A Galois Connection for Valued Constraint Languages of Infinite Size*

Peter Fulla[1] and Stanislav Živný[1]

Department of Computer Science, University of Oxford, UK
{peter.fulla,standa.zivny}@cs.ox.ac.uk

Abstract. A Galois connection between clones and relational clones on a fixed finite domain is one of the cornerstones of the so-called algebraic approach to the computational complexity of non-uniform Constraint Satisfaction Problems (CSPs). Cohen et al. established a Galois connection between *finitely-generated* weighted clones and *finitely-generated* weighted relational clones [SICOMP'13], and asked whether this connection holds in general. We answer this question in the affirmative for weighted (relational) clones with *real* weights and show that the complexity of the corresponding Valued CSPs is preserved.

1 Introduction

The constraint satisfaction problem (CSP) is a general framework capturing decision problems arising in many contexts of computer science [13]. The CSP is NP-hard in general but there has been much success in finding tractable fragments of the CSP by restricting the types of relations allowed in the constraints. A set of allowed relations has been called a *constraint language* [11]. 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 [11]; 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 two-element domains [19], three-element domains [5], for conservative (containing all unary relations) constraint languages [7], for maximal constraint languages [8, 4], for graphs (corresponding to languages containing a single binary symmetric relation) [12], and for digraphs without sources and sinks (corresponding to languages containing a single binary relations without sources and sinks) [2]. The most successful approach to classifying the complexity of constraint languages has been the algebraic approach [15, 6, 1].

The *valued* constraint satisfaction problem (VCSP) is a general framework that captures not only feasibility problems but also optimisation problems [10, 14]. A VCSP instance represents each constraint by a *weighted relation*, which

* The authors were supported by a Royal Society Research Grant. Stanislav Živný was supported by a Royal Society University Research Fellowship.

is a $\overline{\mathbb{Q}}$-valued function where $\overline{\mathbb{Q}} = \mathbb{Q} \cup \{\infty\}$, and the goal is to find a labelling of variables minimising the sum of the values assigned by the constraints to that labelling. Tractable fragments of the VCSP have been identified by restricting the types of allowed weighted relations that can be used to define the valued constraints. A set of allowed weighted relations has been called a *valued constraint language* [10]. Classifying the complexity of *all* valued constraint languages is a challenging task as it includes as a special case the classification of $\{0, \infty\}$-valued languages (i.e. constraint languages); this would answer the conjecture of Feder and Vardi [11], which asserts that every constraint language is either tractable or NP-hard, and its algebraic refinement, which specifies the precise boundary between tractable and NP-hard languages [6]. However, several nontrivial results are known, see [14] for a recent survey. Dichotomy theorems, which classify each possible valued constraint language as either tractable or NP-hard, have been established for valued constraint languages over two-element domains [10], for conservative (containing all $\{0, 1\}$-valued unary cost functions) valued constraint languages [17], and also for finite-valued (all weighted relations are $\mathbb{Q}$-valued) constraint languages [22]. Moreover, the power of the basic linear programming relaxation for valued constraint languages has been characterised [21, 16].

Cohen et al. have introduced an algebraic theory of weighted clones [9], further extended in [18], for classifying the computational complexity of valued constraint languages. This theory establishes a one-to-one correspondence between valued constraint languages closed under expressibility (which does not change the complexity of the associated class of optimisation problems), called weighted relational clones, and weighted clones [9]. This is an extension of (a part of) the algebraic approach to CSPs which relies on a one-to-one correspondence between constraint languages closed under pp-definability (which does not change the complexity of the associated class of decision problems), called relational clones, and clones [6], thus making it possible to use deep results from universal algebra. In fact, the recent progress on the power of the basic linear programming relaxation [16] and the classification of finite-valued constraint languages [22], as well as results on special cases of Valued CSPs such as Min-Sol-Hom [23], rely on the work of Cohen et al [9].

Contributions

The Galois connection between weighted clones and weighted relational clones established in [9] was proved only for weighted (relational) clones generated by a set of a *finite* size. The authors asked whether such a correspondence holds also for weighted (relational) clones in general. In this paper we answer this question in the affirmative.

Firstly, we show that the Galois connection from [9] (using only rational weights) does *not* work for general weighted (relational) clones. Secondly, we alter the definition of weighted (relational) clones and establish a new Galois connection that holds even when the generating set has an infinite size. We allow weighted relations and weightings to assign real weights instead of rational, require weighted relational clones to be closed under operator Opt, and prove that these changes preserve tractability of a constraint language.

Including the Opt operator in the definition of weighted relational clones simplifies the structure of the space of all weighted clones, and guarantees that every non-projection polymorphism of a weighted relational clone Γ is assigned a positive weight by some weighted polymorphism of Γ.

The proof of the Galois connection in [9] relies on results on linear programming duality; we used their generalisation from the theory of convex optimisation in order to establish the connection even for infinite sets.

2 Background

2.1 Valued CSPs

Throughout the paper, let D be a fixed finite set of size at least two.

Definition 1. *An m-ary* relation[1] *over D is any mapping $\phi : D^m \to \{c, \infty\}$ for some $c \in \mathbb{Q}$. We denote by $\mathbf{R}_D^{(m)}$ the set of all m-ary relations and let $\mathbf{R}_D = \bigcup_{m \geq 1} \mathbf{R}_D^{(m)}$.*

Given an m-tuple $\mathbf{x} \in D^m$, we denote its ith entry by $\mathbf{x}[i]$ for $1 \leq i \leq m$.

Let $\overline{\mathbb{Q}} = \mathbb{Q} \cup \{\infty\}$ denote the set of rational numbers with (positive) infinity.

Definition 2. *An m-ary* weighted relation *over D is any mapping $\gamma : D^m \to \overline{\mathbb{Q}}$. We denote by $\mathbf{\Phi}_D^{(m)}$ the set of all m-ary weighted relations and let $\mathbf{\Phi}_D = \bigcup_{m \geq 1} \mathbf{\Phi}_D^{(m)}$.*

From Definition 2 we have that relations are a special type of weighted relations.

Example 1. An important example of a (weighted) relation is the binary equality $\phi_=$ on D defined by $\phi_=(x, y) = 0$ if $x = y$ and $\phi_=(x, y) = \infty$ if $x \neq y$.

Another example of a relation is the unary empty relation $\phi_\emptyset$ defined on D by $\phi_\emptyset(x) = \infty$ for all $x \in D$.

For any m-ary weighted relation $\gamma \in \mathbf{\Phi}_D^{(m)}$, we denote by $\mathrm{Feas}(\gamma) = \{\mathbf{x} \in D^m | \gamma(\mathbf{x}) < \infty\} \in \mathbf{R}_D^{(m)}$ the underlying *feasibility relation*, and by $\mathrm{Opt}(\gamma) = \{\mathbf{x} \in \mathrm{Feas}(\gamma) \mid \gamma(\mathbf{x}) \leq \gamma(\mathbf{y})$ for every $\mathbf{y} \in D^m\} \in \mathbf{R}_D^{(m)}$ the relation of minimal-value tuples.

[1] An m-ary relation over D is commonly defined as a subset of D^m. Note that Definition 1 is equivalent to the standard definition as any mapping ϕ can be seen as set $R = \{\mathbf{x} \in D^m \mid \phi(\mathbf{x}) < \infty\}$, and any set $R \subseteq D^m$ can be represented by mapping ϕ such that $\phi(\mathbf{x}) = 0$ when $\mathbf{x} \in R$ and $\phi(\mathbf{x}) = \infty$ otherwise. Consequently, we shall use both definitions interchangeably.

Definition 3. *Let $V = \{x_1, \ldots, x_n\}$ be a set of variables. A* valued constraint *over V is an expression of the form $\gamma(\mathbf{x})$ where $\gamma \in \mathbf{\Phi}_D^{(m)}$ and $\mathbf{x} \in V^m$. The number m is called the* arity *of the constraint, the weighted relation γ is called the* constraint weighted relation, *and the tuple $\mathbf{x}$ the* scope *of the constraint.*

We call D the *domain*, the elements of D *labels* (for variables), and say that the weighted relation in $\mathbf{\Phi}_D$ take *values* or *weights*.

Definition 4. *An instance of the* valued constraint satisfaction problem, *VCSP, is specified by a finite set $V = \{x_1, \ldots, x_n\}$ of variables, a finite set D of labels, and an* objective function *I expressed as follows:*

$$I(x_1, \ldots, x_n) = \sum_{i=1}^{q} \gamma_i(\mathbf{x}_i)\,, \tag{1}$$

where each $\gamma_i(\mathbf{x}_i)$, $1 \leq i \leq q$, is a valued constraint over V. Each constraint can appear multiple times in I.

The goal is to find an assignment *(or a* labelling*) of labels to the variables that minimises I.*

CSPs are a special case of VCSPs using only (unweighted) relations with the goal to determine the existence of a feasible assignment.

Definition 5. *Any set $\Gamma \subseteq \mathbf{\Phi}_D$ is called a* (valued) constraint language *over D, or simply a* language. *We will denote by* $\mathrm{VCSP}(\Gamma)$ *the class of all VCSP instances in which the constraint weighted relations are all contained in Γ.*

Definition 6. *A constraint language Γ is called* tractable *if* $\mathrm{VCSP}(\Gamma')$ *can be solved (to optimality) in polynomial time for every finite subset $\Gamma' \subseteq \Gamma$, and Γ is called* intractable *if* $\mathrm{VCSP}(\Gamma')$ *is NP-hard for some finite $\Gamma' \subseteq \Gamma$.*

We are interested in the computational complexity of various constraint languages, see [14] for a recent survey on this topic.

2.2 Weighted relational clones

Definition 7. *A weighted relation γ of arity r can be obtained by* addition *from the weighted relation γ_1 of arity s and the weighted relation γ_2 of arity t if γ satisfies the identity*

$$\gamma(x_1, \ldots, x_r) = \gamma_1(y_1, \ldots, y_s) + \gamma_2(z_1, \ldots, z_t) \tag{2}$$

for some (fixed) choice of $y_1, \ldots, y_s$ and $z_1, \ldots, z_t$ from amongst the $x_1, \ldots, x_r$.

Definition 8. *A weighted relation γ of arity r can be obtained by* minimisation *from the weighted relation γ' of arity $r + s$ if γ satisfies the identity*

$$\gamma(x_1, \ldots, x_r) = \min_{y_1 \in D, \ldots, y_s \in D} \gamma'(x_1, \ldots, x_r, y_1, \ldots, y_s)\,. \tag{3}$$

Definition 9. *A constraint language* $\Gamma \subseteq \mathbf{\Phi}_D$ *is called a* weighted relational clone *if it contains the binary equality relation* $\phi_=$ *and the unary empty relation* $\phi_\emptyset$*,*[2] *and is closed under addition, minimisation, scaling by non-negative rational constants, and addition of rational constants.*

For any Γ*, we define* $\mathrm{wRelClone}(\Gamma)$ *to be the smallest weighted relational clone containing* Γ*.*

Note that for any weighted relational clone Γ, if $\gamma \in \Gamma$ then $\mathrm{Feas}(\gamma) \in \Gamma$ as $\mathrm{Feas}(\gamma) = 0\gamma$ (we define $0 \cdot \infty = \infty$).

Definition 10. *Let* $\Gamma \subseteq \mathbf{\Phi}_D$ *be a constraint language,* $I \in \mathrm{VCSP}(\Gamma)$ *an instance with variables* V*, and* $L = (v_1, \ldots, v_r)$ *a list of variables from* V*. The* projection *of* I *onto* L*, denoted* $\pi_L(I)$*, is the* r*-ary weighted relation on* D *defined as*

$$\pi_L(I)(x_1, \ldots, x_r) = \min_{\{s:V \to D \mid (s(v_1),\ldots,s(v_r))=(x_1,\ldots,x_r)\}} I(s)\,. \tag{4}$$

We say that a weighted relation γ *is* expressible *over a constraint language* Γ *if* $\gamma = \pi_L(I)$ *for some* $I \in \mathrm{VCSP}(\Gamma)$ *and list of variables* L*. We call the pair* (I, L) *a* gadget *for expressing* γ *over* Γ*.*

The list of variables L in a gadget may contain repeated entries. The minimum over an empty set is ∞.

Example 2. For any $\Gamma \subseteq \mathbf{\Phi}_D$, we can express the binary equality relation $\phi_=$ on D over language Γ using the following gadget. Let $I \in \mathrm{VCSP}(\Gamma)$ be the instance with a single variable v and no constraints, and let $L = (v, v)$. Then, by Definition 10, $\pi_L(I) = \phi_=$.

We may equivalently define a weighted relational clone as a set $\Gamma \subseteq \mathbf{\Phi}_D$ that contains the unary empty relation $\phi_\emptyset$ and is closed under expressibility, scaling by non-negative rational constants, and addition of rational constants [9, Proposition 4.5].

The following result has been shown in [9].

Theorem 1. *A constraint language* Γ *is tractable if and only if* $\mathrm{wRelClone}(\Gamma)$ *is tractable, and* Γ *is intractable if and only if* $\mathrm{wRelClone}(\Gamma)$ *is intractable.*

Consequently, when trying to identify tractable constraint languages, it is sufficient to consider only weighted relational clones.

2.3 Weighted clones

Any mapping $f : D^k \to D$ is called a k-ary *operation.* We will apply a k-ary operation f to k m-tuples $\mathbf{x}_1, \ldots, \mathbf{x}_k \in D^m$ coordinatewise, that is,

$$f(\mathbf{x}_1, \ldots, \mathbf{x}_k) = (f(\mathbf{x}_1[1], \ldots, \mathbf{x}_k[1]), \ldots, f(\mathbf{x}_1[m], \ldots, \mathbf{x}_k[m])) \in D^m\,. \tag{5}$$

[2] Although the definition in [9] does not require inclusion of $\phi_\emptyset$, the proofs there implicitly assume its presence in any weighted relational clone.

Definition 11. *Let γ be an m-ary weighted relation on D and let f be a k-ary operation on D. Then f is a* polymorphism *of γ if, for any $X = (\mathbf{x}_1, \dots, \mathbf{x}_k) \in (\mathrm{Feas}(\gamma))^k$, we have that $f(X) = f(\mathbf{x}_1, \dots, \mathbf{x}_k) \in \mathrm{Feas}(\gamma)$.*

For any constraint language Γ over a set D, we denote by $\mathrm{Pol}(\Gamma)$ the set of all operations on D which are polymorphisms of all $\gamma \in \Gamma$. We write $\mathrm{Pol}(\gamma)$ for $\mathrm{Pol}(\{\gamma\})$.

A k-ary *projection* is an operation of the form $e_i^{(k)}(x_1, \dots, x_k) = x_i$ for some $1 \leq i \leq k$. Projections are (trivial) polymorphisms of all constraint languages.

Definition 12. *The* superposition *of a k-ary operation $f : D^k \to D$ with k ℓ-ary operations $g_i : D^\ell \to D$ for $1 \leq i \leq k$ is the ℓ-ary function $f[g_1, \dots, g_k] : D^\ell \to D$ defined by*

$$f[g_1, \dots, g_k](x_1, \dots, x_\ell) = f(g_1(x_1, \dots, x_\ell), \dots, g_k(x_1, \dots, x_\ell))\,. \tag{6}$$

Definition 13. *A* clone *of operations, C, is a set of operations on D that contains all projections and is closed under superposition. The k-ary operations in a clone C will be denoted by $C^{(k)}$.*

Example 3. For any D, let $\mathbf{J}_D$ be the set of all projections on D. By Definition 13, $\mathbf{J}_D$ is a clone.

It is well known that $\mathrm{Pol}(\Gamma)$ is a clone for all constraint languages Γ.

Definition 14. *A k-ary* weighting *of a clone C is a function $\omega : C^{(k)} \to \mathbb{Q}$ such that $\omega(f) < 0$ only if f is a projection and*

$$\sum_{f \in C^{(k)}} \omega(f) \;=\; 0\,. \tag{7}$$

We will call a function $\omega : C^{(k)} \to \mathbb{Q}$ that satisfies Equation (7) but assigns a negative weight to some operation $f \notin \mathbf{J}_D^{(k)}$ an improper weighting*. In order to emphasise the distinction we may also call a weighting a* proper weighting.

Definition 15. *For any clone C, a k-ary weighting ω of C, and $g_1, \dots, g_k \in C^{(\ell)}$, the* superposition *of ω and $g_1, \dots, g_k$, is the function $\omega[g_1, \dots, g_k] : C^{(\ell)} \to \mathbb{Q}$ defined by*

$$\omega[g_1, \dots, g_k](f') = \sum_{\{f \in C^{(k)} \mid f[g_1, \dots, g_k] = f'\}} \omega(f)\,. \tag{8}$$

If the result of a superposition is a proper weighting (that is, negative weights are only assigned to projections), then that superposition will be called a proper superposition.

Definition 16. *A* weighted clone*, Ω, is a non-empty set of weightings of some fixed clone C, called the* support clone *of Ω, which is closed under scaling by non-negative rational constants, addition of weightings of equal arity, and proper superposition with operations from C.*

We now link weightings and weighted relations by the concept of weighted polymorphism, which will allow us to establish a useful correspondence between weighted clones and weighted relational clones.

Definition 17. *Let γ be an m-ary weighted relation on D and let ω be a k-ary weighting of a clone C of operations on D. We call ω a* weighted polymorphism *of γ if $C \subseteq \mathrm{Pol}(\gamma)$ and for any $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) \in (\mathrm{Feas}(\gamma))^k$, we have*

$$\sum_{f \in C^{(k)}} \omega(f) \cdot \gamma(f(X)) = \sum_{f \in C^{(k)}} \omega(f) \cdot \gamma(f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k)) \; \leq \; 0 \,. \tag{9}$$

If ω is a weighted polymorphism of γ, we say that γ is improved *by ω.*

Example 4. Consider the class of submodular functions. These are precisely the functions γ defined on $D = \{0, 1\}$ satisfying $\gamma(\min(\mathbf{x}_1, \mathbf{x}_2)) + \gamma(\max(\mathbf{x}_1, \mathbf{x}_2)) - \gamma(\mathbf{x}_1) - \gamma(\mathbf{x}_2) \leq 0$, where min and max are the two binary operations that return the smaller and larger of their two arguments respectively (with respect to the usual order $0 < 1$). In other words, the set of submodular functions is the set of weighted relations improved by the binary weighting ω_{sub} defined by: $\omega_{sub}(f) = -1$ if $f \in \{e_1^{(2)}, e_2^{(2)}\}$, $\omega_{sub}(f) = +1$ if $f \in \{\min, \max\}$, and $\omega_{sub}(f) = 0$ for all other binary operations on D.

Definition 18. *For any $\Gamma \subseteq \mathbf{\Phi}_D$, we define* $\mathrm{wPol}(\Gamma)$ *to be the set of all weightings of* $\mathrm{Pol}(\Gamma)$ *which are weighted polymorphisms of all weighted relations $\gamma \in \Gamma$. We write* $\mathrm{wPol}(\gamma)$ *for* $\mathrm{wPol}(\{\gamma\})$.

Definition 19. *We denote by $\mathbf{W}_C$ the set of all possible weightings of clone C, and define $\mathbf{W}_D$ to be the union of the sets $\mathbf{W}_C$ over all clones C on D.*

Any $\Omega \subseteq \mathbf{W}_D$ may contain weightings of *different* clones over D. We can then extend each of these weightings with zeros, as necessary, so that they are weightings of the same clone C, where C is the smallest clone containing all the clones associated with weightings in Ω.

Definition 20. *We define* $\mathrm{wClone}(\Omega)$ *to be the smallest weighted clone containing this set of extended weightings obtained from Ω.*

For any $\Omega \subseteq \mathbf{W}_D$, we denote by $\mathrm{Imp}(\Omega)$ the set of all weighted relations in $\mathbf{\Phi}_D$ which are improved by all weightings $\omega \in \Omega$.

The main result in [9] establishes a 1-to-1 correspondence between weighted relational clones and weighted clones.

Theorem 2 ([9]).

1. *For any finite D and any finite $\Gamma \subseteq \mathbf{\Phi}_D$,* $\mathrm{Imp}(\mathrm{wPol}(\Gamma)) = \mathrm{wRelClone}(\Gamma)$.
2. *For any finite D and any finite $\Omega \subseteq \mathbf{W}_D$,* $\mathrm{wPol}(\mathrm{Imp}(\Omega)) = \mathrm{wClone}(\Omega)$.

Thus, when trying to identify tractable constraint languages, it is sufficient to consider only languages of the form $\mathrm{Imp}(\Omega)$ for some weighted clone Ω.

3 Results

First we show that Theorem 2 can be slightly extended to certain constraint languages and sets of weightings of infinite size.

Theorem 3.

1. *Let* $\Gamma \subseteq \mathbf{\Phi}_D$. *Then* $\mathrm{Imp}(\mathrm{wPol}(\Gamma)) = \mathrm{wRelClone}(\Gamma)$ *if and only if* $\mathrm{wRelClone}(\Gamma) = \mathrm{Imp}(\Omega)$ *for some* $\Omega \subseteq \mathbf{W}_D$.
2. *Let* $\Omega \subseteq \mathbf{W}_D$. *Then* $\mathrm{wPol}(\mathrm{Imp}(\Omega)) = \mathrm{wClone}(\Omega)$ *if and only if* $\mathrm{wClone}(\Omega) = \mathrm{wPol}(\Gamma)$ *for some* $\Gamma \subseteq \mathbf{\Phi}_D$.

Proof. We will only prove the first case as the second one is analogous. Suppose that $\mathrm{wRelClone}(\Gamma) = \mathrm{Imp}(\Omega)$ for some $\Omega \subseteq \mathbf{W}_D$. As $\Gamma \subseteq \mathrm{wRelClone}(\Gamma)$, every weighting in Ω improves Γ, hence $\Omega \subseteq \mathrm{wPol}(\Gamma)$ and $\mathrm{Imp}(\mathrm{wPol}(\Gamma)) \subseteq \mathrm{Imp}(\Omega) = \mathrm{wRelClone}(\Gamma)$. The inclusion $\mathrm{wRelClone}(\Gamma) \subseteq \mathrm{Imp}(\mathrm{wPol}(\Gamma))$ follows from the fact that $\mathrm{Imp}(\mathrm{wPol}(\Gamma))$ is a weighted relational clone [9, Proposition 6.2] that contains Γ.

The converse implication holds trivially for $\Omega = \mathrm{wPol}(\Gamma)$.

We remark that any *finitely generated* weighted relational clone on a finite domain satisfies, by Theorem 2 (1), the condition of Theorem 3 (1). Similarly, any finitely generated weighted clone on a finite domain, by Theorem 2 (2), satisfies the condition of Theorem 3 (2).

However, our next result shows that Theorem 2 does *not* hold for all infinite constraint languages and infinite sets of weightings.

Theorem 4. *There is a finite* D *and an infinite* $\Gamma \subseteq \mathbf{\Phi}_D$ *with* $\mathrm{Imp}(\mathrm{wPol}(\Gamma)) \neq \mathrm{wRelClone}(\Gamma)$. *Moreover, there is a finite* D *and an infinite* $\Omega \subseteq \mathbf{W}_D$ *with* $\mathrm{wPol}(\mathrm{Imp}(\Omega)) \neq \mathrm{wClone}(\Omega)$.

Our aim is to establish a Galois connection even for infinite sets of weighted relations and weightings. As we demonstrate in the proof of Theorem 4, this cannot be done when restricted to rational weights; hence we allow weighted relations and weightings to assign *real-valued* weights. To distinguish them from their formerly defined rational-valued counterparts, we will use a subscript/superscript $\mathbb{R}$.

We will show that $\mathrm{wPol}_{\mathbb{R}}(\Gamma)$ is a *closed* weighted clone for any set of weighted relations Γ; analogously, we will show that $\mathrm{Imp}_{\mathbb{R}}(\Omega)$ is a *closed* weighted relational clone for any set of weightings Ω. Therefore, the one-to-one correspondence between weighted relational clones and weighted clones which we want to establish cannot possibly hold for sets that are not closed. As there exist (infinite) sets $\Gamma \subseteq \mathbf{\Phi}_D^{\mathbb{R}}$, $\Omega \subseteq \mathbf{W}_D^{\mathbb{R}}$ such that $\mathrm{wRelClone}_{\mathbb{R}}(\Gamma)$, $\mathrm{wClone}_{\mathbb{R}}(\Omega)$ are not closed, we need to include the closure operator in the statement defining the Galois connection.

Inspired by weighted pp-definitions [20], we extend the notion of weighted relational clones: we require them to be closed under the Opt operator. This change is justified by a result in which we prove that the inclusion of Opt preserves tractability. In order to retain the one-to-one correspondence with weighted clones, we need to alter their definition too: weightings now assign weights to all operations and hence are independent of the support clone (which becomes meaningless and we discard it).

Including the Opt operator brings two advantages to the study of weighted clones. Firstly, it slightly simplifies the structure of the space of all weighted clones. According to the original definition, a weighted clone is determined by its support clone and the set of weightings it consists of; by our definition a weighted clone equals the set of its weightings. Secondly, any non-projection polymorphism of a weighted relational clone Γ is assigned a positive weight by some weighted polymorphism of Γ.

Our main result is the following theorem, which holds for our new definition of real-valued weightings and weighted relations.

Theorem 5 (Main).

1. *For any finite D and any $\Gamma \subseteq \mathbf{\Phi}_D^{\mathbb{R}}$,* $\mathrm{Imp}_{\mathbb{R}}(\mathrm{wPol}_{\mathbb{R}}(\Gamma)) = \overline{\mathrm{wRelClone}_{\mathbb{R}}(\Gamma)}$. *Moreover, if Γ is finite, then* $\mathrm{Imp}_{\mathbb{R}}(\mathrm{wPol}_{\mathbb{R}}(\Gamma)) = \mathrm{wRelClone}_{\mathbb{R}}(\Gamma)$.
2. *For any finite D and any $\Omega \subseteq \mathbf{W}_D^{\mathbb{R}}$,* $\mathrm{wPol}_{\mathbb{R}}(\mathrm{Imp}_{\mathbb{R}}(\Omega)) = \overline{\mathrm{wClone}_{\mathbb{R}}(\Omega)}$. *Moreover, if Ω is finite, then* $\mathrm{wPol}_{\mathbb{R}}(\mathrm{Imp}_{\mathbb{R}}(\Omega)) = \mathrm{wClone}_{\mathbb{R}}(\Omega)$.

Finally, we show that taking the weighted relational clone of a constraint language preserves solvability with an absolute error bounded by ϵ (for any $\epsilon > 0$).

4 New Galois Connection

Let $\overline{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ denote the set of real numbers with (positive) infinity. We will allow weights in relations and weighted relations, as defined in Definition 1 and 2 respectively, to be real numbers. In other words, an m-ary weighted relation γ on D is a mapping $\gamma : D^m \to \overline{\mathbb{R}}$. We will add a subscript/superscript $\mathbb{R}$ to the notation introduced in Section 2 in order to emphasise the use of real weights.

For any fixed arity m and any $F \subseteq D^m$, consider the set of all m-ary weighted relations $\gamma \in \mathbf{\Phi}_D^{\mathbb{R}}$ with $\mathrm{Feas}(\gamma) = F$. Let us denote this set by H and equip it with the inner product defined as

$$\langle \alpha, \beta \rangle = \sum_{\mathbf{x} \in F} \alpha(\mathbf{x}) \cdot \beta(\mathbf{x}) \tag{10}$$

for any $\alpha, \beta \in H$; H is then a real Hilbert space. Set $\mathbf{\Phi}_D^{\mathbb{R}}$ is a disjoint union of such Hilbert spaces for all m and F, and therefore a topological space with the disjoint union topology induced by inner products on the underlying Hilbert spaces. When we say a set of weighted relations is open/closed, we will be referring to this topology.

Definition 21. *A constraint language* $\Gamma \subseteq \mathbf{\Phi}_D^{\mathbb{R}}$ *is called a* weighted relational clone *if it contains the binary equality relation* $\phi_=$ *and the unary empty relation* $\phi_\emptyset$*, and is closed under addition, minimisation, scaling by non-negative real constants, addition of real constants, and under the* Opt *operator.*

For any Γ*, we define* $\mathrm{wRelClone}_{\mathbb{R}}(\Gamma)$ *to be the smallest weighted relational clone containing* Γ*.*

For a weighted relational clone Γ, its topological closure $\overline{\Gamma}$ is also a weighted relational clone, as all the operations that we require weighted relational clones to be closed under are continuous mappings.

As opposed to Definition 9, our new definition requires weighted relational clones to be closed under operator Opt. In order to establish a Galois connection now, we need to make an adjustment to the definition of weighted clone too. We will discard the explicit underlying support clone; instead, (k-ary) weightings will assign weights to all (k-ary) operations. The role of the support clone of a weighted clone Ω is then taken over by $\mathrm{supp}(\Omega)$ (see Lemma 1).

We denote by $\mathcal{O}_D^{(k)}$ the set of all k-ary operations on D and let $\mathcal{O}_D = \bigcup_{k \geq 0} \mathcal{O}_D^{(k)}$.

Definition 22. *A k-ary* weighting *is a function* $\omega : \mathcal{O}_D^{(k)} \to \mathbb{R}$ *such that* $\omega(f) < 0$ *only if* f *is a projection and*

$$\sum_{f \in \mathcal{O}_D^{(k)}} \omega(f) \;=\; 0\,. \tag{11}$$

We define $\mathrm{supp}(\omega) = \{f \in \mathcal{O}_D^{(k)} \mid \omega(f) > 0 \vee f \in \mathbf{J}_D^{(k)}\}$.

We will call a function $\omega : \mathcal{O}_D^{(k)} \to \mathbb{R}$ *that satisfies Equation* (11) *but assigns a negative weight to some operation* $f \notin \mathbf{J}_D^{(k)}$ *an* improper weighting. *In order to emphasise the distinction we may also call a weighting a* proper weighting.

We denote by $\mathbf{W}_D^{\mathbb{R}}$ the set of all weightings on domain D. For any fixed arity k, consider the set H of all functions $\mathcal{O}_D^{(k)} \to \mathbb{R}$ equipped with the inner product defined as

$$\langle \alpha, \beta \rangle = \sum_{f \in \mathcal{O}_D^{(k)}} \alpha(f) \cdot \beta(f) \tag{12}$$

for any $\alpha, \beta \in H$; H is then a real Hilbert space. Set $\mathbf{W}_D^{\mathbb{R}}$ lies in the disjoint union of such Hilbert spaces for all k, which is a topological space with the disjoint union topology induced by inner products on the underlying Hilbert spaces. When we say a set of weightings is open/closed, we will be referring to this topology. Clearly, any closure point of a set of weightings is itself a weighting.

Definition 23. *Let* Ω *be a non-empty set of weightings on a fixed domain* D*. We define* $\mathrm{supp}(\Omega) = \mathbf{J}_D \cup \bigcup_{\omega \in \Omega} \mathrm{supp}(\omega)$.

We call Ω *a* weighted clone *if it is closed under scaling by non-negative real constants, addition of weightings of equal arity, and proper superposition with operations from* $\mathrm{supp}(\Omega)$.

For any weighted clone Ω, its topological closure $\overline{\Omega}$ is also a weighted clone, as all the operations that we require weighted clones to be closed under are continuous mappings.

Again, we link weightings and weighted relations by the concept of weighted polymorphism.

Definition 24. *Let γ be an m-ary weighted relation on D and let ω be a k-ary weighting on D. We call ω a* weighted polymorphism *of γ if $\operatorname{supp}(\omega) \subseteq \operatorname{Pol}(\gamma)$ and for any $X = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k) \in (\operatorname{Feas}(\gamma))^k$, we have*

$$\sum_{f \in \operatorname{supp}(\omega)} \omega(f) \cdot \gamma(f(X)) = \sum_{f \in \operatorname{supp}(\omega)} \omega(f) \cdot \gamma(f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_k)) \;\leq\; 0\,. \tag{13}$$

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

In the proof of Theorem 5, we will often use the following characterisation of weighted polymorphisms. Let $\gamma \in \mathbf{\Phi}_D^{\mathbb{R}}$ be a weighted relation and $\omega \in \mathbf{W}_D^{\mathbb{R}}$ a k-ary weighting such that $\operatorname{supp}(\omega) \subseteq \operatorname{Pol}(\gamma)$. Let us denote by H the Hilbert space of functions $\operatorname{Pol}^{(k)}(\gamma) \to \mathbb{R}$ with the inner product analogous to (12). As weighting ω assigns non-zero weights only to operations from $\operatorname{supp}(\omega) \subseteq \operatorname{Pol}^{(k)}(\gamma)$, we can identify ω with its restriction to $\operatorname{Pol}^{(k)}(\gamma)$. For any $X \in (\operatorname{Feas}(\gamma))^k$, we define $\gamma[X] \in H$ as $\gamma[X](f) = \gamma(f(X))$. Inequality (13) is then equivalent to $\langle \omega, \gamma[X] \rangle \leq 0$.

The (internal) polar cone K° of a set $K \subseteq H$ is defined as

$$K^\circ = \{\alpha \in H \mid \langle \alpha, \beta \rangle \leq 0 \text{ for all } \beta \in H\}\,. \tag{14}$$

It is well known ([3]) that K° is a convex cone, i.e. K° is closed under addition of vectors and scaling by non-negative constants. Moreover, K° is a closed set, and $K^{\circ\circ} = (K^\circ)^\circ$ is the closure of the smallest convex cone containing K. If K is a finite set, then the smallest convex cone containing K is closed. Let $K = \{\gamma[X] \mid X \in (\operatorname{Feas}(\gamma))^k\}$; weighting ω is then a weighted polymorphism of γ if and only if $\omega \in K^\circ$.

The following lemma (and its corollary) shows that $\operatorname{supp}(\Omega)$ consists of all polymorphisms of $\operatorname{Imp}_{\mathbb{R}}(\Omega)$ and hence fulfills the same role as the support clone in Definition 16.

Lemma 1. *Let $\Omega \subseteq \mathbf{W}_D^{\mathbb{R}}$ be a weighted clone. Then $\operatorname{supp}(\Omega) = \operatorname{Pol}(\operatorname{Imp}_{\mathbb{R}}(\Omega))$.*

Corollary 1. *Let $\Gamma \subseteq \mathbf{\Phi}_D^{\mathbb{R}}$ be a weighted relational clone. Then we have that $\operatorname{supp}(\operatorname{wPol}_{\mathbb{R}}(\Gamma)) = \operatorname{Pol}(\Gamma)$.*

Theorem 6. *Let $\Gamma, \Gamma' \subseteq \mathbf{\Phi}_D^{\mathbb{R}}$ be finite constraint languages such that Γ contains only weighted relations of the form $c \cdot \gamma'$ for $c \geq 0, \gamma' \in \Gamma'$. For any $\epsilon > 0$ there is a polynomial-time reduction that for any instance $I \in \operatorname{VCSP}(\Gamma)$ outputs an instance $I' \in \operatorname{VCSP}(\Gamma')$ such that for any optimal assignment s' of I' it holds $I(s') \in [v, v + \epsilon]$, where v is the value of an optimal assignment of I.*

References

1. L. Barto, M. Kozik, Constraint Satisfaction Problems Solvable by Local Consistency Methods, Journal of the ACM 61 (1), article No. 3.
2. L. Barto, M. Kozik, T. Niven, The CSP dichotomy holds for digraphs with no sources and no sinks SIAM Journal on Computing 38 (5) (2009) 1782–1802.
3. S.P. Boyd, L. Vandenberghe, Convex Optimization, CUP, 2004.
4. A. Bulatov, A Graph of a Relational Structure and Constraint Satisfaction Problems, in: Proc. LICS'04, IEEE Computer Society, 2004, pp. 448–457.
5. A. Bulatov, A dichotomy theorem for constraint satisfaction problems on a 3-element set, Journal of the ACM 53 (1) (2006) 66–120.
6. A. Bulatov, A. Krokhin, P. Jeavons, Classifying the Complexity of Constraints using Finite Algebras, SIAM Journal on Computing 34 (3) (2005) 720–742.
7. A. A. Bulatov, Complexity of conservative constraint satisfaction problems, ACM Transactions on Computational Logic 12 (4), article 24.
8. A. A. Bulatov, A. A. Krokhin, P. G. Jeavons, The complexity of maximal constraint languages, in: Proc. STOC'01, 2001, pp. 667–674.
9. D. A. Cohen, M. C. Cooper, P. Creed, P. Jeavons, S. Živný, An algebraic theory of complexity for discrete optimisation, SIAM Journal on Computing 42 (5) (2013) 915–1939.
10. D. A. Cohen, M. C. Cooper, P. G. Jeavons, A. A. Krokhin, The Complexity of Soft Constraint Satisfaction, Artificial Intelligence 170 (11) (2006) 983–1016.
11. T. Feder, M. Y. Vardi, The Computational Structure of Monotone Monadic SNP and Constraint Satisfaction: A Study through Datalog and Group Theory, SIAM Journal on Computing 28 (1) (1998) 57–104.
12. P. Hell, J. Nešetřil, On the Complexity of H-coloring, Journal of Combinatorial Theory, Series B 48 (1) (1990) 92–110.
13. P. Hell, J. Nešetřil, Colouring, constraint satisfaction, and complexity, Computer Science Review 2 (3) (2008) 143–163.
14. P. Jeavons, A. Krokhin, S. Živný, The complexity of valued constraint satisfaction, Bulletin of the European Association for Theoretical Computer Science (EATCS) 113 (2014) 21–55.
15. P. G. Jeavons, D. A. Cohen, M. Gyssens, Closure Properties of Constraints, Journal of the ACM 44 (4) (1997) 527–548.
16. V. Kolmogorov, J. Thapper, S. Živný, The power of linear programming for general-valued CSPs, SIAM Journal on Computing, 44(1), 1–36, 2015.
17. V. Kolmogorov, S. Živný, The complexity of conservative valued CSPs, Journal of the ACM 60 (2), article No. 10.
18. Marcin Kozik and Joanna Ochremiak. Algebraic Properties of Valued Constraint Satisfaction Problem. In *Proc. ICALP'15*, Springer, 2015.
19. T. J. Schaefer, The Complexity of Satisfiability Problems, in: Proc. STOC'78, ACM, 1978, pp. 216–226.
20. J. Thapper, Aspects of a constraint optimisation problem, Ph.D. thesis, Department of Computer Science and Information Science, Linköping University (2010).
21. J. Thapper, S. Živný, The power of linear programming for valued CSPs, in: Proc. FOCS'12, IEEE, 2012, pp. 669–678.
22. J. Thapper, S. Živný, The complexity of finite-valued CSPs, in: Proc. STOC'13, ACM, 2013, pp. 695–704.
23. H. Uppman, The Complexity of Three-Element Min-Sol and Conservative Min-Cost-Hom, in: Proc. ICALP'13, vol. 7965 of Lecture Notes in Computer Science, Springer, 2013, pp. 804–815.