Bounded Bases of Strong Partial Clones

Victor Lagerkvist*, Magnus Wahlström†, Bruno Zanuttini‡
*Department of Computer and Information Science, Linköpings Universitet, Sweden,
Email: victor.lagerkvist@liu.se
†Department of Computer Science, Royal Holloway, University of London, Great Britain
Email: Magnus.Wahlstrom@rhul.ac.uk
‡GREYC, Normandie Université, UNICAEN, CNRS, ENSICAEN, France,
Email: bruno.zanuttini@unicaen.fr

***Abstract*—Partial clone theory has successfully been applied to study the complexity of the *constraint satisfaction problem* parameterized by a set of relations (CSP(Γ)). Lagerkvist & Wahlström (ISMVL 2014) however shows that the partial polymorphisms of Γ (pPol(Γ)) cannot be finitely generated for finite, Boolean Γ if CSP(Γ) is NP-hard (assuming P$\neq$NP). In this paper we consider stronger closure operators than functional composition which can generate pPol(Γ) from a finite set of partial functions, a *bounded base*. Determining bounded bases for finite languages provides a complete characterization of their partial polymorphisms and we provide such bases for k-SAT and 1-in-k-SAT.**

I. Introduction

A *clone* is a composition-closed set of functions containing all projection functions, *i.e.*, all functions π_i^n of the form $\pi_i^n(x_1, \ldots, x_i, \ldots, x_n) = x_i$. A *strong partial clone* is a composition-closed set of partial functions containing all partial projection functions. Clones and strong partial clones can equivalently well be described through sets of relations. We say that a function f is a *polymorphism* of a set of relations Γ if $f(t_1, \ldots, t_n) \in R$ for all $R \in \Gamma$ and all $t_1, \ldots, t_n \in R$, where f is applied component-wise to $t_1, \ldots, t_n$. Let $\mathrm{Pol}(\Gamma)$ (respectively $\mathrm{pPol}(\Gamma)$) denote the (strong partial) clone consisting of all (partial) polymorphisms of Γ. From a practical point of view clone theory has been successfully applied to study the complexity of, *e.g.*, the constraint satisfaction problem over a set of relations Γ (CSP(Γ)). The set Γ is in this context referred to as a *constraint language*. The CSP problem is well-studied and can for instance be used to model classical NP-hard problems such as κ-SAT and κ-coloring. It is known (1) that if $\mathrm{Pol}(\Gamma) \subseteq \mathrm{Pol}(\Delta)$ then CSP(Δ) is polynomial-time reducible to CSP(Γ) [4] and (2) that if $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(\Delta)$ and CSP(Γ) is solvable in time $O(c^n)$, then CSP(Δ) is also solvable in time $O(c^n)$ [5]. Strong partial clones are therefore useful when comparing the time complexity between NP-hard problems, and have for instance been used to find the computationally easiest NP-hard CSP problem for finite Boolean languages [5].

There are two kinds of clone theoretical investigations that are particularly interesting when considering CSP problems. First, determining the lattice of strong partial clones on a given domain, so as to obtain a complete understanding of the relative complexity between CSP problems on this domain. Second, given a set of relations Γ, characterizing the strong partial clone $\mathrm{pPol}(\Gamma)$, by finding a base of partial functions that generate this strong partial clone. Such a base could then, *e.g.*, be used to efficiently check whether $\mathrm{pPol}(\Gamma) \subseteq \mathrm{pPol}(\Delta)$. It should not come as a surprise that both of these questions are difficult. For the first case it is known that in contrast to Post's lattice of Boolean clones, the cardinality of the lattice of strong partial clones is uncountably infinite even for the Boolean domain [1], and for the second case, Lagerkvist & Wahlström [6] prove that for any finite Boolean set of relations Γ such that CSP(Γ) is NP-hard (assuming P$\neq$NP), $\mathrm{pPol}(\Gamma)$ is of infinite order and cannot be finitely generated. This should be compared to Post's lattice where every clone is of finite order. Hence the step from total polymorphisms to partial polymorphisms leads to interesting applications but at the same time makes reasoning much more complex.

In this paper we tackle the problem of determining bases for strong partial clones by considering stronger notions of closure than composition of partial functions. After introducing the notions needed for the subsequent treatment (Section II), we show that the strong partial clone corresponding to any finite language can be characterized by a finite set of functions, which we call the *bounded base* of the language (Section III). Next we prove that the bounded base of a finite constraint language is expressive enough to generate all functions in $\mathrm{pPol}(\Gamma)$, by defining an operator stronger than functional composition which generates the strong partial clone from the bounded base (Section IV). Finally, we turn to the problem of determining bounded bases of finite Boolean constraint languages (Section V). First we give some general results regarding Boolean constraint languages that contain additional *sign patterns*, *i.e.*, when constraints can have negated arguments, and prove that in many cases the partial polymorphisms of such languages can be described in terms of simpler relations. Then we provide descriptions of the bounded bases for 1-in-κ-SAT and κ-SAT which, thanks to the notions introduced in this paper, have a particularly simple form. We believe that these bases can be used as a starting point to determine bounded bases of other constraint languages, in order to get a better understanding of the seemingly large difference in complexity between various NP-hard CSP problems.

II. Preliminaries

An n-ary partial function f on $D \subseteq \mathbb{N}$ is a map $f : X \to D$ where $X \subseteq D^n$ ($n \geq 1$), that is, a function that is allowed to be undefined for some sequences of arguments. Throughout this paper we always assume that D is finite. For a partial function $f : X \to D$, $X \subseteq D^n$, we let $\mathrm{domain}(f) = X$ and $\mathrm{ar}(f) = n$. If $u = (x_1, \ldots, x_n) \in \mathrm{domain}(f)$ we often write $f(u)$ instead of $f(x_1, \ldots, x_n)$. For a finite set of partial functions we let $\mathrm{ar}(F) = \max_{f \in F}(\mathrm{ar}(f))$. The partial function

g is said to be a *subfunction* of the partial function f if $\text{domain}(g) \subseteq \text{domain}(f)$ and $g(x_1,\ldots,x_n) = f(x_1,\ldots x_n)$ for all $(x_1,\ldots,x_n) \in \text{domain}(g)$. We also say that g is f *restricted to* X. Finally, a set of partial functions F is said to be *strong* if for all (partial or total) functions $f \in F$ and all subfunctions g of f, g is also in F.

Given a relation $R \subseteq D^\kappa$ ($\kappa \geq 1$) over $D \subseteq \mathbb{N}$ we let $\#\text{tuples}(R) = |R|$, *i.e.*, the number of tuples in the relation, and $\text{ar}(R) = \kappa$ be its arity. For a κ-ary tuple t and $1 \leq i \leq \kappa$, we write $t[i]$ for the ith element of t. For $\kappa \geq 0$ and $c \in \{0,1\}$, we write $\vec{c}^\kappa$ for $(c,\ldots,c)$ (κ times), *i.e.*, for a κ-ary sequence of 0's or 1's. If κ is clear from the context we simply write $\vec{c}$. If R is a κ-ary relation and $t_1,\ldots,t_n \in R$ a sequence of n tuples then we let $\mathit{Cols}(t_1,\ldots,t_n)$ be the sequence $\big((t_1[1],\ldots,t_n[1]),\ldots,(t_1[\kappa],\ldots,t_n[\kappa])\big)$. In other words, $\mathit{Cols}(t_1,\ldots,t_n)$ are the columns in the $n \times \kappa$ matrix formed by letting each element $a_{i,j} = t_i[j]$. For instance, $\mathit{Cols}((0,1,1),(1,1,1))$ is the sequence of tuples $((0,1),(1,1),(1,1))$. Whenever convenient we also use $\mathit{ColsSet}(t_1,\ldots,t_\kappa)$ for the set (instead of a sequence) $\{(t_1[1],\ldots,t_n[1]),\ldots,(t_1[\kappa],\ldots,t_n[\kappa])\}$.

A set of relations Γ is known as a *constraint language*, or simply *language*. Given a finite language Γ we let $\#\text{tuples}(\Gamma) = \max_{R\in\Gamma}(\#\text{tuples}(R))$ and $\text{ar}(\Gamma) = \max_{R\in\Gamma}(\text{ar}(R))$. Note that $\#\text{tuples}(\Gamma) \leq |D|^{\text{ar}(\Gamma)}$ always holds. A relation R is said to be *column-irredundant* if for any two indexes $i,j \leq \text{ar}(R)$, there is a tuple $t \in R$ with $t[i] \neq t[j]$. A constraint language Γ is said to be column-irredundant if all relations in Γ are column-irredundant. Clearly, if Γ is not column-irredundant, it can easily be transformed into a column-irredundant language resulting in a CSP problem with equivalent complexity. Observe that if Γ is a finite column-irredundant constraint language over D, then $\text{ar}(\Gamma) \leq |D|^{\#\text{tuples}(\Gamma)}$ holds.

As a convention, we use the Greek letters κ and λ to denote arities of relations and μ and ν for numbers of tuples in relations. For functions we instead use Latin letters n and m for arities and k and ℓ for size of domains. We denote tuples in relations by t, and tuples in the domain of functions by u.

A. Total and Partial Clones

Let $D \subseteq \mathbb{N}$. For $n \in \mathbb{N}$ and $i \in \{1,\ldots,n\}$, we write π_i^n for the *projection* function defined by $\pi_i^n(x_1,\ldots,x_i,\ldots,x_n) = x_i$ for all $(x_1,\ldots,x_n) \in D^n$. We write Π_D for the set of all projection functions over $D \subseteq \mathbb{N}$, and Π_D^p for the set of all partial functions that are subfunctions of projections. If f is an n-ary function and $g_1,\ldots,g_n$ are m-ary functions, then the *composition* of f and $g_1,\ldots,g_n$, denoted $f\circ(g_1,\ldots,g_n)$, is the m-ary function defined by $(f \circ (g_1,\ldots,g_n))(x_1,\ldots,x_m) = f(g_1(x_1,\ldots,x_m),\ldots,g_n(x_1,\ldots,x_m))$. Composition of partial functions is defined in the same way but the result is only defined when all involved function applications are defined. A *clone* over D is then a set of functions C such that (1) $\mathsf{C} \supseteq \Pi_D$ and (2) C is closed under composition. Similarly, a set of partial functions C is a *strong partial clone* if (1) $\mathsf{C} \supseteq \Pi_D^p$ and (2) C is closed under composition. Note that such C contains all subfunctions of its (partial or total) functions [9], hence the term *strong*. For a set of total functions F and a set of partial functions F' we let $[F]$ be the smallest clone containing F and $[F']_s$ be the smallest strong partial clone containing F'. The sets F and F' are said to be *bases* of $[F]$ or $[F']_s$. The *order* of a clone or a strong partial clone is the cardinality of its smallest base. In particular, we are interested in whether this order is finite or infinite.

Example 1: For any Boolean projection function π_i^n we have that $[\{\pi_i^n\}]$ is the smallest element in the lattice of Boolean clones, consisting only of projections. If we instead let $\text{neg}(0) = 1$ and $\text{neg}(1) = 0$ then $[\{\text{neg}\}]$ is the clone consisting of compositions of unary negation and projection functions. Observe that $[\{\pi_i^n\}] \subset [\{\text{neg}\}]$. From Post's classification of Boolean clones [8] we also know that $[\{\text{neg}\}]$ is *minimal*, in the sense that there is no Boolean clone C with $[\{\pi_i^n\}] \subset C \subset [\{\text{neg}\}]$.

Clones and strong partial clones can also be characterized by sets of relations. An n-ary function $f : D^n \to D$ ($n \geq 1$) is said to be a *polymorphism* of a κ-ary relation $R \subseteq D^\kappa$ if $f(t_1,\ldots,t_n) \in R$ for every sequence $t_1,\ldots,t_n \in R$ (possibly with repetitions), where $f(t_1,\ldots,t_n)$ denotes the tuple $\big(f(t_1[1],\ldots,t_n[1]),\ldots,f(t_1[\kappa],\ldots,t_n[\kappa])\big)$, *i.e.*, f applied to every element in $\mathit{Cols}(t_1,\ldots,t_n)$. Similarly, a partial function f is said to be a *partial polymorphism* of R if $f(t_1,\ldots,t_n) \in R$ for every sequence $(t_1,\ldots,t_n)$ of tuples of R such that all components of $f(t_1,\ldots,t_n)$ are defined. In both cases R is said to be *invariant* under f, and that f *preserves* R, and we let $\text{Inv}(F)$ denote the set of all relations that are invariant under the set of (total or partial) functions F. Dually, for a set of relations Γ we let $\text{Pol}(\Gamma)$ denote the set of all polymorphisms of Γ, and $\text{pPol}(\Gamma)$ denote the set of all partial polymorphisms of Γ. We then have the following *Galois connections* relating the operators together.

Theorem 1 ([2], [3], [9]): Let Γ, Γ' be constraint languages. Then $\text{Inv}(\text{Pol}(\Gamma')) \subseteq \text{Inv}(\text{Pol}(\Gamma)) \iff \text{Pol}(\Gamma) \subseteq \text{Pol}(\Gamma')$ and $\text{Inv}(\text{pPol}(\Gamma')) \subseteq \text{Inv}(\text{pPol}(\Gamma)) \iff \text{pPol}(\Gamma) \subseteq \text{pPol}(\Gamma')$.

Sets of the form $\text{Inv}(\text{Pol}(\Gamma))$ are usually referred to as *co-clones*, and sets of the form $\text{Inv}(\text{pPol}(\Gamma))$ as *weak partial co-clones*, or *weak systems* [11].

B. The Constraint Satisfaction Problem

Given $D \subseteq \mathbb{N}$, the *constraint satisfaction problem* over a constraint language Γ ($\text{CSP}(\Gamma)$) takes as input a set V of variables and a set C of constraint applications of the form $R(x_1,\ldots,x_{\text{ar}(R)})$, with $R \in \Gamma$ and $x_1,\ldots,x_{\text{ar}(R)} \in V$. The question is whether there is a function $s : V \to D$ such that $(s(x_1),\ldots,s(x_{\text{ar}(R)})) \in R$ for all $R(x_1,\ldots,x_{\text{ar}(R)})$ in C. For the Boolean domain, this problem is typically denoted by $\text{SAT}(\Gamma)$. Jeavons [4] proved that for any finite constraint language Γ, the computational complexity of $\text{CSP}(\Gamma)$ is determined by $\text{Pol}(\Gamma)$ up to polynomial-time reductions. With this fact and Post's classification of Boolean clones we can rephrase Schaefer's dichotomy theorem [10] for SAT as follows: $\text{SAT}(\Gamma)$ is NP-hard if and only if $\text{Pol}(\Gamma) \subseteq [\{\text{neg}\}]$ where $\text{neg}(0) = 1$ and $\text{neg}(1) = 0$. Jonsson *et al.* [5] give a classification similar to Jeavons', by showing that if $\text{pPol}(\Gamma) \subseteq \text{pPol}(\Delta)$ and $\text{CSP}(\Gamma)$ is solvable in time $O(c^n)$, then $\text{CSP}(\Delta)$ is also solvable in time $O(c^n)$, with $n = |V|$. Unfortunately, describing the partial polymorphisms of finite languages is a very complex task since Lagerkvist

& Wahlström [6] proved that $\mathrm{pPol}(\Gamma)$ is of infinite order whenever $\mathrm{CSP}(\Gamma)$ is NP-hard and Γ is a finite Boolean language (assuming P≠NP).

III. Bounded Bases

In this section we give our first contribution, namely, for the strong partial clone $\mathrm{pPol}(\Gamma)$ associated to any finite constraint language Γ, we define a finite set of partial functions which completely determines $\mathrm{pPol}(\Gamma)$. The construction of $\mathrm{pPol}(\Gamma)$ from its bounded base will be investigated in Section IV. We start with an investigation of how functions with small arities characterize the set of all partial polymorphisms of Γ (of any arity). Intuitively, if f has "high" arity, we will consider sets of variables which have the same value in all the tuples over which f is defined, and consider the *contraction* of f obtained by removing such duplicates.

Definition 1 (redundant variable): Let g be an n-ary partial function over arguments $x_1, \ldots, x_n$. A variable x_i is said to be *redundant* with x_j for g ($j \neq i$), if $u[i] = u[j]$ for all $u \in \mathrm{domain}(g)$.

If f is an n-ary function and σ is a permutation on $[n]$, then we denote by $g \circ \sigma$ the function defined by $g \circ \sigma(x_1, \ldots, x_n) = g(x_{\sigma(1)}, \ldots, x_{\sigma(n)})$ for all $x_1, \ldots, x_n$.

Definition 2 (contraction, stretching): Let g be an n-ary partial function over arguments $x_1, \ldots, x_n$. If there is a permutation σ of $[n]$ and a function γ from $\{m+1, \ldots, n\}$ to $[m]$ such that for all $i \in \{m+1, \ldots, n\}$, x_i is redundant with $x_{\gamma(i)}$ for $g \circ \sigma$, then the *m-contraction* of g with respect to σ and γ is the m-ary function g' defined by $g'(x_1, \ldots, x_m) = g \circ \sigma(x_1, \ldots, x_m, x_{\gamma(m+1)}, \ldots, x_{\gamma(n)})$. In this case g is also said to be an *n-stretching* of g' with respect to σ and γ.

If g' is the m-contraction of g with respect to some permutation σ and function γ, we simply say that g' is *an* m-contraction of g (and that g is *an* n-stretching of g'). If an n-ary function f has no m-contractions for some $m < n$, then we say that f is *uncontractable* to arity m. For instance, if f is an n-ary function defined only on tuples of Hamming weight 1, then it is uncontractable to arity $n-1$ since for all variables $x_i \neq x_j$, there is a tuple in the domain of f which assigns 1 to x_i and 0 to x_j and hence, x_i is not redundant with x_j for f. Note that if f is uncontractable to arity m, then it is *a fortiori* uncontractable to any arity $m' \leq m$.

Given a set of partial functions F, we write $\mathcal{S}_n(F)$ for the set of all n-ary partial functions that are an n-stretching of some $f \in F$, and $\mathcal{C}_m(F)$ for the set of all m-ary partial functions that are an m-contraction of some $f \in F$. For $m \leq n$ we let $\mathcal{U}_n^m$ be the set of all n-ary partial functions uncontractable to arity m. The following is a simple but essential property. We omit the proof since it easily follows from the definitions.

Lemma 1: Let g' be an m-contraction of an n-ary partial function g. Then $[\{g'\}]_s = [\{g\}]_s$ holds.

The notions of contraction and stretching will allow us to bound the arities of the functions needed to characterize a strong partial clone. Dually, we now introduce the notion of a k-restriction, which will allow us to bound the size of the domain of these functions.

Definition 3 (k-restriction): Let f be an n-ary partial function, and k be an integer with $|\mathrm{domain}(f)| \geq k$. An n-ary function g is said to be a *k-restriction* of f if g is a subfunction of f and $|\mathrm{domain}(g)| = k$ holds.

For a partial function f and an integer k, we write $\mathcal{R}_k(f)$ for the set of all functions that are a k-restriction of f if $|\mathrm{domain}(f)| \geq k$ holds, and otherwise we define $\mathcal{R}_k(f) = \emptyset$. If F is a set of partial functions, then $\mathcal{R}_k(F)$ denotes $\bigcup_{f \in F} \mathcal{R}_k(f)$. The following lemma, which relates partial polymorphisms with their restrictions and contractions, is the cornerstone of our study.

Lemma 2: Let R be a relation, let $\kappa \geq \mathrm{ar}(R)$ and $n > \mu \geq \#\mathrm{tuples}(R)$, and let f be an n-ary partial function. Then $f \in \mathrm{pPol}(\{R\})$ holds if and only if $\mathcal{C}_\mu(\mathcal{R}_{\kappa'}(f)) \subseteq \mathrm{pPol}(\{R\})$ holds for all $\kappa' \leq \kappa$.

Proof: First assume $f \in \mathrm{pPol}(\{R\})$. Then $g \in \mathrm{pPol}(\{R\})$ holds for all subfunctions g of f, and all μ-contractions g' of such g's are in $\mathrm{pPol}(\{R\})$ by Lemma 1 and Theorem 1. For the other direction, towards contradiction assume $f \notin \mathrm{pPol}(\{R\})$. Then there is a sequence of tuples $t_1, \ldots, t_n \in R$ with $f(t_1, \ldots, t_n) = (f(t_1[1], \ldots, t_n[1]), \ldots, f(t_1[\mathrm{ar}(R)], \ldots, t_n[\mathrm{ar}(R)])) \notin R$. Note that at least $n - \mu$ of these tuples must be repeated since we assume $n > \mu \geq \#\mathrm{tuples}(R)$. Let $\lambda \leq \mathrm{ar}(R) \leq \kappa$ be the number of distinct tuples in $Cols(t_1, \ldots, t_n)$, and let g be the λ-restriction g of f defined only on these tuples. Since all but at most μ tuples are repeated in $t_1, \ldots, t_n$ there is at least one μ-contraction g' of g with $g'(t_{i_1}, \ldots, t_{i_\mu}) = g(t_1, \ldots, t_n) = f(t_1, \ldots, t_n) \notin R$. This contradicts that all μ-contractions of all λ-restrictions of f are in $\mathrm{pPol}(\{R\})$. ∎

We immediately get the following generalization to constraint languages (instead of single relations). The proof follows easily by applying Lemma 2.

Corollary 1: Let Γ be a finite constraint language, let $\kappa \geq \mathrm{ar}(\Gamma)$, $n > \mu \geq \#\mathrm{tuples}(\Gamma)$, and let f be an n-ary partial function. Then $f \in \mathrm{pPol}(\Gamma)$ if and only if $\mathcal{C}_\mu(\mathcal{R}_{\kappa'}(f)) \subseteq \mathrm{pPol}(\Gamma)$ for all $\kappa' \leq \kappa$.

Corollary 1 implies that the partial polymorphisms of a finite constraint language can be derived from those with bounded arity and domain (with bounds depending on the language). We now define these to constitute the *bounded base* of the constraint language.

Definition 4 (bounded base): Let Γ be a finite constraint language with $\kappa = \mathrm{ar}(\Gamma)$ and $\mu = \#\mathrm{tuples}(\Gamma)$. The *bounded base* of Γ, $B(\Gamma)$, is defined to be

$$B(\Gamma) = \{f \mid f \in \mathrm{pPol}(\Gamma), \mathrm{ar}(f) \leq \mu, |\mathrm{domain}(f)| \leq \kappa\}.$$

The following proposition shows that representing languages by their bounded bases suffices to distinguish languages with different expressivity. The proof is straightforward and omitted due to space constraints.

Proposition 1: Let Γ, Γ' be finite column-irredundant constraint languages with $B(\Gamma) = B(\Gamma')$. Then $\mathrm{pPol}(\Gamma) = \mathrm{pPol}(\Gamma')$ holds.

It may be the case that two different languages generate the same weak partial co-clone but have different bounded bases because their arities or numbers of tuples are different, but as we show in Section IV, we can still compare their expressivity using bounded bases.

IV. Closure of Bounded Bases

In this section we show how, for a finite constraint language Γ, the whole strong partial clone $\mathrm{pPol}(\Gamma)$ can be recovered from the functions in the finite base $B(\Gamma)$, which is done using a notion of closure stronger than functional composition. With this we investigate how bounded bases can be used to compare the expressivity of finite constraint languages. We first need some preliminary definitions to cope with the fact that the bounded base is a finite set.

Definition 5 (covering): Let G be a set of n-ary partial functions, and let $k \geq 1$. Then G is said to be *k-covering* if for all $U \subseteq \cup_{g \in G} \mathrm{domain}(g)$ satisfying $|U| \leq k$, there is a function $g \in G$ with $\mathrm{domain}(g) = U$.

We say that a set of n-ary partial functions G is *consistent* if for all $g, g' \in G$ and all $(x_1, \ldots, x_n) \in \mathrm{domain}(g) \cap \mathrm{domain}(g')$, it holds that $g(x_1, \ldots, x_n) = g'(x_1, \ldots, x_n)$. Consistency ensures that g can be chosen arbitrarily in the next definition.

Definition 6 (union): Let G be a consistent set of n-ary partial functions. The *union of* G, written $\mathrm{u}(G)$, is the n-ary partial function defined by $\mathrm{u}(G)(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)$ for all $(x_1, \ldots, x_n)$ such that some $g \in G$ is defined on $(x_1, \ldots, x_n)$, and undefined for other tuples.

Given a function f and $k \geq 1$, we define *the k-covering of* f as the set of functions $\mathcal{G}_k(f) = \bigcup_{k'=1}^{k} \mathcal{R}_{k'}(f)$. Observe that $\mathrm{u}(\mathcal{G}_k(f)) = f$ holds. Next recall that $\mathcal{S}_n(F)$ denotes the set of all functions that are an n-stretching of some function in F, and that $\mathcal{U}_n^m$ denotes the set of all n-ary partial functions that have no m-contractions. We are now ready to define our notion of closure.

Definition 7 (closure): Let F be a finite set of partial functions, and let $m = \mathrm{ar}(F)$. Let $k \geq 1, n \geq 1$ be integers. The *k, n-closure* of F, written $Cl_{k,n}(F)$, is the set of n-ary functions defined by $Cl_{k,n}(F) = \{\mathrm{u}(G) \mid G \subseteq F, \mathrm{ar}(G) = n$, G is k-covering and consistent$\}$ for $n \leq m$ and $Cl_{k,n}(F) = \{\mathrm{u}(G) \mid G \subseteq \mathcal{S}_n(F) \cup \mathcal{U}_n^m$ and G is k-covering and consistent$\}$ for $n > m$. The *k-closure* of F is defined to be $Cl_k(F) = \bigcup_{n=1}^{\infty} Cl_{k,n}(F)$.

Example 2: As a simple example let $R_0 = \{(0)\}$. Note that $\mathrm{pPol}(\{R_0\}) = [\{x_1 \wedge x_2, x_1 \oplus x_2\}]_s$ [7]. The bounded base $B(\{R_0\})$ consists of all unary $f \in \mathrm{pPol}(\{R_0\})$ satisfying $|\mathrm{domain}(f)| = 1$. Thus $B(\{R_0\}) = \{f_1, f_2, f_3\}$ where f_1, f_2, f_3 are defined as $f_1(0) = 0$, $f_2(1) = 1$, $f_3(1) = 0$, and undefined otherwise. To exemplify 1,2-closure we see that $x_1 \wedge x_2 = \mathrm{u}(G)(x_1, x_2)$ where $G = \{g_1, g_2, g_3, g_4\}$ and $g_1(0, 0) = 0$, $g_2(0, 1) = 0$, $g_3(1, 0) = 0$, $g_4(1, 1) = 1$, and undefined for all other values. Since $g_1 \in \mathcal{S}_2(\{f_1\})$, $g_4 \in \mathcal{S}_2(\{f_2\})$, and $g_2, g_3 \in \mathcal{U}_2^1$, it follows $x_1 \wedge x_2 \in Cl_{1,2}(\{f_1, f_2, f_3\})$.

Note that we slightly abuse the term "closure", since the k-closure operator fails to satisfy all properties normally associated with closure operators.

With this in hand, we can now show that our notion of closure captures exactly what we want, namely, that the $\mathrm{ar}(\Gamma)$-closure of the bounded base of Γ is exactly $\mathrm{pPol}(\Gamma)$.

Theorem 2: Let Γ be a finite constraint language. Then $Cl_{\mathrm{ar}(\Gamma)}(B(\Gamma)) = \mathrm{pPol}(\Gamma)$ holds.

Proof: Write $\kappa = \mathrm{ar}(\Gamma)$ and $\mu = \#\mathrm{tuples}(\Gamma)$, and first let $f \in Cl_\kappa(B(\Gamma))$ be an n-ary partial function. For $n \leq \mu$, by definition of closure the κ-covering $\mathcal{G}_\kappa(f)$ of f satisfies $\mathcal{G}_\kappa(f) \subseteq B(\Gamma) \subseteq \mathrm{pPol}(\Gamma)$. Since all relations in Γ are at most κ-ary, any application of f to tuples from such a relation is in fact an application of some f' in the κ-covering of f, hence f preserves Γ. Now for $n > \mu$ we have by definition of closure that for all $\kappa' \leq \kappa$ and all κ'-restrictions g of f, either g is uncontractable to arity μ or g is the stretching of a function g' in $B(\Gamma) \subseteq \mathrm{pPol}(\Gamma)$. In the latter case all μ-contractions of g are in $\mathrm{pPol}(\Gamma)$ by Lemma 1, and this also holds vacuously in the former case. Hence we get $f \in \mathrm{pPol}(\Gamma)$ by Corollary 1. Conversely, let $f \in \mathrm{pPol}(\Gamma)$ be an n-ary function. For $n \leq \mu$, by definition the κ-covering $\mathcal{G}_\kappa(f)$ of f consists of functions of arity n that are defined on at most κ tuples, and which are in $\mathrm{pPol}(\Gamma)$ as subfunctions of f. It follows that $\mathcal{G}_\kappa(f) \subseteq B(\Gamma)$ and hence, $f = \mathrm{u}(\mathcal{G}_\kappa(f))$ is in $Cl_\kappa(B(\Gamma))$. Finally, for $n > \mu$, let g be a function in the κ-covering $\mathcal{G}_\kappa(f)$ of f. We get $\mathcal{C}_\mu(g) \subseteq \mathrm{pPol}(\Gamma)$ from Corollary 1, and hence either $g \in \mathcal{U}_n^\mu$ holds or g has at least one μ-contraction g', which is in $\mathrm{pPol}(\Gamma)$ and hence in $B(\Gamma)$. In the latter case g is a stretching of g', and finally, $\mathcal{G}_\kappa(f) \subseteq \mathcal{S}_n(B(\Gamma)) \cup \mathcal{U}_n^\mu$, which concludes the proof. ■

To exemplify this result, let R_0 and f_1, f_2, f_3 be defined as in Example 2. From Theorem 2 we get $Cl_1(\{f_1, f_2, f_3\}) = \mathrm{pPol}(\{R_0\})$, *i.e.*, the set of all total or partial functions that are 0-reproducing. The proof of Theorem 2 also makes it clear why we need the sets $\mathcal{U}_n^\mu$ of uncontractable functions.

V. Determining Bounded Bases of Boolean Constraint Languages

We now turn to the problem of determining bounded bases of Boolean constraint languages. We start with a general construction for retrieving the partial polymorphisms of the language obtained from another one by allowing some arguments to be negated in constraint applications, and then determine bounded bases for two of the most studied languages corresponding to NP-hard SAT problems, namely monotone 1-in-κ-SAT and κ-SAT.

A. Sign Patterns

It is natural to ask how the complexity of CSP(Γ) is influenced by negated arguments in constraint applications. To handle this we extend Γ by additional relations, representing the cases where one or more arguments are negated. Formally, if R is a κ-ary Boolean relation and $s_1, \ldots, s_\kappa \in \{+, -\}$, then we let $R^{(s_1, \ldots, s_\kappa)} = \{(x_1^{s_1}, \ldots, x_\kappa^{s_\kappa}) \mid (x_1, \ldots, x_\kappa) \in R\}$, with $x_i^+ = x_i$ and $x_i^- = \overline{x_i}$. The tuple $(s_1, \ldots, s_\kappa)$ is called a *sign pattern* for R. As a shorthand we write $\overline{R} = R^{(-, \ldots, -)}$. For example, $R_{1/3}^{(+,-,-)}$ is $\{(0, 1, 0), (0, 0, 1), (1, 1, 1)\}$ and $\overline{R_{1/3}}$ is $\{(1, 1, 0), (1, 0, 1), (0, 1, 1)\}$. Interestingly, it is possible to describe $\mathrm{pPol}(\{\overline{R}\})$ and $\mathrm{pPol}(\{R, \overline{R}\})$ in terms of $\mathrm{pPol}(\{R\})$. For an n-ary Boolean partial function f, let the dual f^d be defined by $f^d(\overline{u}) = \overline{f(u)}$ for all $u \in \mathrm{domain}(f)$.

Proposition 2: Let R be Boolean relation. Then $\mathrm{pPol}(\{\overline{R}\}) = \{f^d \mid f \in \mathrm{pPol}(\{R\})\}$ and $\mathrm{pPol}(\{R, \overline{R}\}) = \{f \mid f, f^d \in \mathrm{pPol}(\{R\})\}$ hold.

The proof is not complicated and therefore omitted. The cases when one or more, but not all, arguments are negated are not as straightforward, and we need a few additional definitions. First, call a partial function f *self-dual* if for all $u \in \mathrm{domain}(f)$ we have $\bar{u} \in \mathrm{domain}(f)$ and $f(\bar{u}) = \overline{f(u)}$. Second, call a language Γ *closed under sign patterns* if for all $R \in \Gamma$ and $s_1, \ldots, s_{\mathrm{ar}(R)} \in \{+, -\}$, the relation $R^{(s_1, \ldots, s_{\mathrm{ar}(R)})}$ is also in Γ. Finally, call Γ *closed under fixing arguments* if for all $R \in \Gamma$, $1 \leq i \leq \mathrm{ar}(R)$, $c \in \{0, 1\}$, the relation $R_{x_i \leftarrow c} = \{(x_1, \ldots, x_{i-1}, c, x_{i+1}, \ldots, x_{\mathrm{ar}(R)}) \mid (x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_{\mathrm{ar}(R)}) \in R\}$ is in Γ. Recall that a function is c-reproducing, $c \in \{0, 1\}$, if it returns c when all arguments are c.

Theorem 3: Let R be a Boolean relation, Γ be the closure of $\{R\}$ under sign patterns, and Γ' be the closure of Γ under fixing arguments. If $\mathrm{Pol}(\Gamma) = [\Pi_{\{0,1\}}]$ then $\mathrm{pPol}(\Gamma') = \{f' \mid f'$ is a subfunction of some $f \in \mathrm{pPol}(\{R\}), f$ is 0-reproducing, 1-reproducing, and self-dual$\}$.

The proof is omitted due to space constraints. Proposition 2 and Theorem 3 provide sufficient conditions for when it is possible to describe $\mathrm{pPol}(\Gamma)$ through $\mathrm{pPol}(\{R\})$ if Γ is a sign pattern expansion of R. Naturally this also holds for $B(\Gamma)$, which means that bounded bases for these languages can be expressed in a particularly simple form.

B. 1-in-κ-SAT

In this section we determine the bounded base of $R_{1/\kappa} = \{(x_1, \ldots, x_\kappa) \mid x_1, \ldots, x_\kappa \in \{0, 1\}, \Sigma_{i=1}^{\kappa} x_i = 1\}$. First note that $\#\mathrm{tuples}(R_{1/\kappa}) = \kappa$, and that $\mathrm{CSP}(\{R_{1/\kappa}\})$ can be seen as an alternative formulation of the monotone 1-in-κ-SAT problem, hence it is NP-complete for every $\kappa \geq 3$.

We give a general characterization of the functions in the bounded base $B(\{R_{1/\kappa}\})$. For integers λ, κ' and a set of κ'-ary Boolean tuples U satisfying $|U| = \lambda$, we say that U is an *exact cover* of $\{1, \ldots, \kappa'\}$ if for all $i = 1, \ldots, \kappa'$, there is a unique $u \in U$ with $u[i] = 1$. For instance, for $\lambda = 2$ and $\kappa' = 3$, the set $U = \{(0, 1, 1), (1, 0, 0)\}$ is an exact cover of $\{1, \ldots, 3\}$, and for $\lambda = 4$ and $\kappa' = 3$, the set $U = \{(0, 0, 0), (0, 0, 1), (0, 1, 0), (1, 0, 0)\}$ is an exact cover of $\{1, \ldots, 3\}$. Observe that U is an exact cover of $\{1, \ldots, \kappa'\}$ if and only if $U \setminus \{\vec{0}\}$ and $U \cup \{\vec{0}\}$ are exact covers, and that the size $\lambda = |U|$ of an exact cover U of $\{1, \ldots, \kappa'\}$ satisfies either $\lambda \leq \kappa' + 1$ or $\lambda \leq \kappa'$, depending on whether U contains $\vec{0}$. In the forthcoming proposition we make use of the following two lemmas, which we state without proof.

Lemma 3: Let $\kappa \geq \kappa' \geq 1$. If $U \subseteq \{0, 1\}^{\kappa'}$ is an exact cover of $\{1, \ldots, \kappa'\}$ with $|U| = \kappa$, then $U = \mathit{ColsSet}(t_1, \ldots, t_{\kappa'})$ for some $t_1, \ldots, t_{\kappa'} \in R_{1/\kappa}$.

Lemma 4: Let $\kappa' \geq 1$. Then $U \subseteq \{0, 1\}^{\kappa'}$ is an exact cover of $\{1, \ldots, \kappa'\}$ with $(\vec{0}) \in U$ if and only if $U = \mathit{ColsSet}(t_1, \ldots, t_{\kappa'})$ for some $t_1, \ldots, t_{\kappa'} \in R_{1/\kappa}$ with $\kappa > \kappa'$.

Proposition 3: Let $\kappa \geq 3$. Then the bounded base of $R_{1/\kappa}$ is the set of all functions f with $\kappa' = \mathrm{ar}(f) \leq \kappa$ and defined on at most κ tuples, that satisfy one of the following:

1) for all exact covers U of $\{1, \ldots, \kappa'\}$ contained in $\mathrm{domain}(f)$ and with $\vec{0} \in U$ or $|U| = \kappa$, f restricted to U is a projection, or
2) $\vec{0} \in \mathrm{domain}(f)$, $|\mathrm{domain}(f)| = \kappa$, $\mathrm{domain}(f)$ is an exact cover of $\{1, \ldots, \kappa'\}$, $f(\vec{0}) = 1$, and $f(x_1, \ldots, x_{\kappa'}) = 0$ for all $(x_1, \ldots, x_{\kappa'}) \in \mathrm{domain}(f) \setminus \{\vec{0}\}$.

Proof: Let f be a function in $B(R_{1/\kappa})$. By definition of a bounded base, from $\mathrm{ar}(R_{1/\kappa}) = \#\mathrm{tuples}(R_{1/\kappa}) = \kappa$ we get $\mathrm{ar}(f), |\mathrm{domain}(f)| \leq \kappa$. We now distinguish two cases for the tuple $\vec{0}$. First assume $\vec{0} \notin \mathrm{domain}(f)$ or $f(\vec{0}) = 0$, and let $U \subseteq \mathrm{domain}(f)$ be an exact cover of $\{1, \ldots, \kappa'\}$ with $(\vec{0}) \in U$ or $|U| = \kappa$. Let T be the sequence of tuples from $R_{1/\kappa}$ satisfying $\mathit{ColsSet}(T) = U$. For $|U| = \kappa$ and $\vec{0} \notin U$, such T exists by Lemma 3, and it is easily seen that it has to contain each tuple of $R_{1/\kappa}$ exactly once. For the case when $\vec{0} \in U$ such T exists by Lemma 4 and $|\mathrm{domain}(f)| \leq \kappa$. Now because f preserves $R_{1/\kappa}$ by assumption, f applied columnwise to T must yield a tuple containing exactly one 1. Hence for $|U| = \kappa$ and $\vec{0} \notin U$ it is clear that f restricted to U is a projection, and for $\vec{0} \in U$, using $f(\vec{0}) = 0$ we get that $f(x_1, \ldots, x_{\kappa'})$ must be 1 for some $(x_1, \ldots, x_{\kappa'}) \in U \setminus \{\vec{0}\}$, hence again that f restricted to U is a projection. Finally, f is in Case (1). Now assume $f(\vec{0}) = 1$ and that f is not in Case (1). We first show that $|\mathrm{domain}(f)| = \kappa$ and $\mathrm{domain}(f)$ is an exact cover of $\{1, \ldots, \kappa'\}$. We first claim that $\mathrm{domain}(f)$ does not contain an exact cover of $\{1, \ldots, \kappa'\}$ of size less than $\kappa - 1$. Indeed, otherwise it follows from Lemma 4 that there exists a sequence $T = (t_{i_1}, \ldots, t_{i_{\kappa'}})$ of tuples from $R_{1/\kappa}$ satisfying $\mathit{ColsSet}(T) = U \cup \{\vec{0}\}$ and such that $\vec{0}$ occurs at least twice a column. Hence f applied columnwise to T yields a tuple containing at least two 1's and it follows that f does not preserve $R_{1/\kappa}$, a contradiction. With this in hand, since f is not in Case (1), $\mathrm{domain}(f)$ must contain an exact cover of $\{1, \ldots, \kappa'\}$, hence one of size $\kappa - 1$ or κ, say U. But for $|U| = \kappa - 1$, U cannot contain $\vec{0}$, since otherwise $U \setminus \{\vec{0}\}$ would be a cover of size less than $\kappa - 1$, which we have just shown to be impossible. Now since we have $\vec{0} \in \mathrm{domain}(f)$ we get that $U \cup \{\vec{0}\}$ is an exact cover of size κ, as desired. Hence $|\mathrm{domain}(f)| = \kappa$ and $\mathrm{domain}(f)$ is an exact cover of $\{1, \ldots, \kappa'\}$. Letting T be a sequence of tuples from $R_{1/\kappa}$ with $\mathit{ColsSet}(T) = \mathrm{domain}(f)$, we have that f applied columnwise to T must yield a tuple containing exactly one 1. From $f(\vec{0}) = 1$ we conclude that f is constantly 0 on $\mathrm{domain}(f) \setminus \{\vec{0}\}$. Finally, f is in Case (2).

We now show the reverse inclusion. Let T be a sequence of κ' tuples from $R_{1/\kappa}$ such that f is defined on $\mathit{ColsSet}(T)$. We show that applying f columnwise to T yields a tuple in $R_{1/\kappa}$, which is enough to show the claim. Since T is a sequence of tuples from $R_{1/\kappa}$ we have that $\mathit{ColsSet}(T)$ is an exact cover of $\{1, \ldots, \kappa'\}$, which either contains $\vec{0}$ or is of size κ (or both). Hence if f is in Case (1), then its restriction to U has to be a projection. Now, if f is in Case (2), $\vec{0}$ cannot be repeated in $\mathit{Cols}(T)$ since otherwise $\mathrm{domain}(f) \setminus \{\vec{0}\}$ would be a proper superset of $\mathit{ColsSet}(T) \setminus \{0\}$ (by $|\mathrm{domain}(f)| = \kappa$) and hence, could not be an exact cover of $\{1, \ldots, \kappa'\}$. On the other hand, by definition of $R_{1/\kappa}$, no other column can be repeated in $\mathit{Cols}(T)$, hence from $|\mathrm{domain}(f)| = \kappa$ we conclude $\mathit{ColsSet}(T) = \mathrm{domain}(f)$. From the assumption it follows that f applied columnwise to T yields a tuple containing exactly one 1, as desired. ■

We emphasize that a similar characterization could be given for the whole set $\mathrm{pPol}(\{R_{1/k}\})$, but using bounded bases makes the characterization much simpler. For instance, Case (1) in the proposition encompasses the case when f is a subfunction of a projection, but also the case when $\mathrm{domain}(f)$ does not contain an exact cover of $\{1, \ldots, \kappa'\}$. In this latter case f can take arbitrary values, since there are no $t_1, \ldots, t_{\kappa'} \in R_{1/\kappa}$ such that $f(t_1, \ldots, t_{\kappa'})$ is defined. It also encompasses (through κ, κ-closure) the general case of functions f (with $|\mathrm{domain}(f)| > \kappa$) whose domain contains several exact covers over which f is a projection, and other tuples which do not participate in any exact cover, and over which the value of f is unconstrained.

C. κ-SAT

Let $\Gamma^{\kappa}_{\mathrm{SAT}}$ be the constraint language that for every $t \in \{0,1\}^{\kappa}$ contains the κ-ary relation $\{0,1\}^{\kappa} \setminus \{t\}$. In other words each relation corresponds to a clause of the form $(\ell_1 \vee \ldots \vee \ell_{\kappa})$ where $\ell_i = x_i$ or $\ell_i = \overline{x_i}$. The following proposition establishes that the bounded base of $\Gamma^{\kappa}_{\mathrm{SAT}}$ is particularly simple, since it only consists of subfunctions of projections.

Proposition 4: Let $\kappa \geq 3$. Then the bounded base of κ-SAT is given by $B(\Gamma^{\kappa}_{\mathrm{SAT}}) = \{f \mid |\mathrm{domain}(f)| \leq \kappa, \mathrm{ar}(f) \leq 2^{\kappa} - 1$ and $\exists i, f$ is a subfunction of $\pi_i^{\mathrm{ar}(f)}\}$.

Proof: Since a partial projection function trivially preserves all relations it is clear that every partial projection f with $|\mathrm{domain}(f)| \leq \kappa$ and $\mathrm{ar}(f) \leq 2^{\kappa} - 1$ is included in $B(\Gamma^{\kappa}_{\mathrm{SAT}})$. For the other direction let $f \in B(\Gamma^{\kappa}_{\mathrm{SAT}})$. By definition of a bounded base we have $|\mathrm{domain}(f)| \leq \mathrm{ar}(\Gamma^{\kappa}_{\mathrm{SAT}}) = \kappa$ and $\mathrm{ar}(f) \leq \#\mathrm{tuples}(\Gamma^{\kappa}_{\mathrm{SAT}}) = 2^{\kappa} - 1$. Now assume towards contradiction that f is not a subfunction of a projection. Write $\kappa' \leq \kappa$ for $|\mathrm{domain}(f)|$ and let $u_1, \ldots, u_{\kappa'}$ be the tuples in $\mathrm{domain}(f)$. Finally, for $i = 1, \ldots, \kappa'$ let ℓ_i be x_i if $f(u_i)$ is 0, and $\overline{x_i}$ if $f(u_i)$ is 1, and let R_f be the κ'-ary relation R_f defined by the clause $(\ell_1 \vee \cdots \vee \ell_{\kappa'})$ Now since f_i is not a subfunction of a projection and $u_1, \ldots, u_{\kappa'}$ enumerate $\mathrm{domain}(f)$, the tuple $(f(u_1), \ldots, f(u_{\kappa'}))$ is different from the tuple $(u_1[i], \ldots, u_{\kappa'}[i])$ for all $i = 1, \ldots, \mathrm{ar}(f)$. Moreover, by construction of R_f, $(f(u_1), \ldots, f(u_{\kappa'}))$ is not in R_f. Since R_f is defined by a clause it contains all tuples but one, and it follows that it contains the tuple $(u_1[i], \ldots, u_{\kappa'}[i])$ for all $i = 1, \ldots, \mathrm{ar}(f)$, but not the tuple $(f(u_1), \ldots, f(u_{\kappa'}))$. Hence R_f is not preserved by f. On the other hand we have $R_f(x_1, \ldots, x_{\kappa'}) = R(x_1, \ldots, x_{\kappa'}, x_{\kappa'}, \ldots, x_{\kappa'})$ for an appropriately chosen $R \in \Gamma^{\kappa}_{\mathrm{SAT}}$, hence $f \in B(\Gamma^{\kappa}_{\mathrm{SAT}}) \subseteq \mathrm{pPol}(\Gamma^{\kappa}_{\mathrm{SAT}})$ should be a polymorphism of R_f, a contradiction. ■

Again, using bounded bases makes the characterization much simpler, since $\mathrm{pPol}(\Gamma^{\kappa}_{\mathrm{SAT}})$ includes functions with $|\mathrm{domain}(f)| = \lambda > \kappa$ which are *not* partial projections, as made clear in the following proposition. For a ground set V, a *hypergraph* $\mathcal{H} \subseteq 2^V$ is a set system over V, and a set $S \subseteq V$ is a *hitting set* of $\mathcal{H}$ if for every $E \in \mathcal{H}$ we have $E \cap S \neq \emptyset$.

Proposition 5: Let f be an n-ary partial function that is not a partial projection, and let $\mathrm{domain}(f) = \{u_1, \ldots, u_k\}$. Let $V = \{v_1, \ldots, v_k\}$, and define $\mathcal{H} = \{E_1, \ldots, E_n\} \subseteq 2^V$ by: $v_i \in E_j$ if and only if $u_i[j] \neq f(u_i)$. Then $f \in \mathrm{pPol}(\Gamma^{\kappa}_{\mathrm{SAT}})$ if and only if every hitting set of $\mathcal{H}$ has size at least $\kappa + 1$.

Proof: Observe $\emptyset \notin \mathcal{H}$, since f is not a partial projection. On the one hand, assume that $\mathcal{H}$ has a hitting set S of size κ or less. Consider the subfunction f' of f with $\mathrm{domain}(f') = \{u_i \mid v_i \in S\}$. By construction, f' is not a partial projection; thus f' has a contraction g defined on at most κ tuples and arity at most $2^{\kappa} - 1$. By Proposition 4, g is not a partial polymorphism of κ-SAT, hence neither is f. On the other hand, assume that all hitting sets of $\mathcal{H}$ have size at least $\kappa + 1$. Let B be the bounded base for κ-SAT; by Theorem 2, we must show $f \in Cl_{\kappa,n}(B)$. Let f' be a restriction of f on at most κ tuples, and let $S = \{v_i \mid u_i \in \mathrm{domain}(f')\}$. Then S is not a hitting set of $\mathcal{H}$, hence for some $E_j \in \mathcal{H}$ we have $S \cap E_j = \emptyset$, hence f' is a subfunction of π^n_j. Now we have a few simple cases. If $n \leq 2^{\kappa} - 1$, then $f' \in B$; if $n > 2^{\kappa} - 1$ but f' has a contraction of arity at most $2^{\kappa} - 1$, then f' is a stretching of a function in B; otherwise $f' \in \mathcal{U}^{2^{\kappa}-1}_n$ by definition of $\mathcal{U}^m_n$. In each case, we see that we can construct a κ-covering of f from $B(\Gamma^{\kappa}_{\mathrm{SAT}})$, hence $f \in Cl_{\kappa,n}(B)$ and we are done. ■

VI. Concluding Remarks

We have proposed the new notion of a *bounded base* which, together with a closure operator, allows one to characterize the partial polymorphisms of a finite constraint language from a finite set of functions. This notion gives simple characterizations of the partial polymorphisms of natural and well-studied languages. Continuing this investigation and determining bounded bases for more constraint languages is a natural starting point, in order to explain the complexity differences between NP-hard CSP problems with the help of partial polymorphisms.

Acknowledgements

Bruno Zanuttini is supported by ANR Project ANR-10-BLAN-0210.

References

[1] V. B. Alekseev and A. A. Voronenko. On some closed classes in partial two-valued logic. *Discrete Mathematics and Applications*, 4(5):401–419, 1994.

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

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

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

[5] P. Jonsson, V. Lagerkvist, G. Nordh, and B. Zanuttini. Complexity of SAT problems, clone theory and the exponential time hypothesis. In *Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA-2013)*, pages 1264–1277, 2013.

[6] V. Lagerkist and M. Wahlström. Polynomially closed co-clones. In *Proceedings of the 44th International Symposium on Multiple-Valued Logic (ISMVL-2014)*, pages 85 – 90, 2014.

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

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

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

[10] 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.

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