An Algebraic Approach to Valued Constraint Satisfaction*

Rostislav Horčík[1], Tommaso Moraschini[2], and Amanda Vidal[3]

1 Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic
horcik@cs.cas.cz

2 Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic
moraschini@cs.cas.cz

3 Institute of Computer Science, Czech Academy of Sciences, Prague, Czech Republic
amanda@cs.cas.cz

Abstract

We study the complexity of the valued CSP (VCSP, for short) over arbitrary templates, taking the general framework of integral bounded linearly order monoids as valuation structures. The class of problems considered here subsumes and generalizes the most common one in VCSP literature, since both monoidal and lattice conjunction operations are allowed in the formulation of constraints. Restricting to locally finite monoids, we introduce a notion of polymorphism that captures the pp-definability in the style of Geiger's result. As a consequence, sufficient conditions for tractability of the classical CSP, related to the existence of certain polymorphisms, are shown to serve also for the valued case. Finally, we establish the dichotomy conjecture for the VCSP, modulo the dichotomy for classical CSP.

1998 ACM Subject Classification F.4.1 Mathematical Logic; G.1.6 Optimization

Keywords and phrases Valued CSP, Polymorphism, pp-definability, Geiger's Theorem

Digital Object Identifier 10.4230/LIPIcs.CSL.2017.42

1 Introduction

The *constraint satisfaction problem* (CSP, for short) is a well-established framework for the uniform study of a wide range of both theoretical and applied problems. For the present purpose, it will be convenient to describe it in purely logical terms. A first-order sentence is *primitive positive* (pp, for short) if it is built up from atomic formulas with equality using only conjunction and existential quantifier. The CSP of a finite relational structure $\mathbf{B}$ asks to determine the pp-sentences valid in $\mathbf{B}$, in symbols

$$\mathrm{CSP}(\mathbf{B}) := \{\varphi \mid \varphi \text{ is pp-sentence and } \mathbf{B} \vDash \varphi\}.$$

It is well known that $\mathrm{CSP}(\mathbf{B})$ can be identified with the set of *finite* structures $\mathbf{A}$ for which there is a homomorphism $\mathbf{A} \to \mathbf{B}$.

* The work was partly supported by the grant No. GA17-04630S of the Czech Science Foundation and partly by the long-term strategic development financing of the Institute of Computer Science (RVO:67985807).

26th EACSL Annual Conference on Computer Science Logic (CSL 2017).
Editors: Valentin Goranko and Mads Dam; Article No. 42; pp. 42:1–42:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

The CSP problem of a finite relational structure is clearly decidable and, more precisely, belongs to the complexity class NP. The most outstanding problem in the field, known as the *dichotomy conjecture*, asks whether it is true that for every finite relational structure $\mathbf{B}$, either CSP($\mathbf{B}$) is NP-complete or it is tractable, i.e., solvable in polynomial time [17]. The algebraic approach to CSP has revealed very fruitful in this direction, leading to the discovery of striking connections with the theory of Maltsev conditions in universal algebra (see e.g. [11, 10, 19, 1, 2, 26]).

▶ **Example 1.** For any natural number n, a graph is *n-colorable* if it is possible to assign colors $\{0, \ldots, n-1\}$ to its vertices in such a way that adjacent vertices receive different colors. We denote by $\mathbf{K}_n$ the complete n-element graph. It is easy to see that the set of finite n-colorable graphs is exactly CSP($\mathbf{K}_n$). Moreover, the problem CSP($\mathbf{K}_n$) is NP-complete for $n \geq 3$, and tractable otherwise. ◀

In this paper we consider a weighted generalization of the classical CSP, known as *valued constraint satisfaction problem* (VCSP, for short) see [25, 13, 29]. Roughly speaking, weighted structures are generalizations of classical structures in which relations are allowed to take degrees of truth (or, equivalently, payoffs) into a suitable valuation structure. Accordingly, the VCSP problem of a finite *weighted* relational structure $\mathbf{B}$ asks to determine an *optimal* solution for any pp-sentence φ, that is a tuple $\vec{b} \in B$ such that for any other tuple $\vec{c} \in B$ the value of φ on $\vec{b}$ is better or equal than its value on $\vec{c}$ (see the next section for a precise definition). As in the classical case, a major challenge in VCSP is to classify finite weighted structures according to whether their VCSP is tractable (solvable in polynomial time) or NP-hard.

Our approach differs from the one found in the VCSP literature in two fundamental aspects. On the one hand, the significant majority of works [29, 12, 27, 23, 21] are confined to the algebraic approach for VCSP formulated over a specific valuation structure, namely the set of positive rationals (with or without infinity) equipped with addition. On the contrary, our logically inspired approach is amenable to provide a uniform treatment for the VCSP formulated over arbitrary valuation structures. In this paper we explore this possibility for the case of locally finite valuation structures.

On the other hand, in the VCSP literature pp-sentences are (implicitly) understood as first-order formulas built up from atomic formulas using only existential quantifier and a kind of *monoidal* non-classical conjunction. As a drawback the VCSP, when formulated in these terms, does not embrace the classical aspects of weighted structures that are related to the usage of the classical conjunction (as opposed to the monoidal one). In this paper we allow the presence of both classical and monoidal conjunctions in pp-formulas. Consequently, our VCSP framework allows to model a richer class of optimization problems.

▶ **Example 2.** Using only the monoidal conjunction, one can model in VCSP the MAX-3SAT problem asking to find, for a given 3-CNF formula, a valuation satisfying the maximum number of clauses. Allowing also the classical conjunction, one can model a robust version of MAX-3SAT, where the input can be considered as not fully specified and we optimize w.r.t. the worst case. More precisely, the input is a finite number of 3-CNF formulas $\varphi_1, \ldots, \varphi_m$ built up from the variables V. The task is to find a valuation $e \in \{0,1\}^V$ which maximizes the minimum number of satisfied clauses in all φ_i's. In Example 5, we will show how this problem can be properly formalized in our setting. ◀

The content of paper can be outlined as follows. In the classical CSP, the algebraic approach to the study of the dichotomy has revealed to be very fruitful. It is based on

Geiger's result [18] (independently by [7, 8]) characterizing the pp-definable relations of a finite structure $\mathbf{B}$ by means of *polymorphisms*, i.e., the homomorphisms $f : \mathbf{B}^n \to \mathbf{B}$ for any n. Our first main contribution is a generalization of Geiger's result to the case of VCSP over locally finite valuation structures. This is achieved by introducing a new notion of polymorphism for weighted structures, which generalizes the classical one. It should be mentioned that the VCSP literature contains already some generalizations of Geiger's result in terms of the so-called *weighted or fractional polymorphisms* [12, 22, 29, 23], but all of them are incomparable to our version at least for two reasons. First, the known generalizations are necessarily confined to pp-formulas without classical conjunction. Second, strictly speaking, they are not able to capture pp-definability in the original language: they do it only in an expanded language, where the expansion preserves tractability of VCSP. On the other hand, our approach provides a generalization of classical Geiger's Theorem (characterizing pp-definable relations in terms of polymorphisms) to the setting of weighted structures which does not require any expansion of the language. For this reason our work is not only a logical contribution to the computational study of VCSP (in which expansions preserving tractability are indistinguishable), but also to the development of weighted model theory (where different languages determine different structures).

Further we show that the VCSP over locally finite valuation structures can be reduced to finitely many CSP's. This reduction to the classical setting has two major consequences. On the one hand, it implies that several *tractability criteria* related to the existence of certain polymorphisms, which encompass the ones obtained in [28], transfer from CSP to VCSP. On the other and, this reduction yields that the dichotomy holds for our VCSP provided the dichotomy for classical CSP holds.[1] Interestingly enough, our results hold in full generality, in the sense that they are not confined to VCSP formulated over a specific valuation structure.

2 Preliminaries

The most general approach towards the study of VCSP complexity has been formulated taking semirings as valuation structures [3]. Here we restrict the attention to the so-called valued constraints, i.e., linearly ordered valuation structures (see for instance [15, 13, 25]).

An *integral Abelian pomonoid* is a structure $\mathbf{L} = \langle L, \leq, \odot, 0, 1\rangle$ such that $\langle L, \leq\rangle$ is a poset with a minimum 0 and maximum 1, $\langle L, \odot, 1\rangle$ is an Abelian monoid, and for every $\alpha, \beta, \gamma \in L$, if $\alpha \leq \beta$, then $\alpha \odot \gamma \leq \beta \odot \gamma$. When $\mathbf{L}$ is linearly ordered it can be equipped naturally with a lattice structure $\langle L, \wedge, \vee\rangle$. From now on we will assume that $\mathbf{L}$ is a fixed but arbitrary linearly-ordered integral Abelian pomonoid, and use the term *valuation structures* to denote these algebras.[2] A valuation structure is said to be *locally finite* if all finite generated subalgebras are finite.

▶ **Example 3.** Let $[0, 1]$ be the real unit interval. We denote by $\cdot$ the real multiplication, and by $\odot$ the operation defined as $a \odot b := \max\{0, a + b - 1\}$ for every $a, b \in [0, 1]$. Then the following algebras are valuation structures

$$\text{Ł} = \langle [0, 1], \wedge, \vee, \odot, 0, 1\rangle, \qquad \mathbf{G} = \langle [0, 1], \wedge, \vee, \wedge, 0, 1\rangle, \qquad \mathbf{P} = \langle [0, 1], \wedge, \vee, \cdot, 0, 1\rangle.$$

[1] Recently, the dichotomy conjecture for the classical CSP has been claimed to be proven in [24] and [9].
[2] The usual definition of a valuation structure is the dual one from the order-theoretic point of view, namely the monoid unit is a bottom element. Thus we interpret elements of valuation structures as pay-offs rather than costs as is usual.

For every natural $n \geq 2$, we denote by $Ł_n$ the subalgebra of Ł with finite universe $\{0, \frac{1}{n-1}, \ldots, \frac{n-1}{n-1}\}$. Moreover, let $\mathbb{Q}^-_\perp$ be the set of negative rational with a least point $-\infty$. Then $\mathbf{Q}^-_\perp = \langle \mathbb{Q}^-_\perp, \wedge, \vee, +, -\infty, 0\rangle$ is a valuation structure as well.

The valuation structures Ł, $Ł_n$ and $\mathbf{G}$ are locally finite, while $\mathbf{P}$ and $\mathbf{Q}^-_\perp$ are not. ◀

We formulate VCSP in purely logical terms, by adapting the definition of the classical CSP to the case where the underlying first-order logic is non-classical instead of classical. We start by defining a suitable syntax. A language $\mathcal{L}$ is a countable set of domain variables together with a finite set of relational symbols (and their arities), binary propositional connectives $\odot, \wedge$ and the existential quantifier $\exists$. We call first-order formulas in this language *primitive positive formulas* or pp-formulas for short.[3] A pp-formula is called *pp-sentence* if it has no free variables, i.e., all its variables are under the scope of a quantifier. We further assume that every language contains a binary relational symbol $\approx$ denoting the classical equality.

Given a language $\mathcal{L}$ and a valuation structure $\mathbf{L}$, an $\mathbf{L}$-*structure or* $\mathbf{L}$-*template* for $\mathcal{L}$ is a tuple $\mathbf{B} = \langle B, \{R^{\mathbf{B}} \mid R \in \mathcal{L}\}\rangle$ such that B is a non-empty set, and for every relational symbol $R \in \mathcal{L}$ there is a map $R^{\mathbf{B}}\colon B^k \to L$ where k is the arity of R, i.e., $R^{\mathbf{B}}$ is a $\mathbf{L}$-valued (or, equivalently, weighted) relation assigning a cost/pay-off to any tuple $\vec{b} \in B^k$. [4] The relational symbol $\approx$ is always interpreted as the classical identity relation, i.e., $x \approx^{\mathbf{B}} y$ equals 1 if $x = y$ and 0 otherwise. When no confusion shall occur, we omit the superscripts from $R^{\mathbf{B}}$ and $\approx^{\mathbf{B}}$.[5] A $\mathbf{L}$-structure $\mathbf{B}$ is *finite* when B is finite. We assume throughout the paper that all $\mathbf{L}$-structures are finite.

A *valuation* in $\mathbf{B}$ is a mapping of the set of variables into B. The value $\|\varphi(\vec{b})\|^{\mathbf{B}}$ of a pp-formula $\varphi(\vec{x})$ under a valuation $\vec{x} \mapsto \vec{b}$ in the structure $\mathbf{B}$, is defined recursively as follows:

$$\begin{aligned}
\|R(\vec{b})\|^{\mathbf{B}} &= R^{\mathbf{B}}(\vec{b}),\\
\|\varphi_1(\vec{b_1}) \odot \varphi_2(\vec{b_2})\|^{\mathbf{B}} &= \|\varphi_1(\vec{b_1})\|^{\mathbf{B}} \odot \|\varphi_2(\vec{b_2})\|^{\mathbf{B}}\\
\|\varphi_1(\vec{b_1}) \wedge \varphi_2(\vec{b_2})\|^{\mathbf{B}} &= \|\varphi_1(\vec{b_1})\|^{\mathbf{B}} \wedge \|\varphi_2(\vec{b_2})\|^{\mathbf{B}}\\
\|\exists x \varphi(\vec{b}, x)\|^{\mathbf{B}} &= \bigvee_{a \in B} \|\varphi(\vec{b}, a)\|^{\mathbf{B}}.
\end{aligned}$$

The above join is always defined since $\mathbf{B}$ is finite. Moreover, since $\mathbf{L}$ is linearly ordered, the value of $\exists x \varphi(\vec{b}, x)$ is always witnessed, i.e., there is $a \in B$ such that $\|\exists x \varphi(\vec{b}, x)\|^{\mathbf{B}} = \|\varphi(\vec{b}, a)\|^{\mathbf{B}}$. Observe that every valuation structure satisfies the following distributive laws:

$$x \odot (y \vee z) = (x \odot y) \vee (x \odot z), \quad x \odot (y \wedge z) = (x \odot y) \wedge (x \odot z), \quad x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z).$$

Due to the above laws, pp-formulas have a *canonical normal form* (NF, for short):

$$\exists \vec{x} \bigwedge_{i \in I} \Big(\bigodot_{j \in J_i} \varphi_j \Big) \text{ where } \varphi_j \text{ is atomic.}$$

Observe that the value of a pp-sentence $\exists \vec{x} \varphi(\vec{x})$ in $\mathbf{B}$ is the same under any evaluation, and that it coincides with the maximum value that can be taken by the formula $\varphi(\vec{x})$ in $\mathbf{B}$. Thus,

---

[3] For the sake of simplicity, we give the same name to the propositional connectives of the language and the operations of the valuation structure.

[4] The notion of $\mathbf{L}$-structure corresponds exactly to the notion of *valued constraint language* used in the VCSP literature. Thus an $\mathbf{L}$-structure together with a pp-sentence corresponds to an instance of VCSP (cf. [13]).

[5] Observe that we assume that the identity relation $\approx$ is one of the basic relations R. For this reason, every time we state a definition imposing some condition on the basic relations R, we are assuming that the same condition holds for $\approx$.

we define a *solution* of a pp-sentence in NF $\exists\vec{x}\varphi(\vec{x})$ as any valuation $\vec{x}\mapsto\vec{b}$ such that

$$\|\varphi(\vec{b})\|^{\mathbf{B}} = \|\exists\vec{x}\varphi(\vec{x})\|^{\mathbf{B}}.$$

▶ **Definition 4.** Let $\mathbf{L}$ be a valuation structure and $\mathbf{B}$ an $\mathbf{L}$-structure. The *valued constraint satisfaction problem* for the template $\mathbf{B}$, in symbols $\mathrm{VCSP}(\mathbf{B})$, is the problem asking to find a solution to a given pp-sentence in NF.

▶ **Example 5.** We are now ready to formulate the robust version of MAX-3SAT problem mentioned in the introduction. Let $0<\alpha<1$ and let $\mathbf{B}$ be a $\mathbf{P}$-structure whose domain is $\{0,1\}$ endowed with relations $R_{\langle i,j,k\rangle}\colon\{0,1\}^3\to[0,1]$ for each $\langle i,j,k\rangle\in\{0,1\}^3$, interpreted by letting $R_{\langle i,j,k\rangle}(x,y,z)=\alpha$ if $\langle x,y,z\rangle=\langle i,j,k\rangle$ and equal to 1 otherwise. Given for instance as input two 3-CNF formulas

$$\varphi_1=(x\vee\neg y\vee z)\wedge(\neg x\vee\neg z\vee u),\qquad \varphi_2=(u\vee y\vee z)\wedge(\neg x\vee z\vee u),$$

we can encode it into the following pp-sentence:

$$\exists x,y,z,u\,(R_{\langle 0,1,0\rangle}(x,y,z)\odot R_{\langle 1,1,0\rangle}(x,z,u))\wedge(R_{\langle 0,0,0\rangle}(u,y,z)\odot R_{\langle 1,0,0\rangle}(x,z,u)).$$

Observe this robust version of MAX-3SAT problem cannot be expressed in the VCSP formulated without classical conjunction. ◀

Given a finite $\mathbf{L}$-template $\mathbf{B}$, we say that $\mathrm{VCSP}(\mathbf{B})$ is *tractable* whenever it can be solved by an algorithm running in polynomial time in the length of input pp-sentence. Intractability and NP-hardness for $\mathrm{VCSP}(\mathbf{B})$ are defined analogously. It should be observed that in this definition, we assume that the computation in the valuation structure is done by an oracle and, therefore, does not affect the length of the computation.[6]

We say that a map (or, equivalently, a weighted relation) $R\colon B^k\to L$ is *pp-definable* in $\mathbf{B}$ when there is some pp-formula $\varphi(\vec{x})$ such that $R(\vec{b})=\|\varphi(\vec{b})\|^{\mathbf{B}}$ for all $\vec{b}\in B^k$. It follows from the definition of VCSP that expanding the language of an $\mathbf{L}$-structure $\mathbf{B}$ with (finitely many) relations pp-definable in it does not change the tractability/intractability of $\mathrm{VCSP}(\mathbf{B})$.

Along the following sections, we will write $\mathbf{B}$ to denote an arbitrary $\mathbf{L}$-structure, for some valuation structure $\mathbf{L}$.

3 Polymorphisms and pp-definability characterization

Recall that the complexity of $\mathrm{CSP}(\mathbf{K})$, for a classical finite structure $\mathbf{K}$, depends only on the pp-definable relations in $\mathbf{K}$. These relations can be characterized in terms of polymorphisms as follows. An n-ary *polymorphism* (see e.g. [11]) of the structure $\mathbf{K}$ is an homomorphism $f\colon\mathbf{K}^n\to\mathbf{K}$ where $\mathbf{K}^n$ denotes the direct product of n many copies of $\mathbf{K}$. We denote by $\mathrm{Pol}(\mathbf{K})$ the clone of all polymorphisms of $\mathbf{K}$. The algebraic approach to the classical CSP is based on the following fundamental result [18]:

▶ **Geiger's Theorem 6.** *A relation R on K is pp-definable in $\mathbf{K}$ if and only if $\mathrm{Pol}(\mathbf{K})\subseteq\mathrm{Pol}(K,R)$, where $\mathrm{Pol}(K,R)$ is the set of polymorphisms of the structure having the domain K and the only relation R.*

[6] Observe that, for the sake of simplicity, we defined $\mathrm{VCSP}(\mathbf{B})$ only for the case where the language of $\mathbf{B}$ is finite. However, the definitions of $\mathrm{VCSP}(\mathbf{B})$, tractability and NP-hardness can be reformulated in a way that embraces also the case where the language of $\mathbf{B}$ is infinite (see for instance [29, Section 1.3]). Our results, when suitably reformulated, remain true even if the assumption of the finiteness of the language is dropped.

We devote this section to a generalization of the above result to the case of VCSP, i.e., we characterize pp-definable relation in a finite $\mathbf{L}$-structure $\mathbf{B}$ as those which are preserved by suitably defined polymorphisms. The presented proof works for all locally finite valuation structures $\mathbf{L}$.

As in the classical case, our notion of a polymorphism shall depend on the concepts of homomorphisms and direct powers of $\mathbf{L}$-structures. Since $\mathbf{L}$-structures are in fact two-sorted structures, their natural homomorphisms are pairs of maps. The first acting on domains and the second on valuation structures. Moreover, these maps have to satisfy some additional properties ensuring that homomorphisms preserves values of pp-formulas.

▶ **Definition 7.** Let $\mathbf{L}, \mathbf{K}$ be valuation structures, $\mathbf{A}$ an $\mathbf{L}$-structure and $\mathbf{B}$ a $\mathbf{K}$-structure for the same language. A tuple $\langle f, \tau\rangle$ of maps $f\colon A \to B$ and $\tau\colon L \to K$ is a *homomorphism* if for all $\alpha, \beta \in L$, every k-ary relational symbols R and every $\vec{a} \in A^k$ we have

$$\tau(\alpha \odot \beta) \leq \tau(\alpha) \odot \tau(\beta), \quad \tau(\alpha \wedge \beta) \leq \tau(\alpha) \wedge \tau(\beta), \quad \tau(R^{\mathbf{A}}(\vec{a})) \leq R^{\mathbf{B}}(f(\vec{a})).$$

In the notation $f(\vec{a})$ the map f is applied component-wise.

It is worth to observe that if $\langle f, \tau\rangle$ is a homomorphism as above, then τ is a lattice homomorphism. This is a consequence of the fact that τ is a monotone map between chains.

As we mentioned, this notion of homomorphism preserves values of pp-formulas (modulo τ), in the sense that for every pp-formula $\varphi(\vec{x})$ and every tuple $\vec{a} \in A$, we have

$$\tau(\|\varphi(\vec{a})\|^{\mathbf{A}}) \leq \|\varphi(f(\vec{a}))\|^{\mathbf{B}}. \tag{1}$$

This fact can be easily proved by induction on the construction of φ.[7] If φ is a pp-sentence $\exists\vec{x}\psi(\vec{x})$, then (1) specializes to the following:

$$\tau(\|\exists\vec{x}\varphi(\vec{x})\|^{\mathbf{A}}) = \tau(\|\varphi(\vec{a})\|^{\mathbf{A}}) \leq \|\varphi(f(\vec{a}))\|^{\mathbf{B}} \leq \|\exists\vec{x}\varphi(\vec{x})\|^{\mathbf{B}},$$

where $\vec{a}$ is any tuple of elements from A witnessing the value $\|\exists\vec{x}\varphi(\vec{x})\|^{\mathbf{A}}$.

Direct powers of $\mathbf{L}$-structures are defined in the same way as direct powers of two-sorted structures. In particular, we form a direct power independently on both sorts, i.e., on domains and valuation structures. More precisely, the k-ary relations on the direct power $\mathbf{B}^n$ of n many copies of a $\mathbf{L}$-structure $\mathbf{B}$ are functions mapping k-tuples of elements from B^n to L^n. It may be useful to visualize the k-tuples of elements of B^n as matrices as follows:

$$\mathbb{A} = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{k1} & \dots & a_{kn} \end{pmatrix}.$$

▶ **Definition 8.** Let $\mathbf{L}$ be valuation structures, $\mathbf{B}$ an $\mathbf{L}$-structure and $n \in \omega$. The direct power of n many copies of $\mathbf{B}$ is the $\mathbf{L}^n$-structure[8] $\mathbf{B}^n$ whose domain is B^n and for a k-ary relational symbol its interpretation $R^{\mathbf{B}^n}\colon B^{k\times n} \to L^n$ is defined for every matrix $\mathbb{A} \in B^{k\times n}$ having columns $\vec{a}_1, \dots, \vec{a}_n$ as follows:

$$R^{\mathbf{B}^n}(\mathbb{A}) \coloneqq \langle R^{\mathbf{B}}(\vec{a}_1), \dots, R^{\mathbf{B}}(\vec{a}_n)\rangle.$$

[7] The only non-trivial step of the induction is the one related to the existential quantifier, which requires to apply the observation that $\tau(x \vee y) = \tau(x) \vee \tau(y)$.

[8] Strictly speaking, $\mathbf{L}^n$ is not a valuation structure because it is not linearly ordered. Nevertheless one can introduce $\mathbf{L}^n$-structures analogously as $\mathbf{L}$-structures.

Given a map $f\colon B^n \to B$, we can extend it to a map from $B^{k\times n}$ to B^k by applying f row-wise. Thus, for a matrix $\mathbb{A} \in B^{k\times n}$ whose rows are $\vec{a}_1, \ldots, \vec{a}_k$ we define $f(\mathbb{A}) \coloneqq \langle f(\vec{a}_1), \ldots, f(\vec{a}_k)\rangle$.

▶ **Definition 9.** Let $\mathbf{B}$ be a $\mathbf{L}$-structure and $n \in \omega$. An n-ary *polymorphism* of $\mathbf{B}$ is a homomorphism from $\mathbf{B}^n$ to $\mathbf{B}$. The set of all polymorphisms of $\mathbf{B}$ is denoted $\mathrm{Pol}(\mathbf{B})$.

In other words, an n-ary polymorphism of $\mathbf{B}$ is a pair of maps $\langle f, \tau\rangle$ with $f\colon B^n \to B$ and $\tau\colon L^n \to L$ satisfying the following conditions:

1. For all $\vec{\alpha}, \vec{\beta} \in L^n$, it holds that $\tau(\vec{\alpha} \wedge \vec{\beta}) \leq \tau(\vec{\alpha}) \wedge \tau(\vec{\beta})$ and $\tau(\vec{\alpha} \odot \vec{\beta}) \leq \tau(\vec{\alpha}) \odot \tau(\vec{\beta})$.
2. For each k-ary relation $R^{\mathbf{B}}$ in $\mathbf{B}$ and matrix $\mathbb{A} \in B^{k\times n}$ we have $\tau R^{\mathbf{B}^n}(\mathbb{A}) \leq R^{\mathbf{B}}(f(\mathbb{A}))$.

The main result of this section is the following generalization of Geiger's Theorem:

▶ **Theorem 10.** *Let $\mathbf{B}$ be a finite $\mathbf{L}$-structure over a locally finite valuation structure $\mathbf{L}$. A relation $R\colon B^k \to L$ is pp-definable in $\mathbf{B}$ if, and only if, $\mathrm{Pol}(\mathbf{B}) \subseteq \mathrm{Pol}(B, R)$, where $\mathrm{Pol}(B, R)$ is the set of polymorphisms of the $\mathbf{L}$-structure having the domain B and the only relation R.*

▶ **Remark 11.** *It is worth noticing that valuation structures are subreducts of so-called MTL-algebras, the algebraic counterpart of the monoidal t-norm based logic MTL [16]. Consequently the above theorem in fact characterizes pp-definable relations in structures over locally finite MTL-chains.* ◀

Instead of presenting the technical details of the proof of Theorem 10, which are contained in the Appendix, let us describe the main ideas behind it.

First, we strongly rely on a correspondence between the components of our syntax, i.e., $\approx, \exists, \wedge, \odot$, and closure properties of the set of pp-definable relations in $\mathbf{B}$. More precisely, *permutations* of arguments and their *identifications* in a relation correspond to the presence of equality in the language. Meanwhile, the presence of existential quantifier is reflected in the fact that the pp-definable relations of $\mathbf{B}$ are closed under the following generalization of *projection*: for every relation $R\colon B^k \to L$, and $i \leq k$, the i-th projection of R is the $(k-1)$-ary relation $\pi_i[R]$, defined for every $\langle b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_k\rangle \in B^{k-1}$ as follows:

$$\pi_i[R](b_1, \ldots, b_{i-1}, b_{i+1}, \ldots, b_k) \coloneqq \bigvee\{R(\vec{c}) \mid \vec{c} \in B^k,\ b_j = c_j \text{ for every } j \neq i\}.$$

Finally, the presence of $\odot$ and $\wedge$ in the syntax corresponds the fact that the pp-definable relations of $\mathbf{B}$ are closed under the formation of a certain generalization of *intersections* and *Cartesian products*.

Second, we rely on the fact that the pp-definable relations of a given arity k form a closure system on the lattice $\mathbf{L}^{B^k}$. Let us explain briefly why this is the case. First observe that, since $\mathbf{L}$ is locally finite and the language $\mathbf{B}$ is finite, there are only finitely many pp-definable k-ary relations in $\mathbf{B}$. Then consider a relation $R\colon B^k \to L$. There are only finitely many pp-definable k-ary relations $R_1, \ldots, R_m$ such that $R \leq R_i$. It is easy to see that $S = R_1 \wedge \cdots \wedge R_m$ is the least pp-definable relation which extends R (in the sense $R(\vec{b}) \leq S(\vec{b})$ for every $\vec{b} \in B^k$). Observe that its existence strongly depends on the fact that we allow the presence of *classical conjunction* in our pp-formulas. We denote by $\gamma^{(k)}(\cdot)$ the closure operator that assigns to any relation $R\colon B^k \to L$ the least pp-definable relation $\gamma^{(k)}(R)$ extending R. When no confusion shall occur, we will omit the superscript in $\gamma^{(k)}$.

Building on the two observations above, we are able to prove the following fundamental result, whose proof is contained in the Appendix. For a given $k \geq 1$ we define a matrix $\mathbb{C}_k \in B^{k\times |B|^k}$ whose *columns* are all k-tuples from B^k listed in a fixed linear order.

▶ **Extension Lemma 12.** *Let $\mathbf{B}$ be a finite $\mathbf{L}$-structure, $\vec{b} \in B^k$, $n = |B|^k$ and $R\colon B^k \to L$ a k-ary relation. Then there is an n-ary polymorphism $\langle f, \tau \rangle \in \mathrm{Pol}(\mathbf{B})$ such that $f(\mathbb{C}_k) = \vec{b}$ and $\tau R^{\mathbf{B}^n}(\mathbb{C}_k) = \gamma(R)(\vec{b})$.*

We conclude by sketching the proof of the main result of the section.

Proof of Theorem 10. The left-to-right implications follows from the natural observation that pp-definable relations in $\mathbf{B}$ are preserved by all polymorphisms of $\mathbf{B}$. This can be easily checked by induction on the complexity of formulas.

The converse implication follows from the extension Lemma 12. Take $\vec{b} \in B^k$. We have to show that R is pp-definable, i.e., $\gamma(R)(\vec{b}) = R(\vec{b})$. Let $n = |B|^k$. By Lemma 12 there is an n-ary polymorphism $\langle f, \tau \rangle \in \mathrm{Pol}(\mathbf{B})$ such that $f(\mathbb{C}_k) = \vec{b}$ and $\tau R(\mathbb{C}_k) = \gamma(R)(\vec{b})$. Since we assume that R is preserved by all polymorphisms, we obtain $\gamma(R)(\vec{b}) = \tau R(\mathbb{C}_k) \leq Rf(\mathbb{C}_k) = R(\vec{b})$. Thus $R = \gamma(R)$, i.e., R is pp-definable. ◀

4 Reducing VCSP to classical CSP

In this section we show that if $\mathbf{B}$ is a finite $\mathbf{L}$-structure and $\mathbf{L}$ is locally finite, then $\mathrm{VCSP}(\mathbf{B})$ can be reduced to a finite number of classical CSPs. In particular, this yields the Dichotomy Conjecture holds for these VCSPs modulo the Dichotomy Conjecture for classical CSP.

Along this section, we work with a fixed but arbitrary finite $\mathbf{L}$-structure $\mathbf{B} = \langle B, R_1, \dots, R_m \rangle$ where $\mathbf{L}$ is locally finite. Consider the ranges of relations $R_1, \dots, R_m$. Their union is a finite set $X \subseteq L$, generating a finite subalgebra $\mathbf{L}'$ of $\mathbf{L}$. It is obvious that pp-definable relations of $\mathbf{B}$ can attain values only in L'. Consequently, we can assume w.l.o.g. that $\mathbf{L} = \mathbf{L}'$ and, therefore, that $\mathbf{L}$ itself is finite. Accordingly, let $|L| = n$ and define $r := n^n$. We say that r is the *rank* of $\mathbf{L}$.

▶ **Lemma 13.** *Let $\vec{\alpha} = \langle \alpha_i \in L \mid i \in I \rangle$ be a sequence of elements indexed by a finite set I such that $|I| \geq r$. Then there is $J \subseteq I$ such that $|J| = r$ and $\bigodot_{i \in I} \alpha_i = \bigodot_{i \in J} \alpha_i$.*

Proof. For any $\alpha \in L$ we have $\alpha^{n+1} = \alpha^n$ because $|L| = n$ and $\alpha \geq \cdots \geq \alpha^n \geq \alpha^{n+1}$. Since $|I| \geq n^n$, the sequence $\vec{\alpha}$ contains elements $\alpha \in L$ occurring more than n times. Using $\alpha^{n+1} = \alpha^n$, we can omit a suitable number of them and still preserve the value of $\bigodot_{i \in I} \alpha_i$.

Equivalently, let K be the variety of Abelian monoids axiomatized by the equation $x^{n+1} = x^n$. Since $\mathbf{L}$ has n elements, its monoidal reduct belongs to K. Observe that the n-generated free Abelian monoid over K has exactly n^n elements. Thus every product of elements in $\mathbf{L}$ is equivalent to a product of n^n many of them. ◀

We will make use of two basic transformations on $\mathbf{B}$. A first useful transformation of $\mathbf{B}$ consists in an expansion of the language by means of a particular family of pp-definable relations. The *cardinality* of a multiset I, denoted by $|I|$, is the number of elements of I counting repetitions. We consider the expansion of the language of $\mathbf{B}$ with new relational symbols R_I for each multiset of elements in $\{1, \dots, m\}$ of cardinality $\leq r$. The arity of R_I is the sum of all arities of R_i, $i \in I$, i.e., $ar(R_I) := \sum_{i \in I} ar(R_i)$.

We denote by $\widehat{\mathbf{B}}$ the *expansion* of $\mathbf{B}$ obtained by adding to $\mathbf{B}$ the previous collection of relations R_I's interpreted as $R_I^{\widehat{\mathbf{B}}}(\vec{x}_i \mid i \in I) := \bigodot\limits_{i \in I} R_i^{\mathbf{B}}(\vec{x}_i)$ for $I = \langle t_1, \dots, t_q \rangle$.

Observe that the language of $\widehat{\mathbf{B}}$ is finite and, therefore $\widehat{\mathbf{B}}$ is a $\mathbf{L}$-structure as well. Moreover, since the expansion $\widehat{\mathbf{B}}$ is obtained by adding only pp-definable relations, the complexity of $\mathrm{VCSP}(\mathbf{B})$ coincides with the one of $\mathrm{VCSP}(\widehat{\mathbf{B}})$.

From a pp-formula φ of $\mathbf{B}$ in NF, we will define recursively a new pp-formula $\widehat{\varphi}$ in the language of $\widehat{\mathbf{B}}$ as follows. First consider the case where $\varphi = R_{t_1}(\vec{x}_{t_1}) \odot \cdots \odot R_{t_q}(\vec{x}_{t_q})$. Let I be the multiset whose elements are $t_1, \dots, t_q$ counting repetitions. We define

$$\widehat{\varphi} := \begin{cases} R_I(\vec{x}_{t_1}, \dots, \vec{x}_{t_q}) & \text{if } |I| \leq r, \\ \bigwedge\{R_J(\vec{x}_j \mid j \in J) \mid J \text{ is a submultiset of } I \text{ s.t. } |J| = r\} & \text{otherwise.} \end{cases}$$

Moreover, we set $\widehat{\varphi_1 \wedge \varphi_2} := \widehat{\varphi}_1 \wedge \widehat{\varphi}_2$ and $\widehat{\exists x\, \varphi} := \exists x\, \widehat{\varphi}$.

Observe that the symbol $\odot$ does not appear in $\widehat{\varphi}$. Moreover, it is easy to see that, since the valuation structure $\mathbf{L}$ is fixed, $\widehat{\varphi}$ can be constructed out of φ in polynomial time in the length of φ.[9] On the other hand, while obtaining $\widehat{\varphi}$ for an arbitrary valuation structure $\mathbf{L}$ has a quite high computational cost, lower bounds and easier constructions are likely to exist when each particular structure is studied separately.

▶ **Lemma 14.** *For every pp-formula φ of $\mathbf{B}$ in NF and $\vec{b} \in B$ it holds $\|\varphi(\vec{b})\|^{\mathbf{B}} = \|\widehat{\varphi}(\vec{b})\|^{\widehat{\mathbf{B}}}$.*

Proof. According to the definition of the translation $\widehat{\ }$, it suffices to check the equality for pp-formulas of the form $\varphi = \bigodot_{i \in I} R_i(\vec{x}_i)$. Since the monoidal unit $1 \in L$ is the top element, we have that for every $\alpha, \beta \in L$ it holds that $\alpha \odot \beta \leq \alpha \wedge \beta$. This fact easily implies that $\|\varphi(\vec{b})\|^{\mathbf{B}} \leq \|\widehat{\varphi}(\vec{b})\|^{\widehat{\mathbf{B}}}$. The other inequality is a direct consequence of Lemma 13. ◀

A second family of transformations of a structure $\mathbf{B}$ is given by *classicalizations* of it. Given $\alpha \in L$, we define a classical (non-weighted) structure $\mathbf{B}_\alpha$ whose language coincides with the one of $\mathbf{B}$, and every k-ary basic relation R is interpreted as follows: for every $\vec{b} \in B^k$ we set $\mathbf{B}_\alpha \vDash R(\vec{b}) \iff R^{\mathbf{B}}(\vec{b}) \geq \alpha$. We can apply these classicalizations to the structure $\widehat{\mathbf{B}}$ defined before, obtaining the following useful result:

▶ **Lemma 15.** *Let $\alpha \in L$, φ a pp-formula of $\mathbf{B}$ in NF and $\vec{b} \in B$. The following are equivalent:*
1. $\widehat{\mathbf{B}}_\alpha \vDash \widehat{\varphi}(\vec{b})$.
2. $\|\widehat{\varphi}(\vec{b})\|^{\widehat{\mathbf{B}}} \geq \alpha$.
3. $\|\varphi(\vec{b})\|^{\mathbf{B}} \geq \alpha$.

Proof. The equivalence between 2. and 3. follows from Lemma 14. The equivalence between 1. and 2. can be proven easily by induction on the construction of formulas (recall that $\odot$ does not appear in pp-formulas of the form $\widehat{\varphi}$). ◀

The next result shows that VCSP($\mathbf{B}$) can be reduced to a finite number of classical CSPs.

▶ **Theorem 16.**
- VCSP($\mathbf{B}$) *is tractable iff* CSP($\widehat{\mathbf{B}}_\alpha$) *is tractable for every* $\alpha \in L$.
- VCSP($\mathbf{B}$) *is NP-hard iff* CSP($\widehat{\mathbf{B}}_\alpha$) *is NP-hard for some* $\alpha \in L$.

Proof. Given a pp-sentence φ of $\widehat{\mathbf{B}}$ in which $\odot$ does not occur, we denote by φ^* the formula obtained from φ replacing the relations from the expansion $\widehat{\mathbf{B}}$ with the corresponding formulas

[9] This can be proved by induction on the construction of φ. The only non-trivial case is the one where $\varphi = R_{t_1}(\vec{x}_{t_1}) \odot \cdots \odot R_{t_q}(\vec{x}_{t_1})$. Let I be the multiset whose elements are $t_1, \dots, t_q$ counting repetitions. It is easy to see that the length of $\widehat{\varphi}$ is bounded above by the binomial coefficient $\binom{|I|}{r}$. Since $\binom{|I|}{r}$ is bounded above by $|\varphi|^r$, where $|\varphi|$ is the length of φ, we are done.

of $\mathbf{B}$.[10] Observe that both the length of φ^* and the time needed in its construction are bounded above polynomially in the length of φ.

From Lemma 15 it follows that for every $\alpha \in L$ it holds that $\widehat{\mathbf{B}}_\alpha \vDash \varphi \iff \|\varphi^*\|^{\mathbf{B}} \geq \alpha$. Together with the fact that φ^* can be constructed in polynomial time in the length of φ, the above display easily implies that:

- If $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ is NP-hard for some $\alpha \in L$, then also $\mathrm{VCSP}(\mathbf{B})$ is NP-hard as well.
- If $\mathrm{VCSP}(\mathbf{B})$ is tractable, then so is $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ for every $\alpha \in L$.

Conversely, consider any pp-sentence φ of $\mathbf{B}$. Then define $\beta := \max\{\alpha \in L \mid \widehat{\mathbf{B}}_\alpha \vDash \widehat{\varphi}\}$. From Lemma 15 it follows that the following conditions are equivalent for every $\vec{b} \in B$:

1. $\vec{b}$ is a solution for φ in $\mathbf{B}$.
2. $\vec{b}$ witnesses the fact that $\widehat{\mathbf{B}}_\beta \vDash \widehat{\varphi}$.

Suppose $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ is tractable for every $\alpha \in L$ and consider a pp-sentence φ of $\mathbf{B}$. Recall that $\widehat{\varphi}$ can be constructed in polynomial time in the length of φ. Moreover, β can be calculated in polynomial time in the length of $\widehat{\varphi}$ by the assumption on the tractability of each of the $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ and that fact that $\mathbf{L}$ is finite. It is well known that if $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ is tractable, then given a pp-sentence ψ, the problem of then determining whether $\widehat{\mathbf{B}}_\alpha \vDash \psi$, and in the positive case produce a tuple $\vec{b}$ witnessing $\widehat{\mathbf{B}}_\alpha \vDash \psi$, is a tractable problem too [14]. Building on this, we obtain a tuple $\vec{b}$ from B witnessing the fact that $\widehat{\mathbf{B}}_\beta \vDash \widehat{\varphi}$ in polynomial time in the length of $\widehat{\varphi}$. Hence we conclude that this tuple $\vec{b}$ can be constructed in polynomial time in the length of φ. Since **2**. implies **1**. we conclude that $\vec{b}$ is a solution for φ in $\mathbf{B}$. Hence $\mathrm{VCSP}(\mathbf{B})$ is tractable as desired. Similarly, if $\mathrm{VCSP}(\mathbf{B})$ is NP-hard, so is at least one $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$. ◀

▶ **Corollary 17.** *The dichotomy conjecture holds for the VCSP formulated over finite $\mathbf{L}$-structures with $\mathbf{L}$ locally finite if and only if it holds for the classical CSP over finite templates.*

▶ **Remark 18.** *Observe that Theorem 16 is formulated under the assumption that $\mathbf{L}$ is finite, but it holds equivalently under the assumption that $\mathbf{L}$ locally finite. For this reason, the result applies to the case where $\mathbf{L}$ is Ł, Ł$_n$ or $\mathbf{G}$ (see Example 3). In this setting the VCSP is also known under the names of weighted CSP and fuzzy CSP (cf. [25]).* ◀

As we mentioned, our formulation of the VCSP allows the presence of classical conjunction in pp-formulas. It is natural to ask whether this expansion of the language (w.r.t. pp-formulas without $\wedge$) preserves the tractability of the VCSP. In the next example we show that this is not the case in general.

▶ **Example 19.** Let $\mathbf{K}$ be a classical structure for which $\mathrm{CSP}(\mathbf{K})$ is NP-hard. We define an Ł$_3$-structure $\mathbf{B}$ in the same language of $\mathbf{K}$ as follows. For every basic k-ary relation R, we set $R^{\mathbf{B}}(\vec{x}) = 1/2$ if $\vec{x} \in R^{\mathbf{K}}$ and equal to 0 otherwise.

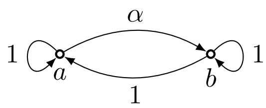
It is easy to see that the basic relations of $\widehat{\mathbf{B}}_{1/2}$ include the ones of $\mathbf{K}$. Hence $\mathrm{CSP}(\widehat{\mathbf{B}}_{1/2})$ is NP-hard. By Theorem 16 we conclude that $\mathrm{VCSP}(\mathbf{B})$ is NP-hard as well. On the other hand, when formulated without classical conjunction, the VCSP of $\mathbf{B}$ is tractable. To see this, let Γ be the set of pp-sentences without $\wedge$. It is easy to see that if $\varphi \in \Gamma$ contains at least two occurrences of a basic relation R different from the identity, then the $\|\varphi\|^{\mathbf{B}} = 0$ and, therefore, every tuple of elements of B is a solution of φ. Therefore the only meaningful pp-sentences have the following form (for some, possibly none, basic relation R): $\exists \vec{x}\, R(\vec{x}) \odot \bigodot_{i \in I}(y_i \approx z_i)$. It is not difficult to see that they can be handled in polynomial time. ◀

[10] If the classical conjunction $\wedge$ is not allowed in the language, this process cannot be done in general. See e.g. Example 19 below.

Even if the addition of classical conjunction to pp-sentences does not preserves tractability, the clone of polymorphisms $\mathrm{Pol}(\mathbf{B})$, which preserves $\wedge$, of an $\mathbf{L}$-structure $\mathbf{B}$ may contain useful information for handling pp-sentences in which $\wedge$ does not occur, as we remark in the next example.

▶ **Example 20.** Recall the $\langle s,t\rangle$-MIN-CUT problem. Given a weighted digraph $\mathbf{H} = \langle H, \mu, s, t\rangle$, where $\mu\colon H^2 \to \mathbb{N} \cup \{\infty\}$ is a map assigning weights to tuples of vertices ($\mu(u,v) = \infty$ if there is no arc between $u, v \in H$) and $s, t \in H$, this problem asks to find an $\langle s,t\rangle$-cut with a minimum weight. A $\langle s,t\rangle$-cut of $\mathbf{H}$ is a subset $C \subseteq H$ such that $s \in C$ and $t \notin C$. The weight of C is defined by $\sum_{u\in C, v\notin C} \mu(u,v)$.

The $\langle s,t\rangle$-MIN-CUT can be formulated within $\mathrm{VCSP}(\mathbf{B})$ over the $\mathbf{P}$-structure $\mathbf{B} = \langle \{a,b\}, R, P_s, P_t\rangle$ where $P_s(a) = 1$ and $P_s(b) = 0$, $P_t(b) = 1$ and $P_t(a) = 0$, and $R\colon \{a,b\}^2 \to [0,1]$ is defined in the following figure for $0 < \alpha < 1$.



The input weighted digraph can be encoded into a pp-sentence whose variables are vertices of $\mathbf{H}$ as follows:

$$\exists \vec{x}\, P_s(s) \odot P_t(t) \odot \bigodot \{R(x_i,x_j)^{\mu(x_i,x_j)} \colon x_i, x_j \in H \text{ and } \mu(x_i,x_j) \neq 0\}$$

where $R(x_i,x_j)^{\mu(x_i,x_j)} = R(x_i,x_j) \odot \ldots \odot R(x_i,x_j)$ repeated $\mu(x_i,x_j)$ many times.

It can be shown that $\mathrm{VCSP}(\mathbf{B})$ is intractable. This holds even if we replace the valuation structure $\mathbf{P}$ by a locally finite one provided that $\alpha^2 > 0$.[11] Nevertheless, the $\wedge$-less fragment is tractable since it admits a symmetric fractional polymorphism $\langle \min, \max\rangle$ of every arity (see [22]). Even though the full language version is intractable, the clone of polymorphisms $\mathrm{Pol}(\mathbf{B})$ still contains useful information. For instance $\langle \min, \tau\rangle$ and $\langle \max, \tau\rangle$ for $\tau(\alpha,\beta) = \alpha \odot \beta$ are polymorphisms of $\mathbf{B}$ where min and max are computed w.r.t. the linear order $a < b$. The fact that $\langle \min, \max\rangle$ is a fractional polymorphism can be captured by $\tau(R(\vec{x}), R(\vec{y})) \leq \tau(R(\min(\vec{x},\vec{y})), R(\max(\vec{x},\vec{y})))$ for all relations R which are pp-definable in the language $\exists, \odot$. Thus one can investigate fractional polymorphisms inside the clone $\mathrm{Pol}(\mathbf{B})$. ◀

5 Tractability conditions

In this section we will focus on tractability conditions for $\mathrm{VCSP}(\mathbf{B})$ consisting in existence of suitable polymorphisms of a finite $\mathbf{L}$-structure $\mathbf{B}$. Remarkably, given a polymorphism $\langle f, \tau\rangle \in \mathrm{Pol}(\mathbf{B})$, the component f is a classical polymorphism of the classical structure $\widehat{\mathbf{B}}_\alpha$ provided that $\alpha \leq \tau(\alpha,\ldots,\alpha)$. This will establish a connection between tractability conditions of the classical $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$ and tractability conditions for $\mathrm{VCSP}(\mathbf{B})$.

▶ **Definition 21.** Let $\alpha \in L$. A polymorphism $\langle f, \tau\rangle$ of $\mathbf{B}$ is *increasing in* α whenever $\alpha \leq \tau(\alpha,\ldots,\alpha)$. We will denote the set of polymorphisms of $\mathbf{B}$ increasing in α by $\mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})$.

We say that a polymorphism is *locally increasing* if it is increasing in some value $\alpha \in L$. Such polymorphisms play a central role in the transfer of tractability conditions between $\mathrm{VCSP}(\mathbf{B})$ and the different $\mathrm{CSP}(\widehat{\mathbf{B}}_\alpha)$. More precisely, we have the following:

[11] This can be proved by showing non-existence of a ternary cyclic polymorphism in $\widehat{\mathbf{B}}_{\alpha^2}$ and then applying Theorem 16.

▶ **Lemma 22.** *We have* $\{f \mid \langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})\} \subseteq \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$ *for all* $\alpha \in L$.

Proof. Let $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})$ be n-ary. Consider any k-ary relation R of $\widehat{\mathbf{B}}_\alpha$, and $\mathbb{B} \in B^{k\times n}$ such that $\mathbb{B} \in R$ (i.e., the columns of $\mathbb{B}$ belong to R). Observe that Lemma 15 implies that, for any $\vec{b} \in R$ it holds that $\alpha \leq \|R^*(\vec{b})\|^{\mathbf{B}}$, where R^* is defined as in the proof of Lemma 16 (so that $\widehat{R^*} = R$). Then, by monotonicity of τ and the fact that $\langle f, \tau\rangle$ is increasing in α it follows that $\alpha \leq \tau(\alpha, \ldots, \alpha) \leq \tau(\|R^*(\mathbb{B})\|^{\mathbf{B}})$, where $\|R^*(\mathbb{B})\|^{\mathbf{B}} = \langle \|R^*(\vec{b}_1)\|^{\mathbf{B}}, \ldots, \|R^*(\vec{b}_m)\|^{\mathbf{B}}\rangle$ for the columns $\vec{b}_1, \ldots, \vec{b}_m$ of $\mathbb{B}$. Since $\langle f, \tau\rangle$ is a polymorphism of $\mathbf{B}$ it preserves pp-definable relations, and in particular R^*. Thus, $\tau(\|R^*(\mathbb{B})\|^{\mathbf{B}}) \leq \|R^*(f(\mathbb{B}))\|^{\mathbf{B}}$. Lemma 15 implies that $f(\mathbb{B}) \in R$ concluding the proof. ◀

In order to provide a converse inclusion to the one from Lemma 22, we will construct for every classical polymorphism $f \in \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$ a polymorphism $\langle f, \tau_f\rangle \in \mathrm{Pol}(\mathbf{B})$. To this end, recall that $\mathbb{C}_n$ is the matrix whose columns are all n-tuples from B^n, so its transposed $\mathbb{C}_n^T$ is the matrix whose rows are all the elements of B^n. Then for any $\vec{\alpha} \in L^n$ we let $R_{\vec{\alpha}} \colon B^{|B|^n} \to L$ be given by $R_{\vec{\alpha}}(\mathbb{C}_n^T) \coloneqq \vec{\alpha}$ and by $R_{\vec{\alpha}}(\vec{b}) \coloneqq 0$ for $\vec{b} \in B^{|B|^n} \setminus \mathrm{Columns}(\mathbb{C}_n^T)$.

Given a mapping $f\colon B^n \to B$, define $\tau_f \colon L^n \to L$ by $\tau_f(\vec{\alpha}) \coloneqq (\gamma(R_{\vec{\alpha}}))(f(\mathbb{C}_n^T))$ where γ is the closure operator defined in Section 3.

▶ **Lemma 23.** *We have* $\langle f, \tau_f\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})$ *iff* $f \in \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$ *for any* $\alpha \in L$.

Proof. The left-to-right implication follows from Lemma 22. The converse implication follows from the Extension Lemma 12. The details are included in the Appendix. ◀

Sufficient tractability conditions for VCSP($\mathbf{B}$) arise now naturally from the ones for the associated CSP($\widehat{\mathbf{B}}_\alpha$).

▶ **Corollary 24.** *If for each* $\alpha \in L$ *there is* $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})$ *such that* f *satisfies a Maltsev condition implying tractability for classical CSP then* VCSP($\mathbf{B}$) *is tractable.*

Proof. Follows easily as a combination of Theorem 16 and Lemma 23. ◀

Of particular interest is the class of polymorphisms that are increasing in each value $\alpha \in L$. Precisely, we say that an n-ary polymorphism $\langle f, \tau\rangle$ of $\mathbf{B}$ is *fully increasing* (and write $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$) whenever for each $\vec{\alpha} \in L^n$ it holds $\bigwedge\limits_{1=i}^{n} \alpha_i \leq \tau(\vec{\alpha})$.

It is easy to see that for an n-ary polymorphism $\langle f, \tau\rangle$ over an $\mathbf{L}$-structure it holds that

$$\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B}) \text{ for all } \alpha \in L \qquad \text{if and only if} \qquad \langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$$

▶ **Lemma 25.** *The following conditions are equivalent:*

1. $f \in \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$ *for all* $\alpha \in L$,
2. *There is some* τ *such that* $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow\alpha}(\mathbf{B})$ *for all* $\alpha \in L$,
3. *There is some* τ *such that* $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$.

Proof. Clearly 3.⇒2., and 2.⇒3. follows easily by contraposition exploiting the monotonicity of τ. 2.⇒1. is a particular application of Lemma 22. Finally, 1.⇒3. follows by considering the mapping τ_f from Lemma 23, which is now increasing for each $\alpha \in L$. ◀

It is immediate that we can reformulate Corollary 24 in the particular case of fully increasing polymorphisms.

▶ **Corollary 26.** *If there is* $\langle f, \tau\rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$ *such that* f *satisfies a Maltsev condition implying tractability for classical CSP then* VCSP($\mathbf{B}$) *is tractable.*

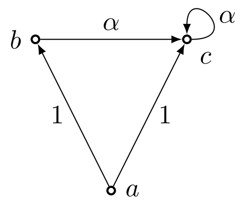


Figure 1 The relation R; the arcs with weight 0 are omitted.

The following example contains an application of fully increasing polymorphisms.

▶ **Example 27.** Let $\mathbf{L}$ be a valuation structure and $\alpha \in L$ such that $\alpha < 1$. Consider an $\mathbf{L}$-structure $\mathbf{B} = \langle \{a, b, c\}, R \rangle$ such that $R \colon B^2 \to L$ is a binary relation depicted in Figure 1. The $\mathbf{L}$-structure $\mathbf{B}$ has a fully-increasing polymorphism $\langle f, \tau \rangle$. If we introduce an order on B by setting $a < b < c$, then $f \colon B^2 \to B$ is given by $f(x, y) = \min\{x, y\}$ and $\tau(x, y) = x \vee y$ if $x, y > 0$ and equal to 0 otherwise. It is easy to check that $\langle f, \tau \rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$. Consequently, if $\mathbf{L}$ is locally finite, then VCSP($\mathbf{B}$) is tractable (Corollary 26) because f is a semilattice operation.

The problem VCSP($\mathbf{B}$) can have different interpretations depending also on the valuation structure $\mathbf{L}$. We describe one of them for $\mathbf{L} = \text{Ł}_{n+2}$ for $n \geq 1$. The input for the problem is a finite number of digraphs $\mathbf{C}_1, \ldots, \mathbf{C}_m$. The task is to partition the union of vertices $\bigcup_{i=1}^{m} C_i$ into three parts denoted respectively a, b, c such that the following conditions are satisfied for all $i \in \{1, \ldots, m\}$:

1. $\mathbf{C}_i$ can have arcs coming from the part a to the parts b or c,
2. $\mathbf{C}_i$ can have at most n many arcs coming from the parts b or c to the part c,
3. no other arcs are allowed in $\mathbf{C}_i$.

If the above condition can be satisfied by some partitions, we are looking among them for a partition which minimizes the numbers of arcs coming from the parts b or c to the part c in the digraph $\mathbf{C}_i$ having the maximum number of such arcs. ◀

In the following example we show that it might be the case that for every $\alpha \in L$ there is a polymorphism f_α of the classical structure $\widehat{\mathbf{B}}_\alpha$ satisfying a fixed Maltsev condition C (e.g. majority), but there is no fully increasing polymorphism $\langle f, \tau \rangle \in \mathrm{Pol}_{\uparrow}(\mathbf{B})$ such that f satisfies C.

▶ **Example 28.** Let $\mathbf{A}$ be the Ł_3-structure with universe $\{a, b, c\}$ and four binary relations:

P	a	b	c
a	1/2	0	1/2
b	0	0	0
c	0	1/2	0

Q	a	b	c
a	1/2	1/2	0
b	0	0	1/2
c	0	0	0

M	a	b	c
a	1/2	1/2	1/2
b	1	1	1/2
c	1/2	1/2	1

N	a	b	c
a	1	1/2	1/2
b	1/2	1	1
c	1/2	1/2	1/2

M, N are the only non empty relations in $\widehat{\mathbf{A}}_1$ and P, Q are the only non total ones in $\widehat{\mathbf{A}}_{1/2}$.

It is easy to check that a ternary majority function that maps $\langle x, y, z \rangle \mapsto b$ whenever x, y, z are different is a majority polymorphism in $\widehat{\mathbf{A}}_1$. Similarly, a majority function that maps $\langle x, y, z \rangle \mapsto a$ whenever x, y, z are different is a majority polymorphism in $\widehat{\mathbf{A}}_{1/2}$.

On the other hand, by simple calculations we can prove that any majority polymorphism f of $\widehat{\mathbf{A}}_1$ necessarily satisfies $f(a, b, c) = b$, and similarly, any majority polymorphism f' of $\widehat{\mathbf{A}}_{1/2}$ necessarily satisfies $f'(a, b, c) = a$. These two incompatible conditions, together with Lemma 25 make it impossible for $\mathbf{A}$ to have an increasing majority polymorphism. ◀

6 Conclusions

In this work we made first steps in the logical and algebraic study of VCSP over an arbitrary valuation structure, obtaining new connections with the classical CSP. This *logical* approach opens the door to several interesting problems. Among them we count the following:

- As model theory turned out to be a fundamental tool in the study of CSP over *infinite* templates [6, 4], we believe that model theory of first-order non-classical logic could shed some light on the analogous generalization of VCSP.
- Drawing our inspiration from [20, 5], we hope to generalize Datalog programs to our setting. In addition, this could help to prove that it is decidable whether Datalog programs can solve VCSP($\mathbf{B}$).
- Our version of Geiger's Theorem can be generalized at least in two directions. On the one hand, we wish to extend it to $\mathbf{L}$-structures where $\mathbf{L}$ is an arbitrary (possibly non-locally finite) valuation structure. On the other hand, we aim towards a version of Geiger's Theorem for pp-formulas without the classical conjunction $\wedge$ holding for arbitrary valuation structures.
- Last but not least, the logical perspective outlined in this paper could contribute to development of universal algebra and model theory of weighted structures. For instance, one might try to develop the notion of pp-interpretability which is crucial for the algebraic approach to the classical CSP.

References

1 L. Barto and M. Kozik. Absorbing Subalgebras, Cyclic Terms, and the Constraint Satisfaction Problem. *Logical Methods in Computer Science*, 8(1):1–26, 2012.

2 L. Barto and M. Kozik. Constraint Satisfaction Problems Solvable by Local Consistency Methods. *Journal of the ACM*, 61(1):1–19, 2014.

3 S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint satisfaction and optimization. *Journal of the ACM*, 44(2):201–236, 1997.

4 M. Bodirsky. Complexity classification in infinite-domain constraint satisfaction. *CoRR*, abs/1201.0856, 2012. URL: `http://arxiv.org/abs/1201.0856`.

5 M. Bodirsky and V. Dalmau. Datalog and constraint satisfaction with infinite templates. *Journal of Computer and System Sciences*, 79(1):79–100, 2013.

6 M. Bodirsky and J. Nešetřil. Constraint Satisfaction with Countable Homogeneous Templates. *Journal of Logic and Computation*, 16(3):359–373, 2006.

7 V. G. Bodnarchuk, L. A. Kaluzhnin, V. N. Kotov, and B. A. Romov. Galois theory for post algebras. I. *Cybernetics*, 5(3):243–252, 1969.

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

9 A. Bulatov. A Dichotomy Theorem for Nonuniform CSPs. *CoRR*, abs/1703.03021, 2017. URL: `http://arxiv.org/abs/1703.03021`.

10 A. Bulatov and V. Dalmau. A Simple Algorithm for Mal'tsev Constraints. *SIAM Journal on Computing*, 36(1):16–27, 2006.

11 A. Bulatov, P. Jeavons, and A. Krokhin. Classifying the Complexity of Constraints Using Finite Algebras. *SIAM Journal on Computing*, 34(3):720–742, 2005.

12 D. Cohen, M. Cooper, P. Creed, P. Jeavons, and S. Živný. An Algebraic Theory of Complexity for Discrete Optimisation. *SIAM Journal on Computing*, 42(5):210–224, 2013.

13 D. Cohen, M. Cooper, P. G. Jeavons, and A. Krokhin. The complexity of soft constraint satisfaction. *Artificial Intelligence*, 170(11):983–1016, 2006.

14 D. A. Cohen. Tractable decision for a constraint language implies tractable search. *Constraints*, 9(3):219–229, 2004.

15 M. Cooper and T. Schiex. Arc consistency for soft constraints. *Artificial Intelligence*, 154(1-2):199–227, 2004.

16 F. Esteva and L. Godo. Monoidal t-norm based logic: towards a logic for left-continuous t-norms. *Fuzzy Sets and Systems*, 124(3):271–288, 2001.

17 T. Feder and M. Vardi. The Computational Structure of Monotone Monadic SNP and Constraint Satisfaction: A Study through Datalog and Group Theory. *SIAM Journal on Computing*, 28(1):57–104, 1998.

18 D. Geiger. Closed systems of functions and predicates. *Pacific Journal of Mathematics*, 27(1), 1968.

19 P. Idziak, R. Mckenzie, M. Valeriote, and R. Willard. Tractability and learnability arising from algebras with few subpowers. In *LICS*, pages 213–222, 2007.

20 P. Kolaitis and M. Vardi. Conjunctive-Query Containment and Constraint Satisfaction. *Journal of Computer and System Sciences*, 61(2):302–332, 2000.

21 V. Kolmogorov, A. Krokhin, and M. Rolinek. The Complexity of General-Valued CSPs. In *FOCS*, pages 1246–1258, 2015.

22 V. Kolmogorov, J. Thapper, and S. Živný. The Power of Linear Programming for General-Valued CSPs. *SIAM Journal on Computing*, 44(1):1–36, 2015.

23 M. Kozik and J. Ochremiak. Algebraic properties of valued constraint satisfaction problem. In *ICALP, Part I*, pages 846–858, 2015.

24 A. Rafiey, J. Kinne, and T. Feder. Dichotomy for digraph homomorphism problems. *CoRR*, abs/1701.02409, 2017. URL: `http://arxiv.org/abs/1701.02409`.

25 F. Rossi, P. van Beek, and T. Walsh. *Handbook of Constraint Programming (Foundations of Artificial Intelligence)*. Elsevier Science Inc., 2006.

26 M. Siggers. A strong Mal'cev condition for locally finite varieties omitting the unary type. *Algebra Universalis*, 64(1-2):15–20, 2010.

27 J. Thapper and S. Živný. The complexity of finite-valued CSPs. In *STOC*, page 695. ACM Press, 2013.

28 E. Vodolazskii, B. Flach, and M. Schlesinger. Minimax problems of discrete optimization invariant under majority operators. *Computational Mathematics and Mathematical Physics*, 54(8):1327–1336, 2014.

29 S. Živný. *The Complexity of Valued Constraint Satisfaction Problems*. Cognitive Technologies. Springer Berlin Heidelberg, 2012.

A Proof of Lemma 12

Mimicking the classical formulation, given two k-ary relations $R, Q\colon B^k \to L$, we will write $R \subseteq Q$ whenever for each $\vec{a} \in B^k$ it holds that $R(\vec{a}) \leq Q(\vec{a})$. We will also identify classical crisp k-ary relations on B with maps from B^k to L whose range is $\{0, 1\}$. Given a k-ary relation $R\colon B^k \to L$, we define its *support* by $\mathrm{Supp}(R) = \{\vec{b} \in B^k \mid R(\vec{b}) \neq 0\}$. For a matrix $\mathbb{A} \in B^{k \times n}$ we denote the set of its columns by $\mathrm{Cols}(\mathbb{A})$.

The proof of Lemma 12 is based on properties of closure operators induced by pp-definable relations. Recall that for every arity k the system of pp-definable relations form a closure system. Thus the pp-definable relations induces a family of closure operators $\gamma^{(k)}$. We abuse the notation and denote all these closure operators simply γ because the arity of arguments fully determines the corresponding closure operator.

▶ **Lemma 29.** *Let $R, S\colon B^k \to L$ be relations. Then $\gamma(R \odot S) \subseteq \gamma(R) \odot \gamma(S)$.*

$$\begin{tikzcd} B^k \arrow[dr, "R"] \arrow[dd, "\pi_l"'] & \\ & L \\ B^l \arrow[ur, "\pi_l[R]"'] & \end{tikzcd}$$

Figure 2 Projection.

Proof. We have $R \odot S \subseteq \gamma(R) \odot \gamma(S)$. Since the right-hand-side is pp-definable, this proves the claim. ◀

In what follows, we will show precisely how the transformations preserving pp-definable relations interact with the closure operators. There are three elementary transformations we are going to apply to relations, corresponding to existential quantification, permutation and identification of variables.

First, for $l < k$ there is a corresponding projection $\pi_l \colon B^k \to B^l$ (over the first l components). Given a relation $R\colon B^k \to L$, we define an l-ary relation $\pi_l[R]\colon B^l \to L$ by $\pi_l[R](\vec{a}) = \bigvee R[\pi_l^{-1}(\vec{a})]$.

Now we collect several properties of this transformation.

▶ **Lemma 30.** *Let $R, S\colon B^k \to L$, $T\colon B^l \to L$ and $\pi_l\colon B^k \to B^l$. Then*

1. *$R \subseteq S$ implies $\pi_l[R] \subseteq \pi_l[S]$.*
2. *$\pi_l[T\pi_l] = T$ where $T\pi_l$ is the composition of π_l followed by T (see Figure 3).*
3. *$R \subseteq \pi_l[R]\pi_l$. Moreover, for each $\vec{b} \in B^l$ there exists $\vec{b}' \in \pi_l^{-1}(\vec{b})$ such that $R(\vec{b}') = \pi_l[R](\vec{b})$.*
4. *Let $C \subseteq B^k$ be a crisp k-ary relation such that for each $\vec{a} \in B^l$ we have $C \cap \pi_l^{-1}(\vec{a}) \neq \emptyset$. Then $\pi_l[T\pi_l \odot C] = T$.*

Proof.

1. For $\vec{a} \in B^l$ we have by definition $\pi_l[R](\vec{a}) = \bigvee R[\pi_l^{-1}(\vec{a})] \leq \bigvee S[\pi_l^{-1}(\vec{a})] = \pi_l[S](\vec{a})$.
2. The composition $T\pi_l$ defines k-ary relation on B^k. Now pulling $T\pi_l$ along π_l gives $\pi_l[T\pi_l](\vec{a}) = \bigvee T\pi_l[\pi_l^{-1}(a)] = T(\vec{a})$ (see Figure 3).
3. Note that the composition of π_l followed by R approximates R from above, i.e., we have $R(\vec{a}) \leq \bigvee R[\pi_l^{-1}(\pi_l(\vec{a}))] = \pi_l[R](\pi_l(\vec{a}))$ for all $\vec{a} \in B^k$.
   Given $\vec{b} \in B^l$, consider $\vec{b}' \in \pi_l^{-1}(\vec{b})$ such that $R(\vec{b}')$ is maximum possible, i.e., $R(\vec{b}') = \bigvee R[\pi_l^{-1}(\vec{b})]$. Consequently, we have $\pi_l[R](\vec{b}) = \bigvee R[\pi_l^{-1}(\vec{b})] = R(\vec{b}')$.
4. We have $\pi_l[T\pi_l \odot C](\vec{a}) = \bigvee (T\pi_l \odot C)[\pi_l^{-1}(\vec{a})]$. Let $\vec{b} \in \pi_l^{-1}(\vec{a})$. If $C(\vec{b}) = 1$ then $(T\pi_l \odot C)(\vec{b}) = T\pi_l(\vec{b}) = T(\vec{a})$. If $C(\vec{b}) = 0$ then $(T\pi_l \odot C)(\vec{b}) = 0$. Since $C \cap \pi_l^{-1}(\vec{a}) \neq \emptyset$, we obtain $\pi_l[T\pi_l \odot C](\vec{a}) = T(\vec{a}) \vee 0 = T(\vec{a})$. ◀

Second, every permutation $\sigma\colon \{1,\dots,k\} \to \{1,\dots,k\}$ induces an automorphism $\sigma\colon B^k \to B^k$ by $\sigma(a_1,\dots,a_k) = \langle a_{\sigma(1)},\dots,a_{\sigma(n)}\rangle$. Given a relation $R\colon B^k \to L$, we can define a relation $\sigma[R] = R\sigma^{-1}$.

▶ **Lemma 31.** *Let $R, S\colon B^k \to L$ and σ a permutation on k elements. Then*

1. *$R \subseteq S$ implies $\sigma[R] \subseteq \sigma[S]$,*
2. *$\sigma[R\sigma] = R\sigma\sigma^{-1} = R$,*
3. *$\sigma[R]\sigma = R\sigma^{-1}\sigma = R$.*

Proof. For the first claim we have $\sigma[R](\vec{a}) = R(\sigma^{-1}(\vec{a})) \leq S(\sigma^{-1}(\vec{a})) = \sigma[S](\vec{a})$. The rest is obvious. ◀

$$\begin{tikzcd} B^k \arrow[dr, "T\pi_l"] \arrow[dd, "\pi_l"'] & \\ & L \\ B^l \arrow[ur, "T = \pi_l[T\pi_l]"'] & \end{tikzcd}$$

Figure 3 Projection identity.

$$\begin{tikzcd} B^k \arrow[dr, "R"] \arrow[dd, bend right, "\delta"'] & \\ & L \\ B^{k+|M|} \arrow[uu, bend right, "\pi_k"'] \arrow[ur, "\delta[R]"'] & \end{tikzcd}$$

Figure 4 Duplicating.

Finally, we introduce a duplicating operator. Let $M\colon \{1,\dots,k\} \to \mathbb{N}$ be a multiset, i.e., a function assigning to every $i \in \{1,\dots,k\}$ its multiplicity. The cardinality of M is defined by $|M| = \sum_{i=1}^{k} M(i)$. Define $\delta\colon B^k \to B^{k+|M|}$ by

$$\delta(a_1,\dots,a_k) = \langle a_1,\dots,a_k, \underbrace{a_1,\dots,a_1}_{M(1)},\dots,\underbrace{a_k,\dots,a_k}_{M(k)}\rangle,$$

i.e., δ duplicates coordinates according to M.

Given a k-ary relation $R\colon B^k \to L$, we define a $k+|M|$-ary relation $\delta[R]\colon B^{k+|M|} \to L$ by

$$\delta[R] = R\pi_k \odot E,$$

where E is the crisp relation expressing that the first coordinate x_1 equals to coordinates $x_{k+1},\dots,x_{k+M(1)}$, the second coordinate x_2 to coordinates $x_{k+M(1)+1},\dots,x_{k+M(1)+M(2)}$ etc. Thus

$$E = \left(\bigodot_{i=1}^{M(1)} (x_1 \approx x_{k+i})\right) \odot \left(\bigodot_{i=M(1)+1}^{M(1)+M(2)} (x_2 \approx x_{k+i})\right) \odot \dots$$

Note that $\delta[R]$ is pp-definable if R is pp-definable via the crisp equality relation which we assume to have in the language.

As in the previous cases, we can show several useful properties of the duplicating operators.

▶ **Lemma 32.** *Let $R, S\colon B^k \to L$ and $T\colon B^{k+|M|} \to L$. Then*

1. *$R \subseteq S$ implies $\delta[R] \subseteq \delta[S]$.*
2. *$\delta[R] \subseteq E$.*
3. *$\delta[T\delta] \subseteq T$. Moreover, if $T \subseteq E$ then $\delta[T\delta] = T$.*
4. *$\delta[R]\delta = R$.*
5. *$\pi_k[\delta[R]] = R$.*
6. *$T \subseteq E$ implies $\delta[\pi_k[T]] = T$.*

Proof.

1. We have that $\delta[R](\vec{a}) = R(\pi_k(\vec{a})) \odot E(\vec{a}) \leq S(\pi_k(\vec{a})) \odot E(\vec{a}) = \delta[S](\vec{a})$.

2. Follows from the definition of δ.
3. We have $\delta[T\delta](\vec{a}) = T\delta(\pi_k(\vec{a})) \odot E(\vec{a}) \leq T(\vec{a})$. The last inequality holds because if $E(\vec{a}) \neq 0$ (i.e., $E(\vec{a}) = 1$) then $\delta\pi_k(\vec{a}) = \vec{a}$.
4. We have $\delta[R](\delta(\vec{a})) = R(\pi_k(\delta(\vec{a}))) \odot E(\delta(\vec{a})) = R(\vec{a})$.
5. We have $\pi_k[\delta[R]](\vec{a}) = \bigvee \delta[R][\pi_k^{-1}(\vec{a})]$. Let $\vec{b} \in \pi_k^{-1}(\vec{a})$. Then

$$\delta[R](\vec{b}) = R(\pi_k(\vec{b})) \odot E(\vec{b}) = \begin{cases} R(\vec{a}) & \text{if } E(\vec{b}) = 1, \\ 0 & \text{otherwise.} \end{cases}$$

   In particular, for $\vec{b} = \delta(\vec{a}) \in \pi_k^{-1}(\vec{a}) \cap E$ we have $\delta[R](\delta(\vec{a})) = R(\vec{a})$. Thus $\pi_k[\delta[R]](\vec{a}) = R(\vec{a})$.
6. We have $\delta[\pi_k[T]](\vec{a}) = \pi_k[T](\pi_k(\vec{a})) \odot E(\vec{a})$. If $E(\vec{a}) = 0$ then $\delta[\pi_k[T]](\vec{a}) = 0$ and $T(\vec{a}) = 0$ as well because $T \subseteq E$. Suppose that $E(\vec{a}) = 1$. We have $\delta[\pi_k[T]](\vec{a}) = \pi_k[T](\pi_k(\vec{a})) = \bigvee T[\pi_k^{-1}\pi_k(\vec{a})] = T(\vec{a})$. The last equality follows from the fact that $\pi_k^{-1}\pi_k(\vec{a}) \cap E = \{\vec{a}\}$. ◀

Relying on the previously proven properties, we can show that the closure operator γ commutes with projections, permutations and duplications.

▶ **Lemma 33.** *Let $R\colon B^k \to L$. Then*
1. *For $l < k$ and $\pi_l\colon B^k \to B^l$ a projection it holds that $\gamma(\pi_l[R]) = \pi_l[\gamma(R)]$.*
2. *For $\sigma\colon B^k \to B^k$ a permutation it holds that $\gamma(\sigma[R]) = \sigma[\gamma(R)]$.*
3. *For $\delta\colon B^k \to B^{k+|M|}$ a duplication it holds that $\gamma(\delta[R]) = \delta[\gamma(R)]$.*

Proof.
1. Since $R \subseteq \gamma(R)$, we have $\pi_l[R] \subseteq \pi_l[\gamma(R)]$ by Lemma 30. Thus $\gamma(\pi_l[R]) \subseteq \pi_l[\gamma(R)]$ follows since pp-definable relations are closed under projections.
   Conversely, we have $R \subseteq \pi_l[R]\pi_l$ by Lemma 30. Further, $\pi_l[R]\pi_l \subseteq \gamma(\pi_l[R])\pi_l$. Since the right-hand side is pp-definable, we obtain $\gamma(R) \subseteq \gamma(\pi_l[R]\pi_l) \subseteq \gamma(\pi_l[R])\pi_l$. Applying π_l to both sides, we obtain $\pi_l[\gamma(R)] \subseteq \gamma(\pi_l[R])$ by Lemma 30.
2. We have $\sigma[R] \subseteq \sigma[\gamma(R)]$ by Lemma 31. Since γ-closed sets are closed under permutations of coordinates, $\gamma(\sigma[R]) \subseteq \sigma[\gamma(R)]$.
   Conversely, we have $\sigma[R] \subseteq \gamma(\sigma[R])$ by the properties of closure operators. By Lemma 31 we have $R = \sigma^{-1}[\sigma[R]] \subseteq \sigma^{-1}[\gamma(\sigma[R])]$. Thus also $\gamma(R) \subseteq \sigma^{-1}[\gamma\sigma[R])]$ since pp-definable relations are closed under permutations of coordinates. Consequently, by Lemma 31 $\sigma[\gamma(R)] \subseteq \sigma[\sigma^{-1}[\gamma(\sigma[R])]] = \gamma(\sigma[R])$.
3. Since $\delta[\gamma(R)]$ is pp-definable, we have $\gamma\delta[R] \subseteq \delta[\gamma(R)]$ using also Lemma 32.
   Conversely we have $\delta[R] \subseteq \gamma(\delta[R])$. Thus by Lemma 30 and 32 we get $R = \pi_k[\delta[R]] \subseteq \pi_k[\gamma(\delta[R])]$. Since the right-hand-side is pp-definable, $\gamma(R) \subseteq \pi_k[\gamma(\delta[R])]$ follows. Applying δ and using Lemma 32, we obtain $\delta[\gamma(R)] \subseteq \delta[\pi_k[\gamma(\delta[R])]]$. Since $\delta[R] \subseteq E$, we have $\gamma(\delta[R]) \subseteq E$ because E is pp-definable. Finally, using Lemma 32, we have $delta[\gamma(R)] \subseteq \delta[\pi_k[\gamma(\delta[R])]] = \gamma(\delta[R])$. ◀

Next we present a useful characterization of polymorphisms. Recall that we extent an n-ary map $f\colon B^n \to B$ so that it acts on matrices $\mathbb{A} \in B^{k\times n}$. Since f acts row-wise on $\mathbb{A}$, it commutes with the projections π_l, permutations σ and duplications δ. In particular, we have $\pi_l\sigma\delta f(\mathbb{A}) = f\pi_l\sigma\delta(\mathbb{A})$ for all $\mathbb{A} \in B^{k\times n}$.

For a given $k \geq 1$ we will define a matrix $\mathbb{C}_k \in B^{k\times|B|^k}$. The matrix $\mathbb{C}_k$ is the matrix whose *columns* are all k-tuples from B^k listed in a linear order, i.e., we have $\mathrm{Cols}(\mathbb{C}_k) = B^k$. Observe that $\mathbb{C}_k$ has pairwise distinct rows. Moreover, the transpose matrix $\mathbb{C}_k^T$ is the matrix whose *rows* are all k-tuples from B^k.

▶ **Lemma 34.** *Let $f\colon B^n \to B$ and $\tau\colon L^n \to L$ such that*

$$\tau(\vec{\alpha} \wedge \vec{\beta}) \leq \tau(\vec{\alpha}) \wedge \tau(\vec{\beta}), \qquad \tau(\vec{\alpha} \odot \vec{\beta}) \leq \tau(\vec{\alpha}) \odot \tau(\vec{\beta}).$$

The tuple $\langle f, \tau\rangle$ is a polymorphism iff for all relations $S\colon B^{|B|^n} \to L$ we have

$$\tau S(\mathbb{C}_n^T) \leq \gamma(S)(f(\mathbb{C}_n^T)).$$

Proof. The left-to-right direction follows since polymorphisms preserve all pp-definable relations. More precisely, since τ is monotone, we have $\tau S(\mathbb{C}_n^T) \leq \tau\gamma(S)(\mathbb{C}_n^T) \leq \gamma(S)(f(\mathbb{C}_n^T))$.

Conversely, suppose $\tau S(\mathbb{C}_n^T) \leq \gamma(S)(f(\mathbb{C}_n^T))$ for all $S\colon B^{|B|^n} \to L$. Let $R\colon B^k \to L$ be a k-ary relation in $\mathbf{B}$ and $\mathbb{A} \in B^{k\times n}$. We have to show that $\tau R(\mathbb{A}) \leq Rf(\mathbb{A})$.

Since the $\mathrm{Rows}(\mathbb{C}_n^T) = B^n$, there is a transformation $\pi_k\sigma\delta\colon B^{|B|^n} \to B^k$ such that $\mathbb{A} = \pi_k\sigma\delta(\mathbb{C}_n^T)$. In other words, applying a suitable duplicating operator followed by a permutation and then by a projection to the matrix $\mathbb{C}_n^T$, we can produce the matrix $\mathbb{A}$. Define $S\colon B^{|B|^n} \to L$ by the composition $R\pi_k\sigma\delta$. By the assumption $\tau S(\mathbb{C}_n^T) \leq \gamma(S)f(\mathbb{C}_n^T)$, Lemma 30, 31, 32 on properties of projections, permutations, duplicating operator and Lemma 33 on properties of the γ's operators, we have the following chain of (in)equalities that concludes the proof:

$$\begin{aligned}
\tau R(\mathbb{A}) &= \tau R\pi_k\sigma\delta(\mathbb{C}_n^T) = \tau S(\mathbb{C}_n^T) \leq \gamma(S)(f(\mathbb{C}_n^T)) = \delta[\gamma(S)](\delta f(\mathbb{C}_n^T)) = \gamma(\delta[S])(f\delta(\mathbb{C}_n^T)) \\
&= \gamma(\delta[R\pi_k\sigma\delta])(f\delta(\mathbb{C}_n^T)) \leq \gamma(R\pi_k\sigma)(f\delta(\mathbb{C}_n^T)) = \sigma[\gamma(R\pi_k\sigma)](\sigma f\delta(\mathbb{C}_n^T)) \\
&= \gamma(\sigma[R\pi_k\sigma])(f\sigma\delta(\mathbb{C}_n^T)) = \gamma(R\pi_k)(f\sigma\delta(\mathbb{C}_n^T)) \leq \pi_k[\gamma(R\pi_k)](\pi_k f\sigma\delta(\mathbb{C}_n^T)) \\
&= \gamma(\pi_k[R\pi_k])(f\pi_k\sigma\delta(\mathbb{C}_n^T)) = \gamma(R)(f(\mathbb{A})) = Rf(\mathbb{A}).
\end{aligned}$$ ◀

Let $\mathbb{A} \in B^{k\times n}$ and $\vec{b} \in B^k$. There is one-to-one correspondence between elements of L^n and relations $R\colon B^k \to L$ such that $\mathrm{Supp}(R) = \mathrm{Cols}(\mathbb{A})$. Given a tuple $\vec{\alpha} \in L^n$, the corresponding relation $R_{\vec{\alpha}}\colon B^k \to L$ is the relation whose support is $\mathrm{Cols}(\mathbb{A})$ and is defined by $R_{\vec{\alpha}}(\mathbb{A}) = \vec{\alpha}$.

▶ **Lemma 35.** *The map $\tau\colon L^n \to L$ defined by $\tau(\vec{\alpha}) = \gamma(R_{\vec{\alpha}})(\vec{b})$ satisfies*

$$\tau(\vec{\alpha} \wedge \vec{\beta}) \leq \tau(\vec{\alpha}) \wedge \tau(\vec{\beta}) \qquad \textit{and} \qquad \tau(\vec{\alpha} \odot \vec{\beta}) \leq \tau(\vec{\alpha}) \odot \tau(\vec{\beta}).$$

Proof. Let $\vec{\alpha}, \vec{\beta} \in L^n$. The first inequality is equivalent to τ being monotone in all arguments. To see that τ is monotone, note that $\vec{\alpha} \leq \vec{\beta}$ implies $\gamma(R_{\vec{\alpha}}) \subseteq \gamma(R_{\vec{\beta}})$. Thus we have

$$\tau(\vec{\alpha}) = \gamma(R_{\vec{\alpha}})(\vec{b}) \leq \gamma(R_{\vec{\beta}})(\vec{b}) = \tau(\vec{\beta}).$$

Further, we have $R_{\vec{\alpha}\odot\vec{\beta}} = R_{\vec{\alpha}} \odot R_{\vec{\beta}}$. Using Lemma 29, we get

$$\tau(\vec{\alpha} \odot \vec{\beta}) = \gamma(R_{\vec{\alpha}} \odot R_{\vec{\beta}})(\vec{b}) \leq \gamma(R_{\vec{\alpha}})(\vec{b}) \odot \gamma(R_{\vec{\beta}})(\vec{b}) = \tau(\vec{\alpha}) \odot \tau(\vec{\beta}).$$ ◀

Now we are ready to finish the proof of the extension lemma.

Proof of Lemma 12. Let $n = |B|^k$. Note that the matrix $\mathbb{C}_k$ cannot contain a row more than once. Thus one can extend the matrix $\mathbb{C}_k$ by adding new rows to a matrix $\mathbb{D}_k \in B^{|B|^n\times n}$ so that the rows of $\mathbb{D}_k$ consists of all n-tuples from B^n and $\mathbb{C}_k = \pi_k(\mathbb{D}_k)$ where π_k is applied column-wise. Let $T\colon B^{|B|^n} \to L$ be a relation such that $T = R\pi_k \odot \mathrm{Cols}(\mathbb{D}_k)$. Note that $T(\mathbb{D}_k) = R(\mathbb{C}_k)$. By Lemma 30 it follows that $\pi_k[T] = R$. Indeed, for each $\vec{a} \in B^k$ (i.e., $\vec{a}$ is a j-th column of $\mathbb{C}_k$) there is a column $\vec{a}'$ in $\mathbb{D}_k$ such that $\pi_k(\vec{a}') = \vec{a}$. Thus we have $\mathrm{Cols}(\mathbb{D}_k) \cap \pi_k^{-1}(\vec{a}) \neq \emptyset$.

Further by Lemma 30 there exists $\vec{b}' \in \pi_k^{-1}(\vec{b})$ such that $\pi_k[\gamma(T)](\vec{b}) = \gamma(T)(\vec{b}')$. We define $f(\mathbb{D}_k) = \vec{b}'$. Further we define $\tau(\vec{\alpha}) = \gamma(R_{\vec{\alpha}})(\vec{b}')$ as in Lemma 35 for $\mathrm{Supp}(R_{\vec{\alpha}}) = \mathrm{Cols}(\mathbb{D}_k)$.

In order to prove that $\langle f, \tau\rangle$ is a polymorphism, it suffices to show by Lemma 34 that $\tau S(\mathbb{D}_k) \leq \gamma(S)(f(\mathbb{D}_k))$ for all relations $S\colon B^{|B|^n} \to L$.[12] Let $S\colon B^{|B|^n} \to L$. There is $\vec{\alpha} \in L^n$ such that $R_{\vec{\alpha}} = S \odot \mathrm{Cols}(\mathbb{D}_k)$. By definition of τ we have

$$\tau S(\mathbb{D}_k) = \tau R_{\vec{\alpha}}(\mathbb{D}_k) = \gamma(R_{\vec{\alpha}})(\vec{b}') \leq \gamma(S)(\vec{b}') = \gamma(S)(f(\mathbb{D}_k)).$$

Thus $\langle f, \tau\rangle$ is a polymorphism.

It remains to prove that $f(\mathbb{C}_k) = \vec{b}$ and $\tau R(\mathbb{C}_k) = \gamma(R)(\vec{b})$. To see the first equality, observe that

$$f(\mathbb{C}_k) = f(\pi_k(\mathbb{D}_k)) = \pi_k(f(\mathbb{D}_k)) = \pi_k(\vec{b}') = \vec{b}.$$

Regarding the second equality, since $\mathrm{Supp}(T) = \mathrm{Cols}(\mathbb{D}_k)$, we have $T = R_{\vec{\alpha}}$ for some $\vec{\alpha} \in L^n$. Consequently,

$$\tau R(\mathbb{C}_k) = \tau T(\mathbb{D}_k) = \gamma(T)(\vec{b}') = \pi_k[\gamma(T)](\vec{b}) = \gamma(\pi_k[T])(\vec{b}) = \gamma(R)(\vec{b}). \qquad \blacktriangleleft$$

B Proof of Lemma 23

Regarding the proof of the right-to-left implication, assume that $f \in \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$ for all $\alpha \in L$. Observe that the relation $R_{\vec{\alpha}}$ is a particular case of the relations introduced before Lemma 35 (with support $\mathbb{C}_n^T$). Thus τ_f satisfies

$$\tau_f(\vec{\alpha} \wedge \vec{\beta}) \leq \tau_f(\vec{\alpha}) \wedge \tau_f(\vec{\beta}) \qquad \text{and} \qquad \tau_f(\vec{\alpha} \odot \vec{\beta}) \leq \tau_f(\vec{\alpha}) \odot \tau_f(\vec{\beta}),$$

as a consequence of Lemma 35. Moreover, following the same reasoning as in the proof of Lemma 12, we know it is a polymorphism of $\mathbf{B}$ (relying on Lemma 34).

It suffices then to show that τ_f is increasing in α. Let $\vec{\alpha} = \langle \alpha, \dots, \alpha\rangle$ be the vector of length n. Since the relation $\gamma(R_{\vec{\alpha}})$ is characterized by a $|B|^n$-ary pp-formula $\varphi(\vec{x})$ in $\mathbf{B}$, $\widehat{\varphi}(\vec{x})$ is a pp-formula in $\widehat{\mathbf{B}}_\alpha$. By definition $\vec{\alpha} = R_{\vec{\alpha}}(\mathbb{C}_n^T) \leq \gamma(R_{\vec{\alpha}})(\mathbb{C}_n^T) = \|\varphi(\mathbb{C}_n^T)\|^{\mathbf{B}}$. Thus, by Lemma 15 it follows that the columns of $\mathbb{C}_n^T$ belong to the relation pp-defined by $\widehat{\varphi}(\vec{x})$ in the classical structure $\widehat{\mathbf{B}}_\alpha$.

Since $f \in \mathrm{Pol}(\widehat{\mathbf{B}}_\alpha)$, it preserves pp-definable formulas, in particular $\widehat{\varphi}(\vec{x})$. Thus $f(\mathbb{C}_n^T)$ belongs to the relation pp-defined by $\widehat{\varphi}(\vec{x})$ in $\widehat{\mathbf{B}}_\alpha$. Consequently, Lemma 15 implies

$$\alpha \leq \|\varphi(f(\mathbb{C}_n^T))\|^{\mathbf{B}} = \gamma(R_{\vec{\alpha}})(f(\mathbb{C}_n^T)) = \tau_f(\vec{\alpha})$$

which concludes the proof. ◀

[12] Observe that $\mathbb{C}_n^T = \mathbb{D}_k$.