enumerating all solutions for constraint satisfaction problems. In *Proceedings of the 24th Annual Symposium on Theoretical Aspects of Computer Science (STACS-2007)*, volume 4393, pages 694–705. Springer, 2007.

[58] H. Schnoor and I. Schnoor. Partial polymorphisms and constraint satisfaction problems. In N. Creignou, P. G. Kolaitis, and H. Vollmer, editors, *Complexity of Constraints*, volume 5250 of *Lecture Notes in Computer Science*, pages 229–254. Springer Berlin Heidelberg, 2008.

[59] I. Schnoor. *The weak base method for constraint satisfaction*. PhD thesis, Gottfried Wilhelm Leibniz Universität, Hannover, Germany, 2008.

[60] K. Schölzel. Dichotomy on intervals of strong partial Boolean clones. *Algebra Universalis*, 73(3-4):347–368, 2015.

[61] U. Schöning. A probabilistic algorithm for k-sat and constraint satisfaction problems. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science (FOCS-1999)*, pages 410–, Washington, DC, USA, 1999. IEEE Computer Society.

[62] M. Wahlström. *Algorithms, measures and upper bounds for satisfiability and related problems.* PhD thesis, Linköping University, Theoretical Computer Science Laboratory, The Institute of Technology, 2007.

[63] D. Zhuk. The proof of CSP dichotomy conjecture. In *Proceedings of the 58th Annual Symposium on Foundations of Computer Science (FOCS-2017)*. IEEE Computer Society, 2017.