On Minimal Weighted Clones*

Páidí Creed[1] and Stanislav Živný[2]

[1] Department of Computer Science, Royal Holloway, University of London, UK
[2] University College, University of Oxford, UK
paidi@rhul.ac.uk standa.zivny@cs.ox.ac.uk

Abstract. The connection between the complexity of constraint languages and clone theory, discovered by Cohen and Jeavons in a series of papers, has been a fruitful line of research on the complexity of CSPs. In a recent result, Cohen et al. [14] have established a Galois connection between the complexity of valued constraint languages and so-called weighted clones. In this paper, we initiate the study of weighted clones. Firstly, we prove an analogue of Rosenberg's classification of minimal clones for weighted clones. Secondly, we show minimality of several weighted clones whose support clone is generated by a single minimal operation. Finally, we classify all Boolean weighted clones. This classification implies a complexity classification of Boolean valued constraint languages obtained by Cohen et al. [13]

1 Introduction

The general constraint satisfaction problem (CSP) is NP-hard, and so is unlikely to have a polynomial-time algorithm. However, there has been much success in finding tractable fragments of the CSP by restricting the types of relation allowed in the constraints. A set of allowed relations has been called a *constraint language* [20]. For some constraint languages the associated constraint satisfaction problems with constraints chosen from that language are solvable in polynomial-time, whilst for other constraint languages this class of problems is NP-hard [21,20,19]; these are referred to as *tractable languages* and *NP-hard languages*, respectively. Dichotomy theorems, which classify each possible constraint language as either tractable or NP-hard, have been established for constraint languages over 2-element domains [27], 3-element domains [8], for conservative constraint languages [10,3], and maximal constraint languages [6,7].

The general *valued* constraint satisfaction problem (VCSP) is also NP-hard, but again we can try to identify tractable fragments by restricting the types of allowed *cost functions* that can be used to define the valued constraints. A set of allowed cost functions has been called a *valued constraint language* [13]. Much less is known about the complexity of the optimisation problems associated with different valued constraint languages, although some results have been obtained for certain special cases. In particular, a complete characterisation of complexity

* This research was supported by EPSRC grant EP/F01161X/1. Stanislav Živný is supported by a Junior Research Fellowship at University College, Oxford.

has been obtained for valued constraint languages over a 2-element domain with real-valued or infinite costs [13]. This result generalises a number of earlier results for particular optimisation problems such as Max-Sat [15] and Min-Ones [16]. Recently, the complete classification of conservative valued languages has been obtained for finite-valued [22] and general-valued languages [23].

In the classical CSP framework it has been shown that the complexity of any constraint language over any finite domain is determined by certain algebraic properties known as *polymorphisms* [21,20]. This result has reduced the problem of the identification of tractable constraint languages to that of the identification of suitable sets of polymorphisms. The set of polymorphisms of a constraint language forms a *clone of operations* and a tight (one to one) correspondence has been shown to exist between clones and constraint languages (closed under expressibility). In other words, we can study properties of constraint languages by studying properties of clones. This algebraic approach has been laid out in detail in [9] and has already proved fruitful in classifying the complexity of constraint languages over finite domains of arbitrary size [19,9,2,4,1,5]. In particular, by considering the set of *minimal clones*, it has been possible to classify the complexity of all *maximal constraint languages* on a finite domain D [6,7] (these are the constraint languages which can express all relations over D if we add a single new type of constraint).

Recently, it has been shown that the complexity of valued constraint languages can be determined by studying properties known as *weighted polymorphisms*[3] [11,14]. The set of weighted polymorphisms of any valued constraint language form an object called a *weighted clone* and is has been shown that there exists a tight (one to one) connection between weighted clones and valued constraint languages (closed under expressibility) [14]. Previously, a special type of weighted polymorphism, called a *multimorphism*, has been used to analyse the complexity of certain valued constraint languages [13]. In particular, multimorphisms have been used to show that there are precisely eight maximal tractable valued constraint languages over a 2-element domain with real-valued or infinite costs, and each of these is characterised by having a particular form of multimorphism [13]. Furthermore, it was shown that many known maximal tractable valued constraint languages over larger finite domains are precisely characterised by a single multimorphism and that key NP-hard examples have (essentially) no multimorphisms [13,12].

Contributions In this paper, we initiate the study of weighted clones. In particular, we focus on *minimal* weighted clones, which define maximal valued constraint languages. As the main contribution, we demonstrate that the theory developed by Cohen et al. [14] can be used for answering non-trivial questions concerning the complexity of valued constraint languages. We see this paper as a first step towards using the theory of weighted clones in the study of the complexity of valued constraint languages. We believe that the techniques from this paper can be used for other problems as well.

[3] In [11] these were called *fractional polymorphisms*.

On the technical side, we prove a Rosenberg-type classification for minimal weighted clones. Furthermore, we prove minimality of several interesting weighted clones, which correspond to well-studied maximal valued constraint languages. Finally, for Boolean domains, we provide a complete classification of weighted clones. This implies a complexity classification of Boolean valued constraint languages.

Paper organisation The rest of the paper is organised as follows. In Section 2, we define valued constraint satisfaction problems (VCSPs), the notion of expressibility, weighted operations and weighted clones. In Section 3 we prove an analogue of Rosenberg's Classification Theorem [26] for weighted clones, which establishes certain properties minimal weighted clones must satisfy. Then, in Section 4 we give several examples of minimal weighted clones. Finally, in Section 5 we show how the results of the preceding sections can be used to obtain the Boolean classification of [13]. Although this paper does not identify any novel tractable valued constraint languages, we believe the tools described herein will prove invaluable in future efforts to identify the tractable cases of the VCSP.

2 Preliminaries

2.1 VCSP

We will use $[k]$ to denote the set $\{1, \ldots, k\}$ for any positive integer k. We shall denote by $\mathbb{Q}_+$ the set of all non-negative rational numbers. We define $\overline{\mathbb{Q}}_+ = \mathbb{Q}_+ \cup \{\infty\}$ with the standard addition operation extended so that for all $a \in \mathbb{Q}_+$, $a + \infty = \infty$. Members of $\overline{\mathbb{Q}}_+$ are called *costs*. Throughout the paper, we denote by D any fixed finite set, called a *domain*, consisting of *values*.

A function ϕ from D^r to $\overline{\mathbb{Q}}_+$ will be called a *cost function* on D of *arity* r. If the range of ϕ lies entirely within $\mathbb{Q}_+$, then ϕ is called a *finite-valued* cost function. If the range of ϕ is $\{0, \infty\}$, then ϕ is called a *crisp* cost function. A *language* is a set of cost functions with the same domain D. Language Γ is called finite-valued (crisp) if all cost functions in Γ are finite-valued (crisp). A language Γ is *Boolean* if $|D| = 2$.

Definition 1. *An instance of the* **valued constraint satisfaction problem**, *(***VCSP***), is a 3-tuple* $\mathcal{P} = \langle V, D, C \rangle$ *where* V *is a finite set of* **variables***;* D *is a set of possible* **values***;* C *is a multi-set of* **constraints***. Each element of* C *is a pair* $c = \langle \sigma, \phi \rangle$ *where* σ *is a tuple of variables called the* **scope** *of* c*, and* $\phi : D^{|\sigma|} \to \overline{\mathbb{Q}}_+$ *is a* $|\sigma|$*-ary cost function on* D*. An* **assignment** *for* $\mathcal{P}$ *is a mapping* $s : V \to D$*. The* **cost** *of an assignment* s*, denoted* $Cost_{\mathcal{P}}(s)$*, is given by the sum of the costs for the restrictions of* s *onto each constraint scope, that is,*

$$Cost_{\mathcal{P}}(s) \stackrel{\text{def}}{=} \sum_{\langle\langle v_1, v_2, \ldots, v_m \rangle, \phi\rangle \in C} \phi(s(v_1), s(v_2), \ldots, s(v_m)).$$

A **solution** *to* $\mathcal{P}$ *is an assignment with minimal cost, and the question is to find a solution.*

We define $\mathrm{VCSP}(\Gamma)$ to be the set of all VCSP instances in which all cost functions belong to Γ. A valued constraint language Γ is called **tractable** if, for every finite subset $\Gamma_f \subseteq \Gamma$, there exists an algorithm solving any instance $\mathcal{P} \in \mathrm{VCSP}(\Gamma_f)$ in polynomial time. Conversely, Γ is called **NP-hard** if there is some finite subset $\Gamma_f \subseteq \Gamma$ for which $\mathrm{VCSP}(\Gamma_f)$ is NP-hard.

2.2 Weighted Relational Clones

We denote by $\mathbf{\Phi}_D$ the set of cost functions on D taking values in $\overline{\mathbb{Q}}_+$ and by $\mathbf{\Phi}_D^{(r)}$ the r-ary cost functions in $\mathbf{\Phi}_D$. Any cost function $\phi : D^r \to \overline{\mathbb{Q}}_+$ has an associated cost function which takes only the values 0 and ∞, known as its **feasibility relation**, denoted $\mathrm{Feas}(\phi)$, which is defined as $\mathrm{Feas}(\phi)(x_1, \dots, x_r) = 0$ if $\phi(x_1, \dots, x_r) < \infty$, and $\mathrm{Feas}(\phi)(x_1, \dots, x_r) = \infty$ otherwise.

We say $\phi, \phi' \in \mathbf{\Phi}_D$ are **cost-equivalent**, denoted by $\phi \sim \phi'$, if there exist $\alpha, \beta \in \mathbb{Q}_+$ with $\alpha > 0$ such that $\phi = \alpha\phi' + \beta$. We denote by $\Gamma_\sim$ the smallest set of cost functions containing Γ which is closed under cost-equivalence.

We now define a closure operator on cost functions, which adds to a set of cost functions all other cost functions which can be obtained from that set by minimising over a subset of variables:

Definition 2. *For any VCSP instance $\mathcal{P} = \langle V, D, C\rangle$, and any list $L = \langle v_1, \dots, v_r\rangle$ of variables of $\mathcal{P}$, the* **projection** *of $\mathcal{P}$ onto L, denoted $\pi_L(\mathcal{P})$, is the r-ary cost function defined as follows:*

$$\pi_L(\mathcal{P})(x_1, \dots, x_r) \stackrel{\text{def}}{=} \min_{\{s:V \to D \ \mid \ \langle s(v_1),\dots,s(v_r)\rangle = \langle x_1,\dots,x_r\rangle\}} Cost_{\mathcal{P}}(s)\,.$$

We say that a cost function ϕ is **expressible** *over a constraint language Γ if there exists a VCSP instance $\mathcal{P} \in \mathrm{VCSP}(\Gamma)$ and a list L of variables of $\mathcal{P}$ such that $\pi_L(\mathcal{P}) = \phi$. We define $\mathrm{Express}(\Gamma)$ to be the* **expressive power** *of Γ; that is, the set of all cost functions expressible over Γ.*

Note that the list of variables L may contain repeated entries, and we define the minimum over an empty set of costs to be ∞.

Example 1. Let $\mathcal{P}$ be the VCSP instance with a single variable v and no constraints, and let $L = \langle v, v\rangle$. Then, by Definition 2,

$$\pi_L(\mathcal{P})(x, y) = \begin{cases} 0 & \text{if } x = y \\ \infty & \text{otherwise} \end{cases}\,.$$

Hence for any valued constraint language Γ, over any set D, $\mathrm{Express}(\Gamma)$ contains this binary cost function, which will be called the **equality** cost function.

Definition 3. *We say a set $\Gamma \subseteq \mathbf{\Phi}_D$ is a* **weighted relational clone** *if it contains the equality cost function and is closed under cost-equivalence and feasibility, rearrangement of arguments, addition of cost functions, and expressibility. For each $\Gamma \subseteq \mathbf{\Phi}_D$ we define $\mathrm{wRelClone}(\Gamma)$ to be the smallest weighted relational clone containing Γ.*

It is known that for any $\Gamma \subseteq \mathbf{\Phi}_D$, $\text{Express}(\Gamma \cup \text{Feas}(\Gamma))_\sim = \text{wRelClone}(\Gamma)$ [14]. Moreover, it follows from [11] that Γ is tractable if and only if $\text{wRelClone}(\Gamma)$ is tractable. Hence, the search for tractable valued constraint languages corresponds to a search for suitable weighted relational clones.

2.3 Weighted Clones

First we recall some basic terminology from clone theory [18]. A function $f : D^k \to D$ is called a k-ary **operation** on D. We denote by $\mathbf{O}_D$ the set of all finitary operations on D and by $\mathbf{O}_D^{(k)}$ the k-ary operations in $\mathbf{O}_D$. The k-ary **projections** on D, defined for $i = 1, \ldots, k$, are the operations $\mathrm{e}_i^{(k)}(a_1, \ldots, a_k) = a_i$. (We drop the superscript (k) if it is clear from the context.) Let $f \in \mathbf{O}_D^{(k)}$ and $g_1, \ldots, g_k \in \mathbf{O}_D^{(l)}$. The **superposition** of f and $g_1, \ldots, g_k$ is the l-ary operation $f[g_1, \ldots, g_k](x_1, \ldots, x_l) = f(g_1(x_1, \ldots, x_l), \ldots, g_k(x_1 \ldots, x_l))$.

A set $F \subseteq \mathbf{O}_D$ is called a **clone** of operations if it contains all the projections on D and is closed under superposition.

For each $F \subseteq \mathbf{O}_D$ we define $\text{Clone}(F)$ to be the smallest clone containing F. For any clone C, we use $C^{(k)}$ to denote the k-ary terms in C. We say a clone C is **minimal** if any non-trivial operation in C generates C, i.e. for all $f \in C$ other than the projections, we have $C = \text{Clone}(\{f\})$. An operation f in a minimal clone C is called **minimal** if f has smallest arity among the non-trivial operations in C.

It has been shown [21] that crisp constraint languages are in one to one correspondence with clones. Recently, Cohen et al. [14] have shown that a similar correspondence exists between valued constraint languages and objects called **weighted clones**. We will now briefly describe their results.

Definition 4. *We define a k-ary* **weighted operation** *on a set D to be a function $\omega : \mathbf{O}_D^{(k)} \to \mathbb{Q}$ such that $\omega(f) < 0$ only if f is a projection and*

$$\sum_{f \in \mathbf{dom}(\omega)} \omega(f) = 0\,.$$

The **domain** *of ω, denoted $\mathbf{dom}(\omega)$, is the subset of $\mathbf{O}_D^{(k)}$ on which ω is defined. We denote by $ar(\omega) = k$ the arity of ω.*

We denote by $\mathbf{W}_D$ the finitary weighted operations on D and by $\mathbf{W}_D^{(k)}$ the k-ary weighted operations in $\mathbf{W}_D$.

Definition 5. *Let C be a clone of operations on D. We define the k-ary* **zero weighted operation** *supported by C to be the k-ary weighted operation which satisfies $\omega(f) = 0$ for all $f \in C^{(k)}$.*

Definition 6. *Let C be a clone of operations on D. A* **weighted clone supported** *by C is a set of weighted operations that contains all zero-weighted operations whose domains are subsets of C and is closed under:*

proper translation *Given a k-ary weighted operation $\omega : C^{(k)} \to \mathbb{Q}$ and $\mathbf{t} = \langle g_1, \dots, g_k \rangle$, where $g_1, \dots, g_k \in C^{(\ell)}$, we define the* **translation** *of ω by $g_1, \dots, g_k$, denoted as $\omega[g_1, \dots, g_k]$ or simply $\omega[\mathbf{t}]$, to be the function $\omega' : C^{(\ell)} \to \mathbb{Q}$ satisfying*

$$\omega'(f') = \sum_{f \in C^{(k)} : f' = f[g_1, \dots, g_k]} \omega(f)\,,$$

for each $f' \in C^{(\ell)}$. A translation is called a **proper translation** *if ω' is a weighted operation.*

addition *Given a pair of k-ary weighted operations $\omega_1, \omega_2 : C^{(k)} \to \mathbb{Q}$, we define the* **addition** *$\omega_1 + \omega_2$ to be the weighted operation ω' satisfying*

$$\omega'(f) = \omega_1(f) + \omega_2(f)\,,$$

for each $f \in C^{(k)}$.

scaling *Let ω be a k-ary weighted operation supported by C and let $\alpha > 0$. We define the α-scaling of ω, $\alpha\omega$, to be the weighted operation ω' satisfying*

$$\omega'(f) = \alpha\omega(f)\,,$$

for each $f \in C^{(k)}$.

Example 2. Let ω be the 4-ary weighted operation on D given by

$$\omega(f) = \begin{cases} -1 & \text{if } f \text{ is a projection} \\ +1 & \text{if } f \in \{\max(x_1, x_2), \min(x_1, x_2), \max(x_3, x_4), \min(x_3, x_4)\} \end{cases}\,,$$

and let

$$\langle g_1, g_2, g_3, g_4 \rangle = \left\langle e_1^{(3)}, e_2^{(3)}, e_3^{(3)}, \max(x_1, x_2) \right\rangle\,.$$

Then, by Definition 6, the translation of ω by $\langle g_1, g_2, g_2, g_3 \rangle$ is:

$$\omega[g_1, g_2, g_3, g_4](f) = \begin{cases} -1 & \text{if } f \text{ is a projection} \\ +1 & \text{if } f \in \{\max(x_1, x_2, x_3), \min(x_1, x_2), \min(x_3, \max(x_1, x_2))\} \\ 0 & \text{if } f = \max(x_1, x_2) \end{cases}\,.$$

Note that $\omega[g_1, g_2, g_3, g_4]$ satisfies the conditions of Definition 4 and hence is a weighted operation. Hence the translation is proper.

For each $W \subseteq \mathbf{W}_D$ we define $\mathrm{wClone}(W)$ to be the smallest weighted clone containing W. In particular, we write $\mathrm{wClone}(\omega)$ for the smallest weighted clone containing weighted operation ω. Note that the support of $\mathrm{wClone}(W)$ is the clone generated by the domains of the elements of W; i.e. the support of $\mathrm{wClone}(W)$ is given by $\mathrm{Clone}(\cup_{\omega \in W} \mathbf{dom}(\omega))$. The following is a direct consequence of the definition of weighted clones.

Proposition 1. *Let ω be a weighted operation supported by a clone C. Then every k-ary element of* $\mathrm{wClone}(\omega)$ *can be obtained as a weighted sum of translations of ω by tuples of terms from $C^{(k)}$.*

Proposition 1 can be used to decide whether $\mu \in \mathrm{wClone}(\omega)$, where $\mu \in \mathbf{W}_D^{(\ell)}$ and $\omega \in \mathbf{W}_D^{(k)}$ are weighted operations. We define the **translation matrix** of ω to be the matrix A_ω whose columns correspond to the translations of ω by $g_1, \ldots, g_k$ where $g_1, \ldots, g_k \in C^{(\ell)}$. By Proposition 1, $\mu \in \mathrm{wClone}(\omega)$ if and only if we can find a non-negative solution to the system of equations $A_\omega x = \mu$.

Definition 7. *Let $\phi \in \mathbf{\Phi}_D^{(r)}$ and let $\omega \in \mathbf{W}_D^{(k)}$. We say that ω is a* **weighted polymorphism** *of ϕ if, for any $x_1, x_2, \ldots, x_k \in D^r$ such that $\phi(x_i) < \infty$ for $i = 1, \ldots, k$, we have*

$$\sum_{f \in \mathbf{dom}(\omega)} \omega(f)\phi(f(x_1, x_2, \ldots, x_k)) \leq 0\,. \tag{1}$$

If ω is a weighted polymorphism of ϕ we say ϕ is **improved** *by ω.*

Note that, because $a\infty = \infty$ for any value $a \in \mathbb{Q}_+$ (in particular, $0\infty = \infty$), if inequality (1) holds we must have $\phi(f(x_1, \ldots, x_k)) < \infty$, for all $f \in \mathbf{dom}(\omega)$, i.e., each $f \in \mathbf{dom}(\omega)$ is a polymorphism of ϕ [14].

Example 3. Consider the class of submodular set functions [24]. These are precisely the cost functions on $\{0, 1\}$ satisfying

$$\phi(\min(x_1, x_2)) + \phi(\max(x_1, x_2)) - \phi(x) - \phi(y) \leq 0\,.$$

In other words, the set of submodular functions are defined as the set of cost functions on $\{0, 1\}$ with the 2-ary weighted polymorphism

$$\omega(f) = \begin{cases} -1 & \text{if } f \in \{\mathrm{e}_1^{(2)}, \mathrm{e}_2^{(2)}\} \\ +1 & \text{if } f \in \{\min(x_1, x_2), \max(x_1, x_2)\} \end{cases}.$$

This shows that weighted polymorphisms capture an important class of submodular functions, which have been studied within various contexts in computer science [29].

3 Weighted Rosenberg

Rosenberg's Classification Theorem [26], given below, gives certain conditions that minimal clones must satisfy. This has been a major tool in the efforts to identify all tractable maximal constraint languages [6,7] and, furthermore, in efforts to classify all tractable constraint languages [8,10].

For a unary operation we define $f^1 = f$ and $f^i(x) = f(f^{i-1}(x))$. A unary operation f is a **retraction** if $f^2(x) = f(x)$ for all $x \in D$, and a cyclic permutation of prime order if $f^p(x) = x$ for some prime p and all $x \in D$.

An operation f is **idempotent** if $f(x, \ldots, x) = x$ for all $x \in D$. A k-ary, $k \geq 3$, operation f is a **semiprojection** if there is $1 \leq i \leq k$ such that $f(x_1, \ldots, x_k) = \mathrm{e}_i^{(k)} = x_i$ for all $x_1, \ldots, x_k \in D$ such that $x_1, \ldots, x_k$ are not pairwise distinct. A ternary operation f is a **majority** operation (denoted by

Mjrty) if $f(x,x,y) = f(x,y,x) = f(y,x,x) = x$ for all $x, y \in D$. A ternary operation f is a **minority** operation (denoted by Mnrty) if $f(x,x,y) = f(x,y,x) = f(y,x,x) = y$ for all $x, y \in D$. A ternary operation f is a **Pixley** operation if $f(y,y,x) = f(x,y,x) = f(y,x,x) = y$ for all $x, y \in D$ (up to permutations of inputs). We say a k-ary operation f is **sharp** if f is not a projection but the operation obtained by equating any two inputs in f is a projection. The following lemma shows that the only sharp operations of arity $k \geq 4$ are semiprojections.

Lemma 1 (Świerczkowski's Lemma [30]). *Given an operation of arity ≥ 4, if every operation arising from the identification of two variables is a projection, then these projections coincide.*

Świerczkowski's Lemma can be used to prove Rosenberg's Classification Theorem [26], stated below.

Theorem 1 (Rosenberg). *If C is a minimal clone on D, then C must contain an operation f satisfying one of the following conditions:*

1. *f is a retraction or a cyclic permutation of prime order.*
2. *f is binary and idempotent.*
3. *f is a ternary minority operation of the form $f(x,y,z) = x - y + z$, where addition is over some elementary 2-group*[4]*.*
4. *f is a ternary majority operation.*
5. *f is a n-ary semiprojection, $n \geq 3$.*

In this section, we define minimal weighted clones, and give some necessary conditions for a weighted operation to generate a minimal weighted clone. These results can be viewed as an analogue to Rosenberg's Classification Theorem for weighted clones.

Definition 8. *Let W be a weighted clone. We say W is* **minimal** *if every non-zero $\omega \in W$ generates W, i.e., for all non-zero $\omega \in W$, $W = \mathrm{wClone}(\omega)$.*

For a minimal weighted clone W, we say $\omega \in W$ is a **minimal weighted operation** *if ω has smallest arity amongst non-zero elements of W and ω assigns weight -1 to each projection.*

The following lemma shows that every minimal weighted clone is generated by a minimal weighted operation.

Lemma 2. *Let ω be a non-zero weighted operation. There exists some $\omega' \in \mathrm{wClone}(\omega)$ of equal arity which assigns weight -1 to each projection.*

Proof. Suppose ω a is non-zero weighted operation of arity k. Let $Cycle(k)$ denote the set of cyclic permutations of $[k]$. For each permutation $\sigma \in Cycle(k)$, let $\mathbf{t}_\sigma = \left\langle \mathrm{e}^{(k)}_{\sigma(1)}, \ldots, \mathrm{e}^{(k)}_{\sigma(k)} \right\rangle$. Then, the weighted operation $\sum_{\sigma \in Cycle(k)} \omega[\mathbf{t}_\sigma]$ assigns equal weight to each projection. Thus, by a suitable scaling we can obtain a k-ary weighted operation $\omega' \in \mathrm{wClone}(\omega)$ satisfying $\omega'(\mathrm{e}^{(k)}_i) = -1$ for each $i \in [k]$. $\square$

[4] An elementary 2-group is an Abelian group of order 2, i.e. for every element x of the group, $x + x = 0$

We will use the following shorthand for candidate minimal weighted operations (weighted operations which assign weight -1 to each projection):

$$\{(\omega(f), f) : \omega(f) > 0\}.$$

We can now give our classification theorem for minimal weighted operations. The format, and indeed the proof, follow directly from Rosenberg [26], see also [17]. Our result is slightly weaker, because we cannot rule out the possibility of sharp, but non-minimal, operations occurring with negative weight in a minimal weighted operation.

Theorem 2. *The set of operations assigned positive weight by a minimal weighted operation is one of the following four types:*

1. *A set of unary operations.*
2. *A set of binary idempotent operations.*
3. *A set consisting of sharp ternary operations, i.e. majority operations, minority operations, Pixley operations and semiprojections.*
4. *A set of k-ary semiprojections ($k > 3$).*

Proof. Suppose ω is a minimal weighted operation of arity at least two. Then, every f with $\omega(f) > 0$ must be idempotent since otherwise translating by $\langle \mathrm{e}_1, \ldots, \mathrm{e}_1 \rangle$ would yield a non-zero unary weighted operation μ. If $\mathrm{wClone}(\mu) = \mathrm{wClone}(\omega)$, then since ω has bigger arity than μ we get a contradiction with ω being a minimal weighted operation; if $\mathrm{wClone}(\mu) \neq \mathrm{wClone}(\omega)$, then $\mathrm{wClone}(\omega)$ is not a minimal weighted clone.

Next, suppose that ω is a ternary minimal weighted operation. We cannot have any f with $\omega(f) > 0$ for which identifying two variables gives a non-projection operation, since otherwise ω would generate a minimal binary weighted operation. There are precisely 8 types of sharp ternary operations, given in Table 1. The first and last correspond to majority and minority respectively. The second, third and fifth correspond to semiprojections, and the other three correspond to Pixley operations.

Input	1	2	3	4	5	6	7	8
(x,x,y)	x	x	x	x	y	y	y	y
(x,y,x)	x	x	y	y	x	x	y	y
(y,x,x)	x	y	x	y	x	y	x	y

Table 1. Sharp ternary operations

Finally, suppose ω is a minimal weighted operation of arity 4 or greater. Every f with $\omega(f) > 0$ must become a projection when we identify any two variables. Thus, by the Świerczkowski Lemma (Lemma 1) each such f must be a semiprojection. □

4 Simple Weighted Clones

In classical clone theory, every minimal clone is generated by a single operation. Rosenberg's classification of minimal operations [26] gives unary operations (retractions and cyclic permutations of prime orders), binary idempotent operations, majority operations, minority operations, and semiprojections.

Definition 9. *For any k-ary operation f we define the* **canonical weighted operation** *of f, ω_f, to be $\{(k, f)\}$.*

In other words, ω_f assigns weight k to f, and weight -1 to each projection.

In the rest of this section we prove that for some minimal operations f, the canonical weighted operation ω_f is a minimal weighted operation. In particular, we prove this for retractions, certain binary operations, majority operations, and minority operations.

Theorem 3. *If f is a retraction, then ω_f is a minimal weighted operation.*

Proof. Let f be a minimal unary operation which is a retraction; i.e. $f(f(x)) = f(x)$ for all $x \in D$. Let ω_f be the canonical weighted operation of f. Since f is a retraction, it is the only non-trivial operation in $\mathrm{Clone}(f)$. Hence, given any $\mu \in \mathrm{wClone}^{(k)}(\omega_f)$, translating by $\left\langle \mathrm{e}_1^{(k)}, \ldots, \mathrm{e}_k^{(k)} \right\rangle$ and applying a suitable scaling yields ω_f. □

Theorem 4. *If f is a binary operation, then ω_f is minimal whenever f is a semilattice operation or a conservative commutative operation.*

Proof. Whenever f is a conservative commutative operation or a semilattice operation, we have that f is the only non-trivial binary operation in $\mathrm{Clone}(f)$. Thus, given any $\mu \in \mathrm{Clone}^{(k)}(\omega_f)$ we can find some tuple of binary projections $\mathbf{t}$ satisfying $g[\mathbf{t}] = f(x_1, x_2)$, for some g with $\mu(g) > 0$. That is, $\mu' = \mu[\mathbf{t}]$ is a binary weighted operation with $\mu'(f) > 0$. Finally, since f is commutative, the weighted operation obtained from μ' by Lemma 2 must be equal to ω_f. □

Theorem 5. *If f is a majority operation, then ω_f is a minimal weighted operation.*

Proof. It is well known that any ternary operation generated by f is a majority operation since f is a majority operation. (This can be proved by induction on the number of occurrences of f.) We want to show that ω_f is minimal; that is, given $\mu \in \mathrm{wClone}(\omega_f)$, we need to show that $\omega_f \in \mathrm{wClone}(\mu)$.

Let μ be a k-ary weighted operation from $\mathrm{wClone}(\omega_f)$ such that $\mu(g) > 0$ for some non-projection g, where $g \in \mathrm{Clone}^{(k)}(f)$. From the argument above, there exists some k-tuple of ternary projections, $\mathbf{t}$, such that $g[\mathbf{t}]$ is a majority operation. Let $\mu' = \mu[\mathbf{t}]$. If $\mu' = c\omega_f$ for some $c > 0$ then we are done. Otherwise, by Lemma 2, there is ternary $\mu' \in \mathrm{wClone}(\mu)$ such that μ' assigns weight -1 to projections and positive weight to some (possibly different) majority operations $g_1, \ldots, g_k \in \mathrm{Clone}^{(3)}(f)$.

Translating μ' by $\langle x_j, x_j, f\rangle$, for $j \in [3]$, gives the weighted operation $2\omega_{j,f}$, where

$$\omega_{j,f}(g) = \begin{cases} -1 & g = \mathrm{e}_j \\ +1 & g = f \\ 0 & \text{otherwise} \end{cases} . \tag{2}$$

Since $\omega_f = \omega_{1,f} + \omega_{2,f} + \omega_{3,f}$, we have proved that $\omega_f \in \mathrm{wClone}(\mu)$. □

Theorem 6. *If f is a minimal minority operation, then ω_f is a minimal weighted operation.*

Proof. Recall that a minority operation $f : D^3 \to D$ is minimal if and only if $f(x, y, z) = x - y + z$, where addition is taken over an elementary 2-group $\langle D, +\rangle$. An elementary 2-group $\langle D, +\rangle$ satisfies $2x = 0$ for all $x \in D$. Thus, we can conclude that f is the only ternary operation in $\mathrm{Clone}(f)$. Now, given any $\mu \in \mathrm{wClone}^{(k)}(\omega_f)$, we can find some k-tuple of ternary projections $\mathbf{t}$ such that $\mu[\mathbf{t}](f) > 0$. Then, using Lemma 2, we can obtain a ternary weighted operation $\mu' \in \mathrm{wClone}(\mu)$ which satisfies $\mu'(\mathrm{e}_i) = 1$ for $i = 1, 2, 3$. Since f is the only ternary operation in $\mathrm{Clone}(f)$ then, necessarily, $\mu' = \omega_f$. □

Due to space constraints, we only state the following result:

Proposition 2. *Let f be the ternary semiprojection on $D = \{0, 1, 2\}$ which returns 0 on every input with all values distinct, and the value of the first input otherwise. The weighted clone $\mathrm{wClone}(\omega_f)$ is not minimal.*

Proposition 2 tells us that not all minimal operations have canonical minimal weighted operations. It is known that the constraint languages preserved by semiprojections are not tractable, so the weighted clones supported by semiprojection clones are of less interest to us.

An operation f is **tractable** if the set of cost functions invariant under f, denote by $\mathrm{Inv}(f)$, is a tractable valued constraint language; see [14] for more details. Having proved minimality of weighted clones corresponding to well-known tractable operations, we finish this section with a conjecture.

Conjecture 1. If f is a minimal tractable operation, then ω_f is a minimal weighted operation.

5 Boolean Classification

In this section, we consider minimal weighted clones on Boolean domain $D = \{0, 1\}$. Since there are no semiprojections on a Boolean domain, we only need to consider the first three cases of Theorem 2. Moreover, for the third case, we need only consider weighted operations assigning negative weight to Mnrty and Mjrty. Post [25] has classified the minimal clones on a Boolean domain.

Theorem 7. *Every minimal clone on a Boolean domain is generated by one of the following operations:*

1. $f_0(x) = 0$
2. $f_1(x) = 1$
3. $f(x) = 1 - x$
4. $\min(x_1, x_2)$ *returns the minimum of the two arguments*
5. $\max(x_1, x_2)$ *returns the maximum of the two arguments*
6. $\mathrm{Mnrty}(x_1, x_2, x_3)$ *returns the minority of the three arguments*
7. $\mathrm{Mjrty}(x_1, x_2, x_3)$ *returns the majority of the three arguments*

First, we show that the canonical weighted operations corresponding to the minimal operations given in Theorem 7 are minimal.

Theorem 8. *For each minimal Boolean operation f, the weighted operation ω_f is minimal.*

Proof. Let $f(x) = 1-x$. Notice that $f^2(x) = x$; that is, f is a cyclic permutation of order 2. Therefore, the only non-trivial unary operation in $\mathrm{Clone}(f)$ is f. Moreover, for any $k > 1$, the only non-trivial operations in $\mathrm{Clone}^{(k)}(f)$ are of the form $g(x) = 1 - x_i$ for some $i \in [k]$. Translating an operation of this form by the k-tuple of unary projections will yield f. Thus, given any non-zero $\mu \in \mathrm{wClone}^{(k)}(\omega_f)$, we can translate by the k-tuple of unary projections and apply a suitable scaling to obtain ω_f. Hence, w_f is minimal. All other cases follow from Theorems 3, 4, 5, and 6. $\square$

Next, we show that there are precisely two other minimal weighted operations on a Boolean domain.

Theorem 9. *On a Boolean domain, there are precisely two minimal weighted operations other than the* 7 *canonical weighted operations arising from the minimal operations. These are the binary weighted operations* $\{(1, \min), (1, \max)\}$ *and* $\{(1, \mathrm{Mnrty}), (2, \mathrm{Mjrty})\}$*.*

Proof. We first consider the binary case. Every binary minimal operation other than $\omega_{\min}$ and $\omega_{\max}$ must be of the form $\omega_a = \{(a, \min), (2-a, \max)\}$ $(0 < a < 2)$. We will show that ω_a is minimal if and only if $a = 1$.

First, suppose $a = 1$. Let $\omega = \omega_1$. It is easy to check that the only non-zero translation is $\omega[\mathrm{e}_1, \mathrm{e}_2]$. Thus, by Proposition 1, every non-zero weighted operation in $\mathrm{wClone}^{(2)}(\omega)$ is equal to $c\omega$, for some $c > 0$.

There is precisely one sharp operation of arity ≥ 3 in $\mathrm{Clone}(\min, \max)$: the majority operation Mjrty. Since $\{(3, \mathrm{Mjrty})\} \notin \mathrm{wClone}(\omega)$ (we can check this using Proposition 1), it follows that any non-zero $\mu \in \mathrm{wClone}^{(k)}(\omega)$ must assign weight to an operation of the form $\min(x_i, x_j)$ or $\max(x_i, x_j)$, or a non-sharp operation of arity k, for any $k > 2$. We can translate any such μ by a k-tuple of binary projections to obtain some non-zero $\mu' \in \mathrm{wClone}^{(2)}(\mu)$. Since μ' must necessarily be contained in $\mathrm{wClone}(\omega)$, and since every binary weighted operation in $\mathrm{wClone}(\omega)$ is equal to $c\omega$, for some constant $c > 0$, it follows that $\omega \in \mathrm{wClone}(\mu)$. Hence, ω is minimal.

Now, suppose $a < 1$ (the other case is symmetric). Consider the weighted operations $\mu_i = \omega_a + \frac{a}{1-a}\omega_a[\mathrm{e}_i, \min]$ $(i = 1, 2)$, which by Proposition 1 are contained in $\mathrm{wClone}(\omega_a)$. Since $\min(x, \min(x, y)) = \min(x, y)$, we have that min is assigned weight $a-1$ in $\omega_a[\mathrm{e}_i, \min]$, and hence 0 in μ_i. To be precise, μ_i is the weighted operation which assigns weight $a-1$ to e_i, -1 to $\mathrm{e}_{\bar{i}}$ ($\bar{i} \in \{1, 2\} \backslash \{i\}$), and $2-a$ to max. Thus, by adding μ_1 and μ_2 and applying a suitable scaling, we can obtain the weighted operation $\{(2, \max)\}$. Since $\{(2, \max)\}$ generates a minimal clone which does not contain ω_a, we can conclude that ω_a is not minimal.

We now move on to the ternary case. Suppose ω is a ternary weighted operation and $\omega \notin \mathrm{wClone}(\omega_f)$ for $f \in \{\mathrm{Mnrty}, \mathrm{Mjrty}\}$. From Theorem 2 and the fact that there are no Boolean semiprojections, ω can only assign positive weight to Mjrty, Mnrty and the three Boolean Pixley operations f_1, f_2 and f_3 (corresponding to the fourth, sixth and seventh columns of Table 1). We first show that we can restrict our attention to weighted operations assigning weight 0 to all Pixley operations.

Let ω be a ternary weighted operation which assigns positive weight to some Pixley operations. Composing $\langle f_1, f_2, f_3 \rangle$ with the tuples of projections $\langle \mathrm{e}_2, \mathrm{e}_3, \mathrm{e}_1 \rangle$ and $\langle \mathrm{e}_3, \mathrm{e}_1, \mathrm{e}_2 \rangle$ yields $\langle f_2, f_3, f_1 \rangle$ and $\langle f_3, f_1, f_2 \rangle$ respectively. Thus, the weighted operation $\frac{1}{3}\omega + \frac{1}{3}\omega[e_2, e_3, e_1] + \frac{1}{3}\omega[e_3, e_1, e_2]$ assigns equal weight to each Pixley operation. Hence, from here on we assume we are working with a weighted operation ω which assigns equal weight to each Pixley operation, as well as assigning weight -1 to each projection (see Lemma 2).

Suppose each Pixley operation is assigned weight $a < 1$ by ω, so at least one of Mjrty and Mnrty is assigned positive weight. We observe that $f_i(f_1, f_2, f_3) = \mathrm{e}_i$ for each $i = 1, 2, 3$. Moreover, $\mathrm{Mjrty}(f_1, f_2, f_3) = \mathrm{Mnrty}$ and $\mathrm{Mnrty}(f_1, f_2, f_3) = \mathrm{Mjrty}$. Thus, the weighted operation $\omega + a\omega[f_1, f_2, f_3]$ is non-zero and assigns weight 0 to each Pixley operation.

Next, suppose each Pixley operation is assigned weight 1 by ω. Let $\mu_1 = \omega[\mathrm{e}_1, \mathrm{e}_2, f_1]$. Since $f_1(\mathrm{e}_1, \mathrm{e}_2, f_1) = \mathrm{Mjrty}$, $f_2(\mathrm{e}_1, \mathrm{e}_2, f_1) = e_1$ and $f_3(\mathrm{e}_1, \mathrm{e}_2, f_1) = e_2$, we have that μ_1 assigns weight -1 to f_1, $+1$ to Mjrty, and is 0 everywhere else. For $i = 2, 3$, we can obtain μ_i, which assigns weight -1 to f_i and $+1$ to Mjrty, by a similar translation. Then the weighted operation obtained as $\omega + \mu_1 + \mu_2 + \mu_3$ will be equal to $\{(3, \mathrm{Mjrty})\}$.

Thus, the weighted clone generated by any minimal ternary weighted operation will contain a non-zero ternary weighted operation assigning weight 0 to all Pixley operations. Hence, when searching for minimal ternary weighted operations other than ω_{Mnrty} and ω_{Mjrty}, we can restrict our attention to weighted operations of the form $\omega_a = \{(a, \mathrm{Mnrty}), (3-a, \mathrm{Mjrty})\}$ $(0 < a < 3)$. We will now show that ω_a is minimal if and only if $a = 1$.

Let $\omega = \omega_1$. Using Proposition 1, we can check that every ternary weighted operation in $\mathrm{wClone}(\omega)$ which assigns positive weight to Mnrty and Mjrty only is of the form $c\omega$ for some $c > 0$. Since there are no semi-projections, we can translate any non-zero $\mu \in \mathrm{wClone}^{(k)}(\omega)$ by a k-tuple of ternary projections to obtain ternary non-zero $\mu' \in \mathrm{wClone}(\omega)$. We have shown that we can obtain some non-zero $\mu'' \in \mathrm{wClone}(\mu')$ which assigns positive weight to Mnrty and Mjrty only. Since μ'' must be in $\mathrm{wClone}(\omega)$, it follows that $\mu'' = c\omega$, for some $c > 0$, so we can obtain ω by scaling. Hence, ω is a minimal weighted operation.

Suppose $a < 1$. Let $\mu_i = \omega_a + \frac{a}{1-a}\omega_a[\mathrm{e}_i, \mathrm{e}_i, \mathrm{Mnrty}]$ $(i \in \{1,2,3\})$. It is easy to check that $\mu_i(\mathrm{e}_i) = -1+a$, $\mu_i(\mathrm{e}_j) = -1$ $(j \neq i)$, $\mu_i(\mathrm{Mjrty}) = 3-a$, and $\mu_i(f) = 0$ everywhere else. Then, as in the binary case, we can obtain $\{(3, \mathrm{Mjrty})\}$ by adding μ_1, μ_2, and μ_3 and applying a suitable scaling. Similarly, if $a > 1$ we can show $\{(3, \mathrm{Mnrty})\} \in \mathrm{wClone}(\omega_a)$. In both cases, we have found non-zero $\mu \in \mathrm{wClone}(\omega_a)$ such that $\omega_a \notin \mathrm{wClone}(\mu)$, so ω_a cannot be minimal. □

We remark that the proof of maximality of $\omega_{\langle \min,\max\rangle}$ in Theorem 9 actually proves a stronger result: minimality of $\omega_{\langle \min,\max\rangle}$ over arbitrary distributive lattices with min and max being the lattice meet and join operations.

6 Conclusions

We have studied minimal weighted clones using the algebraic theory for valued constraint languages developed by Cohen et al. [14]. Thus we have shown that the general theory from [14] can be used to answer interesting questions on the complexity of valued constraint languages.

We have shown an analogue of Rosenberg's classification of minimal clones for weighted clones. Furthermore, we have shown minimality of several weighted clones whose support clone is generated by a single minimal operation. On the other hand, we have demonstrated that this is not true in general: there are minimal operations which give rise to non-minimal weighted clones. We have conjectured that minimal *tractable* operations give rise to minimal weighted clones. Finally, we have classified all Boolean weighted clones. Consequently, we have been able to determine all maximal Boolean valued constraint languages, using proofs based on the algebraic characterisation of [11,14]. This has been originally proved in [13] using gadgets.

We believe that the techniques presented in this paper will be useful in identifying new tractable valued constraint languages and proving maximality of valued constraint languages.

References

1. Barto, L., Kozik, M., Maróti, M., Niven, T.: CSP dichotomy for special triads. Proceedings of the American Mathematical Society 137(9), 2921–2934 (2009)
2. Barto, L., Kozik, M., Niven, T.: The CSP dichotomy holds for digraphs with no sources and no sinks SIAM Journal on Computing 38(5), 1782–1802 (2009)
3. Barto, L.: The dichotomy for conservative constraint satisfaction problems revisited. In: Proc. of LICS'11 (2011)
4. Barto, L., Kozik, M.: Constraint Satisfaction Problems of Bounded Width. In: Proc. of FOCS'09 461–471 (2009)
5. Berman, J., Idziak, P., Marković, P., McKenzie, R., Valeriote, M., Willard, R.: Varieties with few subalgebras of powers. Trans. of AMS 362(3), 1445–1473 (2010)
6. Bulatov, A., Krokhin, A., Jeavons, P.: The complexity of maximal constraint languages. In: Proc. of STOC'01 667–674 (2001)

7. Bulatov, A.: A Graph of a Relational Structure and Constraint Satisfaction Problems. In: Proc. of LICS'04 448–457 (2004)
8. Bulatov, A.: A dichotomy theorem for constraint satisfaction problems on a 3-element set. Journal of the ACM 53(1), 66–120 (2006)
9. Bulatov, A., Krokhin, A., Jeavons, P.: Classifying the Complexity of Constraints using Finite Algebras. SIAM Journal on Computing 34(3), 720–742 (2005)
10. Bulatov, A.A.: Tractable Conservative Constraint Satisfaction Problems. In: Proc. of LICS'03 321–330 (2003)
11. Cohen, D.A., Cooper, M.C., Jeavons, P.G.: An Algebraic Characterisation of Complexity for Valued Constraints. In: Proc. of CP'06, pp 107–121 (2006)
12. Cohen, D.A., Cooper, M.C., Jeavons, P.G.: Generalising submodularity and Horn clauses: Tractable optimization problems defined by tournament pair multimorphisms. Theoretical Computer Science 401(1-3), 36–51 (2008)
13. Cohen, D.A., Cooper, M.C., Jeavons, P.G., Krokhin, A.A.: The Complexity of Soft Constraint Satisfaction. Artificial Intelligence 170(11), 983–1016 (2006)
14. Cohen, D., Creed, P., Jeavons, P., Živný, S.: An algebraic theory of complexity for valued constraints: Establishing a Galois connection. Proc. of MFCS'11 (2011).
15. Creignou, N.: A dichotomy theorem for maximum generalized satisfiability problems. Journal of Computer and System Sciences 51(3), 511–522 (1995)
16. Creignou, N., Khanna, S., Sudan, M.: Complexity Classification of Boolean Constraint Satisfaction Problems, SIAM Monographs on Discrete Mathematics and Applications, vol. 7. SIAM (2001)
17. Csákány, B.: Minimal clones – a minicourse. Algebra Universalis 54(1), 73–89 (2005)
18. Denecke, K., Wismath, S.: Universal Algebra and Applications in Theoretical Computer Science. Chapman and Hall/CRC Press (2002)
19. Feder, T., Vardi, M.: 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)
20. Jeavons, P.: On the Algebraic Structure of Combinatorial Problems. Theoretical Computer Science 200(1-2), 185–204 (1998)
21. Jeavons, P., Cohen, D., Gyssens, M.: Closure Properties of Constraints. Journal of the ACM 44(4), 527–548 (1997)
22. Kolmogorov, V., Živný, S.: The complexity of conservative finite-valued CSPs. Technical repport arXiv:1008.1555 (August 2010)
23. Kolmogorov, V., Živný, S.: The complexity of conservative valued CSPs. Submitted for publication (2011)
24. Nemhauser, G., Wolsey, L.: Integer and Combinatorial Optimization. (1988)
25. Post, E.: The two-valued iterative systems of mathematical logic, Annals of Mathematical Studies, vol. 5. Princeton University Press (1941)
26. Rosenberg, I.: Minimal Clones I: the five types. In: Lectures in Universal Algebra (Proc. Conf. Szeged 1983). Colloq. Math. Soc. Janos Bolyai, vol. 43, pp. 405–427. North-Holland (1986)
27. Schaefer, T.: The Complexity of Satisfiability Problems. In: Proc. of STOC'78 216–226 (1978)
28. Schrijver, A.: Theory of linear and integer programming. (1986)
29. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. (2003)
30. Świerczkowski, S.: Algebras which are independently generated by every n elements. Fundamenta Mathematicae 49, 93–104 (1960),