Kernelization of Constraint Satisfaction Problems: A Study Through Universal Algebra

Victor Lagerkvist[1](✉) and Magnus Wahlström[2]

[1] Institut für Algebra, TU Dresden, Dresden, Germany
victor.lagerqvist@tu-dresden.de

[2] Department of Computer Science, Royal Holloway, University of London, Egham, Great Britain
magnus.wahlstrom@rhul.ac.uk

Abstract. A *kernelization* algorithm for a computational problem is a procedure which compresses an instance into an equivalent instance whose size is bounded with respect to a complexity parameter. For the constraint satisfaction problem (CSP), there exist many results concerning upper and lower bounds for kernelizability of specific problems, but it is safe to say that we lack general methods to determine whether a given problem admits a kernel of a particular size. In this paper, we take an algebraic approach to the problem of characterizing the kernelization limits of NP-hard CSP problems, parameterized by the number of variables. Our main focus is on problems admitting linear kernels, as has, somewhat surprisingly, previously been shown to exist. We show that a finite-domain CSP problem has a kernel with $O(n)$ constraints if it can be embedded (via a domain extension) into a CSP which is preserved by a Maltsev operation. This result utilise a variant of the simple algorithm for Maltsev constraints. In the complementary direction, we give indication that the Maltsev condition might be a complete characterization for Boolean CSPs with linear kernels, by showing that an algebraic condition that is shared by all problems with a Maltsev embedding is also necessary for the existence of a linear kernel unless NP $\subseteq$ co-NP/poly.

1 Introduction

Kernelization is a preprocessing technique based on reducing an instance of a computationally hard problem in polynomial time to an equivalent instance, a *kernel*, whose size is bounded by a function f with respect to a given complexity parameter. The function f is referred to as the *size* of the kernel, and if the size is polynomially bounded we say that the problem admits a *polynomial kernel*. A classical example is Vertex Cover, which admits a kernel with $2k$ vertices, where k denotes the size of the cover [25]. Polynomial kernels are of great interest in parameterized complexity, as well as carrying practical significance in speeding up subsequent computations (e.g., the winning contribution in the 2016 PACE challenge for Feedback Vertex Set used a novel kernelization step as a key component (see https://pacechallenge.wordpress.com/).

J.C. Beck (Ed.): CP 2017, LNCS 10416, pp. 157–171, 2017.
DOI: 10.1007/978-3-319-66158-2_11

When the complexity parameter is a size parameter, e.g., the number of variables n, then such a size reduction is also referred to as *sparsification* (although a sparsification is not always required to run in polynomial time). A prominent example is the famous *sparsification lemma* that underpins research into the Exponential Time Hypothesis [10], which shows that for every k there is a subexponential-time reduction from k-SAT on n variables to k-SAT on $O(n)$ clauses, and hence $\tilde{O}(n)$ bits in size. However, the super-polynomial running time is essential to this result. Dell and van Melkebeek [5] showed that k-SAT cannot be kernelized even down to size $O(n^{k-\varepsilon})$, and Vertex Cover cannot be kernelized to size $O(n^{2-\varepsilon})$, for any $\varepsilon > 0$ unless the polynomial hierarchy collapses (in the sequel, we will make this assumption implicitly). These results suggest that in general, polynomial-time sparsification cannot give non-trivial size guarantees. The first result to the contrary was by Bart Jansen (unpublished until recently [12]), who observed that 1-in-k-SAT admits a kernel with at most n constraints using Gaussian elimination. More surprisingly, Jansen and Pieterse [11] showed that the Not-All-Equal k-SAT problem admits a kernel with $O(n^{k-1})$ constraints, improving on the trivial bound by a factor of n and settling an implicit open problem. In later research, they improved and generalized the method, and also showed that the bound of $O(n^{k-1})$ is tight [12]. These improved upper bounds are all based on rephrasing the SAT problem as a problem of low-degree polynomials, and exploiting linear dependence to eliminate superfluous constraints. Still, it is fair to say that we currently lack the tools for making a general analysis of the kernelizability of a generic SAT problem.

In this paper we take a step in this direction, by studying the kernelizability of the *constraint satisfaction problem* over a constraint language Γ (CSP(Γ)), parameterized by the number of variables n, which can be viewed as the problem of determining whether a set of constraints over Γ is satisfiable. Some notable examples of problems of this kind are k-colouring, k-SAT, 1-in-k-SAT, and not-all-equal-k-SAT. We will occasionally put a particular emphasis on the Boolean CSP problem and therefore denote this problem by SAT(Γ). Note that CSP(Γ) has a trivial polynomial kernel for any finite language Γ (produced by simply discarding duplicate constraints), but the question remains for which languages Γ we can improve upon this. Concretely, our question in this paper is for which languages Γ the problem CSP(Γ) admits a kernel of $O(n^c)$ constraints, for some $c \geq 1$, with a particular focus on linear kernels ($c = 1$).

The Algebraic Approach in Parameterized and Fine-Grained Complexity. For any language Γ, the classical complexity of CSP(Γ) (i.e., whether CSP(Γ) is in P) is determined by the existence of certain algebraic invariants of Γ known as *polymorphisms* [13]. This gave rise to the *algebraic approach* to characterizing the complexity of CSP(Γ) by studying algebraic properties. It has been conjectured that for every Γ, CSP(Γ) is either in P or NP-complete, and that the tractability of a CSP problem can be characterized by a finite list of polymorphisms [3]. Recently, several independent results appeared, claiming to settle this conjecture in the positive [1,26,27]. However, for purposes of parameterized and fine-grained complexity questions, looking at polymorphisms alone is too coarse. More technically, the polymorphisms of Γ characterize the expressive power of Γ up to *primitive positive definitions*, i.e., up to the use of conjunctions, equality constraints, and existential quantification, whereas for many questions a liberal use of existentially quantified local variables is not allowed. In such cases, one may look at the expressive power under *quantifier-free* primitive positive definitions (qfpp-definitions), allowing only conjunctions and equality constraints. This expressive power is characterized by more fine-grained algebraic invariants called *partial polymorphisms*. For example, there are numerous dichotomy results for the complexity of *parameterized* SAT(Γ) and CSP(Γ) problems, both for so-called FPT algorithms and for kernelization [17–19,24], and in each of the cases listed, a dichotomy is given which is equivalent to requiring a finite list of partial polymorphisms of Γ. Similarly, Jonsson et al. [16] showed that the exact running times of NP-hard SAT(Γ) and CSP(Γ) problems in terms of the number of variables n are characterized by the partial polymorphisms of Γ. Unfortunately, studying properties of SAT(Γ) and CSP(Γ) for questions phrased in terms of the size parameter n is again more complicated than for more permissive parameters k. For example, it is known that for every finite set P of strictly partial polymorphisms, the number of relations invariant under P is double-exponential in terms of the arity n (hence they cannot all be described in a polynomial number of bits) [20, Lemma 35]. It can similarly be shown that the existence of a polynomial kernel cannot be characterized by such a finite set P. Instead, such a characterization must be given in another way (for example, Lagerkvist et al. [22] provide a way to finitely characterize all partial polymorphisms of a finite Boolean language Γ).

Our Results. We generalize and extend the results of Jansen and Pieterse [12] in the case of linear kernels to a general recipe for NP-hard SAT and CSP problems in terms of the existence of a *Maltsev embedding*, i.e., an embedding of a language Γ into a tractable language Γ' on a larger domain with a *Maltsev polymorphism*. We show that for any language Γ with a Maltsev embedding into a finite domain, CSP(Γ) has a kernel with $O(n)$ constraints. Attempting an algebraic characterization, we also show an infinite family of *universal* partial operations which are partial polymorphisms of every language Γ with a Maltsev embedding, and show that these operations guarantee the existence of a Maltsev embedding for Γ, albeit into a language with an infinite domain. Turning to lower bounds against linear kernels, we show that the smallest of these universal partial operations is also necessary, in the sense that for any Boolean language Γ which is not invariant under this operation, SAT(Γ) admits no kernel of size $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$. We conjecture that this can be completed into a tight characterization – i.e., that for Boolean languages Γ, SAT(Γ) admits a linear kernel if and only if it is invariant under all universal partial Maltsev operations.

Generalizations for kernels of higher degree are possible, but have been omitted for reasons of length, and we refer the reader to the extended preprint [21].

2 Preliminaries

2.1 The Constraint Satisfaction Problem and Kernelization

A *relation* R over a set of values D is a subset of D^k for some $k \geq 0$, and we write $\text{ar}(R) = k$ to denote the arity of R. A set of relations $\mathit{\Gamma}$ is referred to as a *constraint language*. An instance (V, C) of the *constraint satisfaction problem* over a constraint language $\mathit{\Gamma}$ over D ($\text{CSP}(\mathit{\Gamma})$) is a set V of variables and a set C of constraint applications $R(v_1, \ldots, v_k)$ where $R \in \mathit{\Gamma}$, $\text{ar}(R) = k$, and $v_1, \ldots, v_k \in V$. The question is whether there exists a function $f : V \rightarrow D$ such that $(f(v_1), \ldots, f(v_k)) \in R$ for each $R(v_1, \ldots, v_k)$ in C? If $\mathit{\Gamma}$ is Boolean we denote $\text{CSP}(\mathit{\Gamma})$ by $\text{SAT}(\mathit{\Gamma})$, and we let BR denote the set of all Boolean relations. As an example, let $R_{1/3} = \{(0,0,1),(0,1,0),(1,0,0)\}$. Then $\text{SAT}(\{R_{1/3}\})$ can be viewed as an alternative formulation of the 1-in-3-SAT problem restricted to instances consisting only of positive literals. More generally, if we let $R_{1/k} = \{(x_1, \ldots, x_k) \in \{0,1\}^k \mid x_1 + \ldots + x_k = 1\}$, then $\text{SAT}(\{R_{1/k}\})$ is a natural formulation of 1-in-k-SAT without negation.

A *parameterized problem* is a subset of $\mathit{\Sigma}^* \times \mathbb{N}$ where $\mathit{\Sigma}$ is a finite alphabet. Hence, each instance is associated with a natural number, called the *parameter*.

Definition 1. *A* kernelization algorithm*, or a* kernel*, for a parameterized problem $L \subseteq \mathit{\Sigma}^* \times \mathbb{N}$ is a polynomial-time algorithm which, given an instance $(x, k) \in \mathit{\Sigma}^* \times \mathbb{N}$, computes $(x', k') \in \mathit{\Sigma}^* \times \mathbb{N}$ such that (1) $(x, k) \in L$ if and only if $(x', k') \in L$ and (2) $|x'| + k' \leq f(k)$ for some function f.*

The function f in the above definition is sometimes called the *size* of the kernel. In this paper, we are mainly interested in the case where the parameter denotes the number of variables in a given $\text{CSP}(\mathit{\Gamma})$ instance.

2.2 Operations and Relations

An n-ary function $f : D^n \rightarrow D$ over a domain D is typically referred to as an *operation* on D, although we will sometimes use the terms function and operation interchangeably. We let $\text{ar}(f) = n$ denote the arity of f. Similarly, an n-ary *partial operation* over a set D of values is a map $f : X \rightarrow D$, where $X \subseteq D^n$ is called the *domain* of f. Again, we let $\text{ar}(f) = n$, and furthermore let $\text{domain}(f) = X$. If f and g are n-ary partial operations with $\text{domain}(g) \subseteq \text{domain}(f)$ and $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ for each $(x_1, \ldots, x_n) \in \text{domain}(g)$, then g is said to be a *subfunction* of f.

Definition 2. *An n-ary partial operation f is a* partial polymorphism *of a k-ary relation R if, for every sequence $t_1, \ldots, t_n \in R$, either $f(t_1, \ldots, t_n) \in R$ or $(t_1[i], \ldots, t_n[i]) \notin \text{domain}(f)$ for some $1 \leq i \leq k$, where $f(t_1, \ldots, t_n) = (f(t_1[1], \ldots, t_n[1]), \ldots, f(t_1[k], \ldots, t_n[k]))$.*

If f is total we simply say that f is a *polymorphism* of R, and in both cases we sometimes also say that f *preserves* R, or that R is *invariant* under f. For a constraint language $\mathit{\Gamma}$ we then let $\mathrm{Pol}(\Gamma)$ and $\mathrm{pPol}(\Gamma)$ denote the set of operations and partial operations preserving every relation in $\mathit{\Gamma}$, respectively, and if F is a set of total or partial operations we let $\mathrm{Inv(F)}$ denote the set of all relations invariant under F. It is known that $\mathrm{Pol}(\Gamma)$ and $\mathrm{pPol}(\Gamma)$ are closed under composition of (partial) operations, i.e., if $f \circ g_1, \ldots, g_m(x_1, \ldots, x_n) = f(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$ is included in $\mathrm{Pol}(\Gamma)$ (respectively $\mathrm{pPol}(\Gamma)$) then $f(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n))$ is included in $\mathrm{Pol}(\Gamma)$ (respectively $\mathrm{pPol}(\Gamma)$) [23]. It is also known that $\mathrm{Pol}(\Gamma)$ and $\mathrm{pPol}(\Gamma)$ for each n and $i \leq n$ contain every *projection* $\pi_i^n(x_1, \ldots, x_i, \ldots, x_n) = x_i$. On the relational side, if every operation in F is total, then $\mathrm{Inv(F)}$ is closed under *primitive positive definitions* (pp-definitions) which are logical formulas consisting of existential quantification, conjunction, and equality constraints. In symbols, we say that a k-ary relation R has a pp-definition over a constraint language $\mathit{\Gamma}$ over a domain D if $R(x_1, \ldots, x_k) \equiv \exists y_1, \ldots, y_{k'} \, . \, R_1(\mathbf{x_1}) \wedge \ldots \wedge R_m(\mathbf{x_m})$, where each $R_i \in \mathit{\Gamma} \cup \{\mathrm{Eq}\}$, $\mathrm{Eq} = \{(\mathrm{x}, \mathrm{x}) \mid \mathrm{x} \in \mathrm{D}\}$ and each $\mathbf{x_i}$ is an $\mathrm{ar}(R_i)$-ary tuple of variables over $x_1, \ldots, x_k, y_1, \ldots, y_{k'}$. If F is a set of partial operations then $\mathrm{Inv(F)}$ is closed under *quantifier-free primitive positive definitions* (qfpp-definitions), i.e., pp-definitions that do not make use of existential quantification. As a shorthand, we let $[F] = \mathrm{Pol}(\mathrm{Inv(F)})$, $\langle \mathit{\Gamma} \rangle = \mathrm{Inv}(\mathrm{Pol}(\Gamma))$, and $\langle \mathit{\Gamma} \rangle_{\nexists} = \mathrm{Inv}(\mathrm{pPol}(\Gamma))$. We then have the following *Galois connections* [8].

Theorem 3. *Let $\mathit{\Gamma}, \mathit{\Gamma}'$ be constraint languages. Then (1) $\mathit{\Gamma} \subseteq \langle \mathit{\Gamma}' \rangle_{\nexists}$ if and only if* $\mathrm{pPol}(\Gamma') \subseteq \mathrm{pPol}(\Gamma)$ *and (2) $\mathit{\Gamma} \subseteq \langle \mathit{\Gamma}' \rangle$ if and only if* $\mathrm{Pol}(\Gamma') \subseteq \mathrm{Pol}(\Gamma)$.

Jonsson et al. [16] proved the following theorem, showing that partial polymorphisms are indeed a refinement over total polymorphisms, since the latter are only guaranteed to provide polynomial-time many-one reductions [15].

Theorem 4. *If $\mathit{\Gamma}$, $\mathit{\Gamma}'$ are finite languages and* $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(\Gamma')$ *there exists a constant c and a polynomial-time reduction from* $\mathrm{CSP}(\mathit{\Gamma}')$ *to* $\mathrm{CSP}(\mathit{\Gamma})$ *mapping (V, C) of* $\mathrm{CSP}(\mathit{\Gamma}')$ *to (V', C') of* $\mathrm{CSP}(\mathit{\Gamma})$ *where $|V'| \leq |V|$ and $|C'| \leq c|C|$.*

Last, we will define a particular type of operation which is central to our algebraic approach. A *Maltsev operation* over $D \supseteq \{0, 1\}$ is a ternary operation ϕ which for all $x, y \in D$ satisfies the two identities $\phi(x, x, y) = y$ and $\phi(x, y, y) = x$. Before we can explain the powerful, structural properties of relations invariant under Maltsev operations, we need a few technical definitions from Bulatov and Dalmau [2]. If $t \in D^n$ is a tuple we let $t[i]$ denote the ith element in t and we let $\mathrm{pr}_{i_1, \ldots, i_{n'}}(t) = (t[i_1], \ldots, t[i_{n'}])$, $n' \leq n$, denote the *projection* of t on (not necessarily distinct) coordinates $i_1, \ldots, i_{n'} \in \{1, \ldots, n\}$. Similarly, if R is an n-ary relation we let $\mathrm{pr}_{i_1, \ldots, i_{n'}}(R) = \{\mathrm{pr}_{i_1, \ldots, i_{n'}}(t) \mid t \in R\}$. Let t, t' be two n-ary tuples over D. We say that (t, t') *witnesses* a tuple $(i, a, b) \in \{1, \ldots, n\} \times D^2$ if $\mathrm{pr}_{1, \ldots, i-1}(t) = \mathrm{pr}_{1, \ldots, i-1}(t')$, $t[i] = a$, and $t'[i] = b$. The *signature* $\mathrm{Sig}(R)$ of an n-ary relation R over D is then defined as

$$\{(i, a, b) \in \{1, \ldots, n\} \times D^2 \mid \exists t, t' \in R \text{ such that } (t, t') \text{ witnesses } (i, a, b)\},$$

and we say that $R' \subseteq R$ is a *representation* of R if $\mathrm{Sig}(R) = \mathrm{Sig}(R')$. If R' is a representation of R it is said to be *compact* if $|R'| \leq 2|\mathrm{Sig}(R)|$, and it is known that every relation invariant under a Maltsev operation admits a compact representation. Furthermore, we have the following theorem from Bulatov and Dalmau, where we let $\langle R \rangle_f$ denote the smallest superset of R invariant under f.

Theorem 5 **([2]).** *Let ϕ be a Maltsev operation over a finite domain, $R \in \mathrm{Inv}(\{\phi\})$ a relation, and R' a representation of R. Then $\langle R' \rangle_\phi = R$.*

Hence, relations invariant under Maltsev operations are reconstructible from their compact representations.

3 Maltsev Embeddings and Kernels of Linear Size

In this section we give general upper bounds for kernelization of NP-hard CSP problems, utilising Maltsev operations. At this stage the connection between Maltsev operations, compact representations and tractability of Maltsev constraints might not be immediate. In a nutshell, the Maltsev algorithm [2] works as follows (where ϕ is a Maltsev operation over a finite set D). First, let $(V, \{C_1, \ldots, C_m\})$ be an instance of $\mathrm{CSP}(\mathrm{Inv}(\{\phi\}))$, and let S_0 be a compact representation of $D^{|V|}$. Second, for each $i \in \{1, \ldots, m\}$ compute a compact representation S_i of the solution space of the instance $(V, \{C_1, \ldots, C_i\})$ using S_{i-1}. Third, answer yes if $S_m \neq \emptyset$ and no otherwise. For a full description of the involved procedures we refer the reader to Bulatov and Dalmau [2] and Dyer and Richerby [6].

Example 6. We review two familiar special cases of this result. First, consider a linear equation $\sum_i \alpha_i x_i = b$, interpreted over a finite field $\mathbb{F}$. It is clear that the set of solutions to such an equation is invariant under $x_1 - x_2 + x_3$ (over $\mathbb{F}$), hence systems of linear equations are a special case of Maltsev constraints, and can in principle be solved by the Maltsev algorithm. Second, for a more general example, let $G = (D, \cdot)$ be a finite group, and let $s(x, y, z) = x \cdot (y^{-1}) \cdot z$ be the *coset generating operation* of G. Then s is Maltsev, hence $\mathrm{CSP}(\mathrm{Inv}(\{\mathrm{s}\}))$ is tractable; this was shown by Feder and Vardi [7], but also follows from the Maltsev algorithm. In particular, if $G = (D, +)$ is an Abelian group where $|D|$ is prime, then $R \in \mathrm{Inv}(\{\mathrm{s}\})$ if and only if R is the solution space of a system of linear equations modulo $|D|$ [14].

Since $\mathrm{CSP}(\mathit{\Gamma})$ is tractable whenever $\mathit{\Gamma}$ is preserved by a Maltsev operation, it might not be evident how the Maltsev algorithm can be used for constructing kernels for NP-hard CSPs. The basic idea is to embed $\mathit{\Gamma}$ into a language $\hat{\mathit{\Gamma}}$ over a larger domain, which is preserved by a Maltsev operation. This allows us to use the advantageous properties of relations invariant under Maltsev operations, in order to compute a kernel for the original problem.

Definition 7. *A constraint language $\mathit{\Gamma}$ over D admits an* embedding *over the constraint language $\hat{\mathit{\Gamma}}$ over $D' \supseteq D$ if there exists a bijection $h : \mathit{\Gamma} \rightarrow \hat{\mathit{\Gamma}}$ such that* $\mathrm{ar}(h(R)) = \mathrm{ar}(R)$ *and* $h(R) \cap D^{\mathrm{ar}(R)} = R$ *for every $R \in \mathit{\Gamma}$.*

If $\hat{\Gamma}$ is preserved by a Maltsev operation then we say that Γ admits a *Maltsev embedding*. We do not exclude the possibility that D' is infinite, but in this section we will only be concerned with finite domains, and therefore do not explicitly state this assumption. If the bijection h is efficiently computable and there exists a polynomial p such that $h(R)$ can be computed in $O(p(|R|))$ time for each $R \in \Gamma$, then we say that Γ admits a *polynomially bounded* embedding. In particular, an embedding over a finite domain of any finite Γ is polynomially bounded.

Example 8. Recall from Sect. 2 that $R_{1/3} = \{(0,0,1),(0,1,0),(1,0,0)\}$. We claim that $R_{1/3}$ has a Maltsev embedding over $\{0,1,2\}$. Let $\hat{R}_{1/3} = \{(x,y,z) \in \{0,1,2\}^3 \mid x+y+z = 1 \,(\mathrm{mod}\, 3)\}$. Then $\hat{R}_{1/3} \cap \{0,1\}^3 = R_{1/3}$, and from Example 6 we recall that $\hat{R}_{1/3}$ is preserved by a Maltsev operation. Hence, $\hat{R}_{1/3}$ is indeed a Maltev embedding of $R_{1/3}$. More generally, for every k, $R_{1/k}$ has a Maltsev embedding into equations over a finite field of size at least k.

For a CSP(Γ) instance $I = (\{x_1,\ldots,x_n\}, C)$ we let Ψ_I be the relation $\{(g(x_1),\ldots,g(x_n)) \mid g \text{ satisfies } I\}$, and if ϕ is a Maltsev operation and $I = (V, \{C_1,\ldots,C_m\})$ an instance of CSP(Inv($\{\phi\}$)) we let $\mathrm{Seq}(I) = (S_0, S_1 \ldots, S_m)$ denote the compact representations of the relations $\Psi_{(V,\emptyset)}$, $\Psi_{(V,\{C_1\})}$, $\ldots$, $\Psi_{(V,\{C_1,\ldots,C_m\})}$ computed by the Maltsev algorithm. We remark that the ordering of the constraints in Seq(I) does not influence the upper bound for the kernel.

Definition 9. *Let ϕ be a Maltsev operation, p a polynomial and let $\Delta \subseteq$ Inv($\{\phi\}$). We say that Δ and* CSP(Δ) *have* chain length *p if $|\{\langle S_i \rangle_\phi \mid i \in \{0,1,\ldots,|C|\}\}| \leq p(|V|)$ for each instance $I = (V,C)$ of* CSP(Δ)*, where* $\mathrm{Seq}(I) = (S_0, S_1, \ldots, S_{|C|})$.

We now have everything in place to define our kernelization algorithm.

Theorem 10. *Let Γ be a constraint language over D which admits a polynomially bounded Maltsev embedding $\hat{\Gamma}$ with chain length p. Then* CSP(Γ) *has a kernel with $O(p(|V|))$ constraints.*

Proof. Let $\phi \in \mathrm{Pol}(\hat{\Gamma})$ denote the Maltsev operation witnessing the embedding $\hat{\Gamma}$. Given an instance $I = (V,C)$ of CSP(Γ) we can obtain an instance $I' = (V, C')$ of CSP($\hat{\Gamma}$) by replacing each constraint $R_i(\mathbf{x_i})$ in C by $\hat{R}_i(\mathbf{x_i})$. We arbitrarily order the constraints as $C' = (C_1,\ldots,C_m)$ where $m = |C'|$. We then iteratively compute the corresponding sequence $\mathrm{Seq}(I') = (S_0, S_1, \ldots, S_{|C'|})$. This can be done in polynomial time with respect to the size of I via the same procedure as the Maltsev algorithm. For each $i \in \{1,\ldots,m\}$ we then do the following.

1. Let the ith constraint be $C_i = \hat{R}_i(x_{i_1},\ldots,x_{i_r})$ with $\mathrm{ar}(R_i) = r$.
2. For each $t \in S_{i-1}$ determine whether $\mathrm{pr}_{i_1,\ldots,i_r}(t) \in \hat{R}_i$.
3. If yes, then remove the constraint C_i, otherwise keep it.

This can be done in polynomial time with respect to the size of the instance I', since (1) $|S_{i-1}|$ is bounded by a polynomial in $|V|$ and (2) the test $\mathrm{pr}_{i_1,\ldots,i_r}(t) \in \hat{R}_i$ can naively be checked in linear time with respect to $|\hat{R}_i|$. We claim that the procedure outlined above will correctly detect whether the constraint C_i is redundant or not with respect to $\langle S_{i-1}\rangle_\phi$, i.e., whether $\langle S_{i-1}\rangle_\phi = \langle S_i\rangle_\phi$. First, observe that if there exists $t \in S_{i-1}$ such that $\mathrm{pr}_{i_1,\ldots,i_r}(t) \notin \hat{R}_i$, then the constraint is clearly not redundant. Hence, assume that $\mathrm{pr}_{i_1,\ldots,i_r}(t) \in \hat{R}_i$ for every $t \in S_{i-1}$. Then $S_{i-1} \subseteq \langle S_i\rangle_\phi$, hence also $\langle S_{i-1}\rangle_\phi \subseteq \langle S_i\rangle_\phi$. On the other hand, $\langle S_i\rangle_\phi \subseteq \langle S_{i-1}\rangle_\phi$ holds trivially. Therefore, equality must hold. Let $I'' = (V, C'')$ denote the resulting instance. Since $\mathrm{CSP}(\mathrm{Inv}(\{\phi\}))$ has chain length p it follows that (1) the sequence $\langle S_0\rangle_\phi, \langle S_1\rangle_\phi, \ldots, \langle S_{|C'|}\rangle_\phi$ contains at most $p(|V|)$ distinct elements, hence $|C''| \leq p(|V|)$, and (2) $\Psi_{I'} = \Psi_{I''}$. Clearly, it also holds that $\Psi_I = (\Psi_{I'} \cap \{0,1\}^{|V|}) = (\Psi_{I''} \cap \{0,1\}^{|V|})$. Hence, we can safely transform I'' to an instance I^* of $\mathrm{CSP}(\Gamma)$ by replacing each constraint $\hat{R}_i(\mathbf{x_i})$ with $R_i(\mathbf{x_i})$. Then I^* is an instance of $\mathrm{CSP}(\Gamma)$ with at most $p(|V|)$ constraints, such that $\Psi_I = \Psi_{I^*}$. In particular, I^* has a solution if and only if I has a solution. □

All that remains to be proven now is that there actually exist Maltsev embeddings with bounded chain length.

Theorem 11. $\mathrm{CSP}(\mathrm{Inv}(\{\phi\}))$ *has chain length* $O(|D||V|)$ *for every Maltsev operation* ϕ *over a finite* D.

Proof. Let $I = (V, C)$ be an instance of $\mathrm{CSP}(\mathrm{Inv}(\{\phi\}))$, with $|V| = n$ and $|C| = m$, and let $\mathrm{Seq}(I) = (S_0, S_1, \ldots, S_m)$ be the sequence of compact representations computed by the Maltsev algorithm. First, we claim that $\mathrm{Sig}(S_{i+1}) \subseteq \mathrm{Sig}(S_i)$ for every $i < m$. To see this, pick $(j, a, b) \in \mathrm{Sig}(S_i)$, where $j \in \{1, \ldots, |V|\}$ and $a, b \in D$. Then there exists $t, t' \in S_i$ such that (t, t') witnesses (j, a, b), i.e., $\mathrm{pr}_{1,\ldots,j-1}(t) = \mathrm{pr}_{1,\ldots,j-1}(t')$, and $t[j] = a$, $t'[j] = b$. Since $\langle S_{i-1}\rangle_\phi \supseteq \langle S_i\rangle_\phi \supseteq S_i$, it follows that $t, t' \in \langle S_{i-1}\rangle_\phi$, and hence also that $(j, a, b) \in \mathrm{Sig}(\langle S_{i-1}\rangle_\phi)$. But since S_{i-1} is a representation of $\langle S_{i-1}\rangle_\phi$, $\mathrm{Sig}(S_{i-1}) = \mathrm{Sig}(\langle S_{i-1}\rangle_\phi)$, from which we infer that $(j, a, b) \in \mathrm{Sig}(S_{i-1})$. Second, we claim that the sets $(j, a, b) \in \mathrm{Sig}(S_i)$ induce an equivalence relation on $\mathrm{pr}_j(\langle S_i\rangle_\phi)$ for every $i \leq m$, $j \leq n$[1]. Let $a \sim b$ hold if and only if $(j, a, b) \in \mathrm{Sig}(S_i)$. Note that $(j, a, a) \in \mathrm{Sig}(S_i)$ if and only if $a \in \mathrm{pr}_j(S_i)$, and that $(j, a, b) \notin \mathrm{Sig}(S_i)$ for any b if $a \notin \mathrm{pr}_j(S_i)$. Also note that $\sim$ is symmetric by its definition. It remains to show transitivity. Let $(j, a, b) \in \mathrm{Sig}(S_i)$ be witnessed by (t_a, t_b) and $(j, a, c) \in \mathrm{Sig}(S_i)$ be witnessed by (t'_a, t'_c). We claim that $t_c := \phi(t_a, t'_a, t'_c) \in S_i$ is a tuple such that (t_b, t_c) witnesses $(i, b, c) \in \mathrm{Sig}(S_i)$. Indeed, for every $i' < i$ we have $\phi(t_a[i'], t'_a[i'], t'_c[i']) = \phi(t_a[i'], t'_a[i'], t'_a[i']) = t_a[i']$, whereas $\phi(t_a[i'], t'_a[i'], t'_c[i']) = (a, a, c) = c$. Since $t_a[i'] = t_b[i']$ for every $i' < i$, it follows that (t_b, t_c) witnesses $(j, b, c) \in \mathrm{Sig}(S_i)$. Hence $\sim$ is an equivalence relation on $\mathrm{pr}_j(S_i)$. We wrap up the proof as follows. Note that if $\mathrm{Sig}(S_{i+1}) = \mathrm{Sig}(S_i)$, then $\langle S_i\rangle_\phi = \langle S_{i+1}\rangle_\phi$ since S_{i+1} is a compact representation of $\langle S_i\rangle_\phi$. Hence, we need to bound the number of times that

[1] This property is essentially folklore in universal algebra, and follows from the *rectangularity* property of relations invariant under Maltsev operations.

$\mathrm{Sig}(S_{i+1}) \subset \mathrm{Sig}(S_i)$ can hold. Now, whenever $\mathrm{Sig}(S_{i+1}) \subset \mathrm{Sig}(S_i)$, then either $\mathrm{pr}_j(\langle S_i\rangle_\phi) \subset \mathrm{pr}_j(\langle S_{i+1}\rangle_\phi)$ for some j, or the equivalence relation induced by tuples $(j,a,b) \in \mathrm{Sig}(S_{i+1})$ is a refinement of that induced by tuples $(j,a,b) \in \mathrm{Sig}(S_i)$ for some j. Both of these events can only occur $|D|-1$ times for every position j (unless $S_m = \emptyset$). Hence the chain length is bounded by $2|V||D|$. □

This bound can be slightly improved for a particular class of Maltsev operations. Recall from Example 6 that $s(x,y,z) = x \cdot y^{-1} \cdot z$ is the coset generating operation of a group $G = (D, \cdot)$.

Lemma 12. *Let $G = (D, \cdot)$ be a finite group and let s be its coset generating operation. Then* $\mathrm{CSP}(\mathrm{Inv}(\{\mathrm{s}\}))$ *has chain length* $O(|V| \log |D|)$.

Proof. Let $I = (V, C)$ be an instance of $\mathrm{CSP}(\mathrm{Inv}(\{\mathrm{s}\}))$, where $|V| = n$ and $|C| = m$. Let $\mathrm{Seq}(I) = (S_0, S_1, \ldots, S_m)$ be the corresponding sequence. First observe that S_0 is a compact representation of D^n and that $(D^n, \cdot)$ is nothing else than the nth direct power of G. It is well-known that R is a coset of a subgroup of $(D^n, \cdot)$ if and only if s preserves R [4]. In particular, this implies that S_1 is a compact representation of a subgroup of $(D^n, \cdot)$, and more generally that each S_i is a compact representation of a subgroup of $\langle S_{i-1}\rangle_s$. Lagrange's theorem then reveals that $|\langle S_i\rangle_s|$ divides $|\langle S_{i-1}\rangle_s|$, which implies that the sequence $\langle S_0\rangle_s, \langle S_1\rangle_s, \ldots, \langle S_m\rangle_s$ contains at most $n \log_2 |D| + 1$ distinct elements. □

Note that the bound $|V| \log |D|$ is in fact a bound on the length of a chain of subgroups of G^n; thus it can be further strengthened in certain cases. In particular, if $|D|$ is prime then the bound on chain length is simply $|V| + 1$ and the resulting kernel has at most $|V|$ constraints. Thus, Theorem 10 and Lemma 12 (via Example 8) give an alternate proof of the result that $\mathrm{SAT}(\{R_{1/k}\})$ has a kernel with at most $|V|$ constraints. More generally, we get the following cases. First, if Γ can be represented via linear equations over a finite field, then $\mathrm{CSP}(\Gamma)$ has a kernel with at most $|V|$ constraints. This closely mirrors the result of Jansen and Pieterse [12]. Second, if Γ can be embedded into cosets of a finite group over a set D, then $\mathrm{CSP}(\Gamma)$ has a kernel of $O(|V| \log |D|)$ constraints, but not necessarily $|V|$ constraints (for example, $x = 0 \pmod 2$ and $x = 0 \pmod 3$ are independent over Z_6). Third, in the general case, where Γ has an embedding into a language on domain D with some arbitrary Maltsev polymorphism with no further structure implied, $\mathrm{CSP}(\Gamma)$ has a kernel with $O(|V||D|)$ constraints. (More generally, for $|\Gamma|$ finite, we may use different Maltsev embeddings for different $R \in \Gamma$, and apply the above kernel to each relation R in turn, for a kernel of $O(|\Gamma||D||V|)$ constraints, where $|D|$ is the largest domain used in these embeddings.) Each case is more general than the previous: there are groups whose coset generating operations cannot be represented by Abelian groups (for example A_n, the group of all even permutations over $\{1, \ldots, n\}$ for $n \geq 3$), and it is known that a Maltsev operation ϕ over D is the coset generating operation of a group $(D, \cdot)$ if and only if $\phi(\phi(x,y,z),z,u) = \phi(x,y,u)$, $\phi(u,z,\phi(z,y,x)) = \phi(u,y,x)$ for all $x,y,z,u \in D$ [4]. Hence, any Maltsev operation which does not satisfy these two identities cannot be viewed as the coset generating operation of a group.

4 Partial Polymorphisms and Lower Bounds

We have seen that Maltsev embeddings provide an algebraic criterion for determining that a $\mathrm{CSP}(\Gamma)$ problem admits a kernel of a fixed size. In this section we develop a connection between the partial polymorphisms of a constraint language and the existence of a Maltsev embedding, and leverage these results in order to prove lower bound on kernelization for $\mathrm{SAT}(\Gamma)$. Let $f : D^k \rightarrow D$ be a k-ary operation over $D \supseteq \{0, 1\}$. We can then associate a partial Boolean operation $f_{|\mathbb{B}}$ with f by restricting f to the Boolean arguments which also result in a Boolean value. In other words $\mathrm{domain}(f_{|\mathbb{B}}) = \{(x_1, \ldots, x_k) \in \{0, 1\}^k \mid f(x_1, \ldots, x_k) \in \{0, 1\}\}$, and $f_{|\mathbb{B}}(x_1, \ldots, x_k) = f(x_1, \ldots, x_k)$ for every $(x_1, \ldots, x_k) \in \mathrm{domain}(f_{|\mathbb{B}})$. We then characterize the partial polymorphisms of Boolean constraint languages admitting Maltsev embeddings as follows.

Theorem 13. *Let Γ be a Boolean constraint language, ϕ a Maltsev operation, and $\hat{\Gamma} = \{\langle R\rangle_\phi \mid R \in \Gamma\}$. Then $\hat{\Gamma}$ is a Maltsev embedding of Γ if and only if $f_{|\mathbb{B}} \in \mathrm{pPol}(\Gamma)$ for every $f \in \mathrm{Pol}(\hat{\Gamma})$.*

Proof. For the first direction, assume that $\hat{\Gamma}$ is a Maltsev embedding of Γ, and assume that there exists $R \in \Gamma$ and an n-ary $f \in \mathrm{Pol}(\hat{\Gamma})$ such that $f_{|\mathbb{B}}(t_1, \ldots, t_n) \notin R$ for $t_1, \ldots, t_n \in R$. By construction, $f_{|\mathbb{B}}(t_1, \ldots, t_n) = t$ is a Boolean tuple. But since $\hat{R} \cap \{0, 1\}^{\mathrm{ar}(R)} = R$, this implies (1) that $t \notin \hat{R}$ and (2) that $f_{|\mathbb{B}}(t_1, \ldots, t_n) = f(t_1, \ldots, t_n) = t \notin \hat{R}$. Hence, f does not preserve $\hat{R}$ or $\hat{\Gamma}$, and we conclude that $f_{|\mathbb{B}} \in \mathrm{pPol}(\Gamma)$. For the other direction, assume that $\{f_{|\mathbb{B}} \mid f \in \mathrm{Pol}(\hat{\Gamma})\} \subseteq \mathrm{pPol}(\Gamma)$ but that there exists $\hat{R} \in \hat{\Gamma}$ such that $\hat{R} \cap \{0, 1\}^{\mathrm{ar}(R)} \supset R$. Let $t \in \hat{R} \cap \{0, 1\}^{\mathrm{ar}(R)} \setminus R$. By construction of $\hat{R}$ it follows that there exists an n-ary $f \in [\{\phi\}]$ and $t_1, \ldots, t_n \in R$ such that $f(t_1, \ldots, t_n) = t \notin R$. But then it follows that $f_{|\mathbb{B}}(t_1, \ldots, t_n)$ is defined as well, implying that $f_{|\mathbb{B}}(t_1, \ldots, t_n) \notin R$. This contradicts the assumption that $f_{|\mathbb{B}} \in \mathrm{pPol}(\Gamma)$ for every $f \in \mathrm{Pol}(\hat{\Gamma})$. □

Hence, the existence of a Maltsev embedding can always be witnessed by the partial polymorphisms of a constraint language. We will now describe the partial operations that preserve every Boolean language with a Maltsev embedding. Therefore, say that f is a *universal partial Maltsev operation* if $f \in \mathrm{pPol}(\Gamma)$ for every Boolean Γ admitting a Maltsev embedding. Due to Theorem 13 this is tantamount to finding a Maltsev operation ϕ such that every Boolean language with a Maltsev embedding admits a Maltsev embedding over ϕ.

Definition 14. *Let the infinite domain D_∞ be recursively defined to contain* 0, 1, *and ternary tuples of the form (x, y, z) where $x, y, z \in D_\infty$, $x \neq y$, $y \neq z$. The Maltsev operation u over D_∞ is defined as $u(x, x, y) = y$, $u(x, y, y) = x$, and $u(x, y, z) = (x, y, z)$ otherwise.*

We will now prove that $q_{|\mathbb{B}}$ is a universal partial Maltsev operation if $q \in [\{u\}]$.

Theorem 15. *Let $q \in [\{u\}]$. Then $q_{|\mathbb{B}}$ is a universal partial Maltsev operation.*

Proof. We provide a sketch of the most important ideas. Let $q \in [\{u\}]$ be n-ary, and let $\varGamma$ be a Boolean constraint language admitting a Maltsev embedding with respect to an operation ϕ. It is known that every operation in $[\{u\}]$ can be expressed as a term over u [9], and if we let p denote the operation defined by replacing each occurence of u in this term by ϕ we obtain an operation included in $[\{\phi\}]$. We then claim that $q_{|\mathbb{B}}$ can be obtained as a subfunction of $p_{|\mathbb{B}}$, which is sufficient to prove the result since $p_{|\mathbb{B}} \in \mathrm{pPol}(\Gamma)$ via Theorem 13 and since $\mathrm{pPol}(\Gamma)$ is known to be closed under taking subfunctions [23]. The intuition behind this step is that $q(x_1, \ldots, x_n)$ for $x_1, \ldots, x_n \in \{0, 1\}$ may only return a Boolean value through a sequence of Maltsev conditions, and since ϕ is also a Maltsev operation, it has to abide by these conditions as well. Formally, this can be proven straightforwardly through induction on the terms defining q and p. □

We may thus combine Theorem 13 and Theorem 15 to obtain a complete description of all universal partial Maltsev operations. Even though these proofs are purely algebraic we will shortly see that universal Maltsev operations have strong implications for kernelizability of SAT. For this purpose we define the *first partial Maltsev operation* ϕ_1 as $\phi_1(x, y, y) = x$ and $\phi_1(x, x, y) = y$ for all $x, y \in \{0, 1\}$, and observe that $\mathrm{domain}(\phi_1) = \{(0, 0, 0), (1, 1, 1), (0, 0, 1), (1, 1, 0), (1, 0, 0), (0, 1, 1)\}$. Via Theorem 15 it follows that ϕ_1 is equivalent to $u_{|\mathbb{B}}$, and is therefore a universal partial Maltsev operation. We will now prove that $\phi_1 \in \mathrm{pPol}(\Gamma)$ is in fact a necessary condition for the existence of a linear-sized kernel for $\mathrm{SAT}(\varGamma)$, modulo a standard complexity theoretical assumption. A pivotal part of this proof is that if $\phi_1 \notin \mathrm{pPol}(\Gamma)$, then $\varGamma$ can qfpp-define a relation $\varPhi_1$, which can be used as a gadget in a reduction from the Vertex Cover problem. This relation is defined as $\varPhi_1(x_1, x_2, x_3, x_4, x_5, x_6) \equiv (x_1 \vee x_4) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_4) \wedge (x_5 = 0) \wedge (x_6 = 1)$. The following lemma shows a strong relationship between ϕ_1 and $\varPhi_1$.

Lemma 16. *If $\varGamma$ is a Boolean constraint language such that $\langle \varGamma \rangle = BR$ and $\phi_1 \notin \mathrm{pPol}(\Gamma)$ then $\varPhi_1 \in \langle \varGamma \rangle_{\nexists}$.*

Proof. Before the proof we need two central observations. First, the assumption that $\langle \varGamma \rangle = BR$ is well-known to be equivalent to that $\mathrm{Pol}(\Gamma)$ consists only of projections. Second, $\varPhi_1$ consists of three tuples which can be ordered as s_1, s_2, s_3 in such a way that for every $s \in \mathrm{domain}(\phi_1)$ there exists $1 \leq i \leq 6$ such that $s = (s_1[i], s_2[i], s_3[i])$. Now, assume that $\langle \varGamma \rangle = BR$, $\phi_1 \notin \mathrm{pPol}(\Gamma)$, but that $\varPhi_1 \notin \langle \varGamma \rangle_{\nexists}$. Then there exists an n-ary $f \in \mathrm{pPol}(\Gamma)$ such that $f \notin \mathrm{pPol}(\{\Phi_1\})$, and $t_1, \ldots, t_n \in \varPhi_1$ such that $f(t_1, \ldots, t_n) \notin \varPhi_1$. Now consider the value $k = |\{t_1, \ldots, t_n\}|$, i.e., the number of distinct tuples in the sequence. If $n > k$ then it is known that there exists a closely related partial operation g of arity at most k such that $g \notin \mathrm{pPol}(\{\Phi_1\})$ [22], and we may therefore assume that $n = k \leq |\varPhi_1| = 3$. Assume first that $1 \leq n \leq 2$. Then, for every $t \in \{0, 1\}^n$ there exists i such that $(t_1[i], \ldots, t_n[i]) = t$. But then f must be a total operation which is not a projection, which is impossible since we assumed that $\langle \varGamma \rangle = BR$.

Hence, it must be the case that $n = 3$, and that $\{t_1, t_2, t_3\} = \Phi_1$. Assume without loss of generality that $t_1 = s_1$, $t_2 = s_2$, $t_3 = s_3$, and note that this implies that $\text{domain}(f) = \text{domain}(\phi_1)$ (otherwise f can simply be described as a permutation of ϕ_1). First, we will show that $f(0,0,0) = 0$ and that $f(1,1,1) = 1$. Indeed, if $f(0,0,0) = 1$ or $f(1,1,1) = 0$, it is possible to define a unary total f' as $f'(x) = f(x,x,x)$ which is not a projection since either $f'(0) = 1$ or $f'(1) = 0$. Second, assume there exists $(x,y,z) \in \text{domain}(f)$, distinct from $(0,0,0)$ and $(1,1,1)$, such that $f(x,y,z) \neq \phi_1(x,y,z)$. Without loss of generality assume that $(x,y,z) = (a,a,b)$ for $a, b \in \{0,1\}$, and note that $f(a,a,b) = a$ since $\phi_1(a,a,b) = b$. If also $f(b,b,a) = a$ it is possible to define a binary total operation $f'(x,y) = f(x,x,y)$ which is not a projection, therefore we have that $f(b,b,a) = b$. We next consider the values taken by f on the tuples (b,a,a) and (a,b,b). If $f(b,a,a) = f(a,b,b)$ then we can again define a total, binary operation which is not a projection, therefore it must hold that $f(b,a,a) \neq f(a,b,b)$. However, regardless of whether $f(b,a,a) = b$ or $f(b,a,a) = a$, f must be a partial projection. This contradicts the assumption that $f \notin \text{pPol}(\{\Phi_1\})$, and we conclude that $\Phi_1 \in \langle \Gamma \rangle_{\nexists}$. □

We will shortly use Lemma 16 to give a reduction from the Vertex Cover problem, since it is known that Vertex Cover does not admit a kernel with $O(n^{2-\varepsilon})$ edges for any $\varepsilon > 0$, unless NP $\subseteq$ co-NP/poly [5]. For each n and k let $H_{n,k}$ denote the relation $\{(b_1, \ldots, b_n) \in \{0,1\}^n \mid b_1 + \ldots + b_n = k\}$.

Lemma 17. *Let Γ be a constraint language. If $\langle \Gamma \rangle = BR$ then Γ can pp-define $H_{n,k}$ with $O(n+k)$ constraints and $O(n+k)$ existentially quantified variables.*

Proof. We first observe that one can recursively design a circuit consisting of fan-in 2 gates which computes the sum of n input gates as follows. At the lowest level, we split the input gates into pairs and compute the sum for each pair, producing an output of 2 bits for each pair. At every level i above that, we join each pair of outputs from the previous level, of i bits each, into a single output of $i+1$ bits which computes their sum. This can be done with $O(i)$ gates by chaining full adders. Finally, at level $\lceil \log_2 n \rceil$, we will have computed the sum. The total number of gates will be $\sum_{i=1}^{\lceil \log_2 n \rceil} (\frac{n}{2^i}) \cdot O(i)$, which sums to $O(n)$. Let $z_1, \ldots, z_{\log_2 n}$ denote the output gates of this circuit. By a standard Tseytin transformation we then obtain an equisatisfiable 3-SAT instance with $O(n)$ clauses and $O(n)$ variables. For each $1 \leq i \leq \log_2 n$, add the unary constraint $(z_i = k_i)$, where k_i denotes the ith bit of k written in binary. Each such constraint can be pp-defined with $O(1)$ existentially quantified variables over Γ. We then pp-define each 3-SAT clause in order to obtain a pp-definition of R over Γ, which in total only requires $O(n)$ existentially quantified variables. This is possible since if $\langle \Gamma \rangle = BR$ then Γ can pp-define every Boolean relation. □

Theorem 18. *Let Γ be a finite Boolean constraint language such that $\langle \Gamma \rangle = BR$ and $\phi_1 \notin \text{pPol}(\Gamma)$. Then $\text{SAT}(\Gamma)$ does not have a kernel of size $O(n^{2-\varepsilon})$ for any $\varepsilon > 0$, unless NP $\subseteq$ co-NP/poly.*

Proof. We will give a reduction from Vertex Cover parameterized by the number of vertices to $\mathrm{SAT}(\Gamma \cup \{\Phi_1\})$, which via Theorem 4 and Lemma 16 has a reduction to $\mathrm{SAT}(\Gamma)$ which does not increase the number of variables. Let (V, E) be the input graph and let k denote the maximum size of the cover. First, introduce two variables x_v and x'_v for each $v \in V$, and one variable y_i for each $1 \leq i \leq k$. Furthermore, introduce two variables x and y. For each edge $\{u, v\} \in E$ introduce a constraint $\Phi_1(x_u, x'_v, x'_u, x_v, x, y)$, and note that this enforces the constraint $(x_u \vee x_v)$. Let $\exists z_1, \ldots, z_m.\phi(x_1, \ldots, x_{|V|}, y_1, \ldots, y_k, z_1, \ldots, z_m)$ denote the pp-definition of $H_{|V|+k,k}$ over Γ where $m \in O(k + |V|)$, and consisting of at most $O(k + |V|)$ constraints. Such a pp-definition must exist according to Lemma 17. Drop the existential quantifiers and add the constraints of $\phi(x_1, \ldots, x_{|V|}, y_1, \ldots, y_k, z_1, \ldots, z_m)$. Let (V', C) denote this instance of $\mathrm{SAT}(\Gamma \cup \{\Phi_1\})$. Assume first that (V, E) has a vertex cover of size $k' \leq k$. We first assign x the value 0 and y the value 1. For each v in this cover assign x_v the value 1 and x'_v the value 0. For any vertex not included in the cover we use the opposite values. Then set $y_1, \ldots, y_{k-k'}$ to 1 and $y_{k-k'+1}, \ldots, y_k$ to 0. For the other direction, assume that (V', C) is satisfiable. For any x_v variable assigned 1 we then let v be part of the vertex cover. Since $x_1 + \ldots + x_{|V|} + y_1 + \ldots + y_k = k$, the resulting vertex cover is smaller than or equal to k. □

For example, let $R^k = \{(b_1, \ldots, b_k) \in \{0, 1\}^k \mid b_1 + \ldots + b_k \in \{1, 2\} \pmod 6\}$ and let $P = \{R^k \mid k \geq 1\}$. The kernelization status of $\mathrm{SAT}(P)$ was left open in Jansen and Pieterse [12], and while a precise upper bound seems difficult to obtain, we can easily prove that this problem does not admit a kernel of linear size, unless NP $\subseteq$ co-NP/poly. Simply observe that $(0, 0, 1), (0, 1, 1), (0, 1, 0) \in R^3$ but $\phi_1((0, 0, 1), (0, 1, 1), (0, 1, 0)) = (0, 0, 0) \notin R^3$. The result then follows from Theorem 18. At this stage, it might be tempting to conjecture that $\phi_1 \in \mathrm{pPol}(\Gamma)$ is also a sufficient condition for a Maltsev embedding. We can immediately rule this out by finding a relation R and a universal partial Maltsev operation ϕ such that R is invariant under ϕ_1 but not under ϕ. For example, let q be the 9-ary function defined by $u(u(x_1, x_2, x_3), u(x_4, x_5, x_6), u(x_7, x_8, x_9))$. Then we by computer experiments have verified that there exists a relation R of cardinality 9, invariant under ϕ_1 but not under $q_{|\mathbb{B}}$ [21].

5 Concluding Remarks and Future Research

We have studied kernelization properties of SAT and CSP with tools from universal algebra. We focused on problems with linear kernels, and showed that a CSP problem has a kernel with $O(n)$ constraints if it can be embedded into a CSP problem preserved by a Maltsev operation; thus extending previous results in this direction. On the other hand, we showed that a SAT problem not preserved by a partial Maltsev operation does not admit such a kernel, unless NP $\subseteq$ co-NP/poly. This shows that the algebraic approach is viable for studying such fine-grained kernelizability questions. Our work opens several directions for future research.

A Dichotomy Theorem for Linear Kernels? Our results suggest a possible dichotomy theorem for the existence of linear kernels for SAT problems. However, two gaps remain towards such a result. On the one hand, we proved that if Γ is preserved by the universal partial Maltsev operations then it admits a Maltsev embedding over an infinite domain. However, the kernelization algorithm only works for finite domains. Does the existence of an infinite-domain Maltsev embedding for a finite language imply the existence of a Maltsev embedding over a finite domain? Alternatively, can the algorithms be adjusted to work for languages with infinite domains, since D_∞ is finitely generated in a simple way? On the other hand, we only have necessity results for ϕ_1 out of an infinite set of conditions for the positive results. Is it true that every universal partial Maltsev operation is a partial polymorphism of every language with a linear kernel, or do there exist SAT problems with linear kernels that do not admit Maltsev embeddings?

The Algebraic CSP Dichotomy Conjecture. Several solutions to the CSP dichotomy conjecture have been announced [1,26,27]. If correct, these algorithms solve CSP(Γ) in polynomial time whenever Γ is preserved by a *Taylor term*. One can then define the concept of a Taylor embedding, which raises the question of whether the proposed algorithms can be modified to construct polynomial kernels. More generally, when can an operation f such that CSP(Inv({f})) is tractable be used to construct improved kernels? On the one hand, one can prove that *k-edge operations*, which are generalized Maltsev operations, can be used to construct kernels with $O(n^{k-1})$ constraints via a variant of the *few subpowers algorithm*. On the other hand, it is known that relations invariant under *semilattice operations* can be described as generalized Horn formulas, but it is not evident how this property could be useful in a kernelization procedure.

Acknowledgements. We thank the anonymous reviewers for several helpful suggestions. The first author is supported by the DFG-funded project "Homogene Strukturen, Bedingungserfüllungsprobleme, und topologische Klone" (Project number 622397).

References

1. Bulatov, A.: A dichotomy theorem for nonuniform CSPs. CoRR, abs/1703.03021 (2017)
2. Bulatov, A., Dalmau, V.: A simple algorithm for Mal'tsev constraints. SICOMP **36**(1), 16–27 (2006)
3. Bulatov, A., Jeavons, P., Krokhin, A.: Classifying the complexity of constraints using finite algebras. SICOMP **34**(3), 720–742 (2005)
4. Dalmau, V., Jeavons, P.: Learnability of quantified formulas. TCS **306**(1–3), 485–511 (2003)
5. Dell, H., van Melkebeek, D.: Satisfiability allows no nontrivial sparsification unless the polynomial-time hierarchy collapses. J. ACM **61**(4), 23:1–23:27 (2014)
6. Dyer, M., Richerby, D.: An effective dichotomy for the counting constraint satisfaction problem. SICOMP **42**(3), 1245–1274 (2013)

7. Feder, T., Vardi, M.: The computational structure of monotone monadic SNP and constraint satisfaction: a study through datalog and group theory. SICOMP **28**(1), 57–104 (1998)
8. Geiger, D.: Closed systems of functions and predicates. Pac. J. Math. **27**(1), 95–100 (1968)
9. Goldstern, M., Pinsker, M.: A survey of clones on infinite sets. Algebra Univers. **59**(3), 365–403 (2008)
10. Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity? J. Comput. Syst. Sci. **63**, 512–530 (2001)
11. Jansen, B.M.P., Pieterse, A.: Sparsification upper and lower bounds for graphs problems and not-all-equal SAT. In: Proceedings of IPEC 2015, Patras, Greece (2015)
12. Jansen, B.M.P., Pieterse, A.: Optimal sparsification for some binary CSPs using low-degree polynomials. In: Proceedings of MFCS 2016, vol. 58, pp. 71:1–71:14 (2016)
13. Jeavons, P.: On the algebraic structure of combinatorial problems. TCS **200**, 185–204 (1998)
14. Jeavons, P., Cohen, D., Gyssens, M.: A unifying framework for tractable constraints. In: Montanari, U., Rossi, F. (eds.) CP 1995. LNCS, vol. 976, pp. 276–291. Springer, Heidelberg (1995). doi:10.1007/3-540-60299-2_17
15. Jeavons, P., Cohen, D., Gyssens, M.: Closure properties of constraints. JACM **44**(4), 527–548 (1997)
16. Jonsson, P., Lagerkvist, V., Nordh, G., Zanuttini, B.: Strong partial clones and the time complexity of SAT problems. JCSS **84**, 52–78 (2017)
17. Kratsch, S., Marx, D., Wahlström, M.: Parameterized complexity and kernelizability of max ones and exact ones problems. TOCT **8**(1), 1 (2016)
18. Kratsch, S., Wahlström, M.: Preprocessing of min ones problems: a dichotomy. In: Abramsky, S., Gavoille, C., Kirchner, C., Meyer auf der Heide, F., Spirakis, P.G. (eds.) ICALP 2010. LNCS, vol. 6198, pp. 653–665. Springer, Heidelberg (2010). doi:10.1007/978-3-642-14165-2_55
19. Krokhin, A.A., Marx, D.: On the hardness of losing weight. ACM Trans. Algorithms **8**(2), 19 (2012)
20. Lagerkvist, V., Wahlström, M.: The power of primitive positive definitions with polynomially many variables. JLC **27**, 1465–1488 (2016)
21. Lagerkvist, V., Wahlström, M.: Kernelization of constraint satisfaction problems: a study through universal algebra. ArXiv e-prints, June 2017
22. Lagerkvist, V., Wahlström, M., Zanuttini, B.: Bounded bases of strong partial clones. In: Proceedings of ISMVL 2015 (2015)
23. Lau, D.: Function Algebras on Finite Sets: Basic Course on Many-Valued Logic and Clone Theory (Springer Monographs in Mathematics). Springer, New York (2006)
24. Marx, D.: Parameterized complexity of constraint satisfaction problems. Comput. Complex. **14**(2), 153–183 (2005)
25. Nemhauser, G.L., Trotter, L.E.: Vertex packings: structural properties and algorithms. Math. Program. **8**(1), 232–248 (1975)
26. Rafiey, A., Kinne, J., Feder, T.: Dichotomy for digraph homomorphism problems. CoRR, abs/1701.02409 (2017)
27. Zhuk, D.: The proof of CSP dichotomy conjecture. CoRR, abs/1704.01914 (2017)