The Complexity of Phylogeny Constraint Satisfaction

Manuel Bodirsky[1], Peter Jonsson[2], and Trung Van Pham[3]

1 **Institut für Algebra, TU Dresden, Germany**
manuel.bodirsky@tu-dresden.de
2 **Department of Computer and System Science, Linköpings Universitet, Sweden**
peter.jonsson@liu.se
3 **Institut für Algebra, TU Dresden, Germany**
pvtrung@math.ac.vn

Abstract

We systematically study the computational complexity of a broad class of computational problems in phylogenetic reconstruction. The class contains for example the rooted triple consistency problem, forbidden subtree problems, the quartet consistency problem, and many other problems studied in the bioinformatics literature. The studied problems can be described as *constraint satisfaction problems* where the constraints have a first-order definition over the rooted triple relation. We show that every such phylogeny problem can be solved in polynomial time or is NP-complete. On the algorithmic side, we generalize a well-known polynomial-time algorithm of Aho, Sagiv, Szymanski, and Ullman for the rooted triple consistency problem. Our algorithm repeatedly solves linear equation systems to construct a solution in polynomial time. We then show that every phylogeny problem that cannot be solved by our algorithm is NP-complete. Our classification establishes a dichotomy for a large class of infinite structures that we believe is of independent interest in universal algebra, model theory, and topology. The proof of our main result combines results and techniques from various research areas: a recent classification of the model-complete cores of the reducts of the homogeneous binary branching C-relation, Leeb's Ramsey theorem for rooted trees, and universal algebra.

1 Introduction

Phylogenetic consistency problems are computational problems that have been studied for phylogenetic reconstruction in computational biology, but also in other areas dealing with large amounts of possibly inconsistent data about trees, such as computational genealogy or computational linguistics. Given a collection of *partial information* about a tree, we would like to know whether the information is *consistent* in the sense that there exists a single tree that it is compatible with all the given partial information. A concrete example of a computational problem in this context is the *rooted triple consistency problem*. In an instance of this problem, we are given a set V of variables, and a set of triples from V^3, written in the form $ab|c$ where $a, b, c \in V$, and we would like to know whether there exists a rooted tree T whose leaves are from V such that for each of the given triples $ab|c$ the youngest common ancestor of a and b in this tree is below the youngest common ancestor of a and c. Aho, Sagiv, Szymanski, and Ullmann presented a polynomial-time algorithm for this problem [1].

Many computational problems that are defined similarly as the rooted triple consistency problem have been studied in the literature. Examples include the *subtree avoidance problem* (Ng, Steel, and Wormald [24]) and the *forbidden triple problem* (Bryant [16]) which are NP-hard problems. Bodirsky & Mueller [8] have determined the complexity of rooted phylogeny problems for the special case where the where the relations are disjunctive combinations of

the rooted triple relation. This result covers, for instance, the subtree avoidance problem and the forbidden triple problem.

We present a considerable strengthening of this result, and classify the complexity of phylogeny problems for all sets of phylogeny constraints that can be first-order defined with the mentioned rooted triple relation and equality (of leaves). The reader should be aware that many problems of this type may appear exotic from a biological point of view — the name "phylogeny" should not be taken too literally. Our results show that each of the problem problems obtained in this way is either polynomial-time solvable or NP-complete. As we will demonstrate later (see Section 2), this class of problems is expressive enough to contain also *unrooted phylogeny problems*. A famous example of such an unrooted phylogeny problem is the NP-complete *quartet consistency problem* (Steel [25]): here we are given a set V of variables, and a set of quartets ab:cd with $a, b, c, d \in V$, and we would like to know whether there exists a tree T with leaves from V such that for each of the given quartets ab:cd the shortest path from a to b does not intersect the shortest path from c to d in T. Another phylogeny problem that has been studied in the literature and that falls into the framework of this paper (but not into the one in [8]) is the *tree discovery problem* [1]: here, the input consists of a set of 4-tuples of variables, and the task is to find a rooted tree T such that for each quadruple (x, y, u, v) in the input the youngest common ancestor of x and y is a proper descendant of the youngest common ancestor of u and v.

The proof of the complexity classification is based on a variety of methods and results. Our first step is that we give an alternative description of phylogeny problems as constraint satisfaction problems (CSPs) over a countably infinite domain where the constraint relations are first-order definable over the (up to isomorphism unique) *homogeneous binary branching C-relation*, a well-known structure in model theory. We let C denote this particular relation. A central result that simplifies our work considerably is a recent analysis of the endomorphism monoids of such relations [5]. Informally, this result implies that there are precisely four types of phylogeny problems: (1) trivial (i.e., if there is a solution, there is a constant solution), (2) rooted, (3) unrooted, and (4) degenerate cases that have been called *equality CSPs* [6]. Rooted and unrooted phylogeny problems will be introduced formally in Theorem 14. We will show that all unrooted phylogeny problems are NP-hard, and the complexity of all equality CSPs is already known.

The basic method to proceed from there is the *algebraic approach* to constraint satisfaction problems. Here, one studies certain sets of operations (known as *polymorphisms*) instead of analyzing the constraints themselves. An important tool to work with polymorphisms over infinite domains is Ramsey theory. In this paper, we need a Ramsey result for rooted trees due to Leeb [23], for proving that polymorphisms behave canonically on large parts of the domain (in the sense of Bodirsky & Pinsker [10]), and this allows us to perform a simplified combinatorial analysis.

Interestingly, all phylogeny problems that can be solved in polynomial time fall into one class and can be solved by the same algorithm. This algorithm is a considerable extension of the algorithm by Bodirsky & Mueller [8] for the rooted triple consistency problem, and the algorithm by Bodirsky & Mueller is in turn a considerable extension of the algorithm by Aho, Sagiv, Szymanski, and Ullmann [1]. The algorithm by Aho et al. is based on analysing connected components in particular graphs while our algorithm is based on repeatedly solving systems of linear Boolean equations. An illustrative example of a phylogeny problem that can be solved in polynomial time by our algorithm, but not the algorithms from [1, 8], is the following computational problem: the input is a 4-uniform hypergraph with vertex set V; the question is whether there exists a rooted tree T with leaf set V such that every

hyperedge in the input T has two disjoint subtrees that each contain precisely two of the vertices of the hyperedge.

All phylogeny problems that cannot be solved by our algorithm are NP-complete. Our results are stronger than this complexity dichotomy, though, and we prove that every phylogeny problem satisfies a universal-algebraic dichotomy statement that holds for a large class of infinite structures (Theorem 24), which is of independent interest in the study of homogeneous structures and their polymorphism clones. In this respect, the situation is similar as in previous classifications for CSPs where the constraints are first-order definable over the order of the rationals $(\mathbb{Q}; <)$ from [7] or the random graph [11]. In comparison to these previous works, the dichotomy we present here is easier to state (there is just one tractable class), but harder to prove with the existing methods: in particular, unlike the situation for constraints that are first-order definable over the random graph [11], the polymorphisms that characterise the tractable cases cannot be chosen to be canonical (in the sense of Bodirsky & Pinsker [10]) on the entire domain. As such, our dichotomy provides an important test-case for potentially much wider classifications of CSPs over homogeneous structures.

The paper has the following structure. We give basic definitions concerning phylogeny problems in Section 2 and explain how these problems can be viewed as constraint satisfaction problems for reducts of the homogeneous binary branching C-relation. Section 3 provides a brief but self-contained introduction to the universal-algebraic approach. In Section 4 we translate structural properties of phylogenetic relations into definability properties in terms of syntactically restricted formulas, which we call *affine Horn formulas*. Here we also state our tractability result. In Section 5, we characterize the tractable class of phylogeny problems with polymorphisms. Finally, in Section 6 we put everything together and state and prove our main results including the previously mentioned complexity dichotomy. Section 5 can safely be skipped by readers that are only interested in the complexity dichotomy and not in the stronger algebraic dichotomy. A report version of this paper with full proofs can be found in the appendix.

2 Phylogeny problems

All structures in this paper are assumed to be countable. In this section, we first define (in Sections 2.1 and 2.2) a class of phylogeny problems and illustrate it by showing instances from this class that have been studied in the literature. We continue in Section 2.3 by showing how to formulate such phylogeny problems as *constraint satisfaction problems* over an infinite domain.

2.1 Rooted trees

We fix some standard terminology concerning rooted trees. Let T be a tree (i.e., an undirected, acyclic, and connected graph) with a distinguished vertex r, the *root* of T. The vertices of T are denoted by $V(T)$. All trees in this paper will be *binary*, i.e., all vertices except for the root have either degree 3 or 1, and the root has either degree 2 or 0. The *leaves* $L(T)$ of T are the vertices of T of degree one.

For $u, v \in V(T)$, we say that u *lies below* v if the path from u to r passes through v. We say that u *lies strictly below* v if u lies below v and $u \neq v$. The *youngest common ancestor (yca)* of a set of vertices $S \subseteq V(T)$ is the node u that lies above all vertices in S and has maximal distance from r; this node is uniquely determined by S.

▶ **Definition 1.** The *leaf structure* of a binary rooted tree T is the relational structure $(L(T); C)$ where $C(a, b, c)$ holds in C if and only if $\mathrm{yca}(\{b, c\})$ lies strictly below $\mathrm{yca}(\{a, b, c\})$ in T. We also call T the *underlying tree* of the leaf structure.

It is well-known that a rooted tree is uniquely determined by its leaf structure.

▶ **Definition 2.** For finite $S_1, S_2 \subseteq L(T)$, we write $S_1|S_2$ if neither of $\mathrm{yca}(S_1)$ and $\mathrm{yca}(S_2)$ lies below the other. For sequences of (not necessarily distinct) vertices $x_1, \dots, x_n$ and $y_1, \dots, y_m$ with $n, m \geq 1$ we write $x_1, \dots, x_n|y_1, \dots, y_m$ if $\big(\bigcup_{1\leq i\leq n}\{x_i\}\big)\big|\big(\bigcup_{1\leq i\leq m}\{y_i\}\big)$.

In particular, $x|yz$ (which is the notation that is typically used in the literature on phylogeny problems) is equivalent to $C(x, y, z)$. Note that if $x|yz$ then this includes the possibility that $y = z$; however, $x|yz$ implies that $x \neq y$ and $x \neq z$. Hence, for every triple x, y, z of leaves in a rooted binary tree, exactly one of $x|yz$, $y|xz$, $z|xy$, $x = y = z$ holds. Also note that $x_1, \dots, x_n|y_1, \dots, y_m$ if and only if $x_ix_j|y_k$ and $x_i|y_ky_l$ for all $i, j \leq n$ and $k, l \leq m$.

2.2 Phylogeny problems

An *atomic phylogeny formula* is a formula of the form $x|yz$ or of the form $x = y$. A *phylogeny formula* is a quantifier-free formula ϕ that is built from atomic phylogeny formulas with the usual Boolean connectives (disjunction, conjunction, negation).

We say that a phylogeny formula ϕ with variables V is *satisfiable* if there exists a rooted binary tree T and a mapping $s\colon V \to L(T)$ such that ϕ is satisfied by T under s (with the usual semantics of first-order logic). In this case we also say that (T, s) is a *solution* to ϕ.

Let $\Phi = \{\phi_1, \phi_2, \dots\}$ be a finite set of phylogeny formulas. Then the *phylogeny problem* for Φ is the following computational problem.

Phylo(Φ)
INSTANCE: A finite set V of variables, and a finite set Ψ of phylogeny formulas obtained from phylogeny formulas $\phi \in \Phi$ by substituting the variables from ϕ by variables from V.
QUESTION: Is there a tree T and a mapping $s\colon V \to L(T)$ such that (T, s) satisfies all formulas from Ψ?

We use $x_1, \dots, x_n|y_1, \dots, y_m$ as a shortcut for $\bigwedge_{i,j\in\{1,\dots,n\},k,l\in\{1,\dots,m\}}(y_k|x_ix_j \wedge x_i|y_ky_l)$ and we use $\text{all-diff}(x_1, \dots, x_k)$ as a shortcut for $\bigwedge_{1\leq i<j\leq k} x_i \neq x_j$.

▶ **Example 3.** The following NP-complete problem was introduced and studied in a closely related form by Ng, Steel, and Wormald [24]. We are given a set of rooted trees on a common leave set V, and we would like to know whether there exists a tree T with leave set V such that, intuitively, for each of the given trees T' the tree T does *not* match with the tree T'. The hardness proof for this problem given Ng, Steel, and Wormald [24] shows that already the phylogeny problem $\mathrm{Phylo}\big(\{\neg x|yz \wedge \text{all-diff}(x, y, z), \neg(u|xy \wedge v|yu) \wedge \text{all-diff}(x, y, u, v)\}\big)$, which can be seen as a special case of the problem above, is NP-hard. ◀

▶ **Example 4.** The quartet consistency problem described in the introduction can be cast as $\mathrm{Phylo}(\{\phi\})$ where ϕ is the phylogeny formula $(xy|u \wedge xy|v) \vee (x|uv \wedge y|uv)$. Indeed, this formula describes all rooted trees with leaves x, y, u, v where the shortest path from x to y does not intersect the shortest path from u to v (whether or not this is true is in fact independent from the position of the root). ◀

Our main result is a full classification of the computational complexity of $\mathrm{Phylo}(\Phi)$.

▶ **Theorem 5.** *Let Φ be a finite set of phylogeny formulas. Then Phylo(Φ) is in P or NP-complete.*

2.3 Phylogeny problems as CSPs

As mentioned in the introduction, every phylogeny problem can be formulated as a constraint satisfaction problem over an infinite domain. This reformulation will be essential for using universal-algebraic and Ramsey-theoretic tools.

Let Γ be a structure with relational signature $\tau = \{R_1, R_2, \dots\}$. This is, Γ is a tuple $(D; R_1^\Gamma, R_2^\Gamma, \dots)$ where D is the (finite or infinite) *domain* of Γ and where $R_i^\Gamma \subseteq D^{k_i}$ is a relation of arity k_i over D. When Δ and Γ are two τ-structures, then a *homomorphism* from Δ to Γ is a mapping h from the domain of Δ to the domain of Γ such that for all $R \in \tau$ and for all $(x_1, \dots, x_k) \in R^\Delta$ we have $(h(x_1), \dots, h(x_k)) \in R^\Gamma$.

Suppose that the signature τ of Γ is finite. Then the *constraint satisfaction problem for* Γ, denoted by $\mathrm{CSP}(\Gamma)$, is the following computational problem.

CSP(Γ)
INSTANCE: A finite τ-structure Δ.
QUESTION: Is there a homomorphism from Δ to Γ?

We say that Γ is the *template* or the *constraint language* of the problem $\mathrm{CSP}(\Gamma)$. To formulate phylogeny problems as CSPs, let $\Phi = \{\phi_1, \dots, \phi_n\}$ be a finite set of phylogeny formulas. If $x_1, \dots, x_{k_i}$ are the variables of ϕ_i, then we introduce a new relation symbol R_i of arity k_i, and we write τ for the set of all these relation symbols.

For an instance Ψ of $\mathrm{Phyl}(\Phi)$ with variables V, we associate to Ψ a τ-structure Δ_Ψ with domain V as follows. For $R \in \tau$ of arity k, the relation R^Δ contains the tuple $(y_1, \dots, y_k) \in V^k$ if and only if the instance Ψ contains a formula ψ that has been obtained from a formula $\phi \in \Phi$ by replacing the variables $x_1, \dots, x_k$ of ϕ by the variables $y_1, \dots, y_k \in V$.

▶ Proposition 6. Let Φ be a finite set of phylogeny formulas. Then there exists a τ-structure Γ_Φ with countable domain $\mathbb{L}$ and the following property: an instance Ψ of $\mathrm{Phyl}(\Phi)$ is satisfiable if and only if Δ_Ψ homomorphically maps to Γ_Φ.

The structure Γ_Φ in Proposition 6 is by no means unique, and such structures are easy to construct. The specific choice for Γ_Φ presented below is important later in the proof of our complexity classification; as we will see, it has many pleasant model-theoretic properties. To define Γ_Φ, we first define a 'base structure' $(\mathbb{L}; C)$, and then define Γ_Φ in terms of $(\mathbb{L}; C)$. The structure $(\mathbb{L}; C)$ is a well-studied object in model theory and the theory of infinite permutation groups, and will be defined via *Fraïssé-amalgamation*.

Homomorphisms from Γ to Γ are called *endomorphisms* of Γ. An *automorphism* of Γ is a bijective endomorphism whose inverse is also an endomorphism. The set containing all endomorphisms of Γ is denoted $\mathrm{End}(\Gamma)$ while the set of all automorphisms is denoted $\mathrm{Aut}(\Gamma)$. A relational structure Γ is called *homogeneous* if every isomorphism between finite induced substructures of Γ can be extended to an automorphism of Γ. Homogeneous structures Γ with finite relational signature are *ω-categorical*, i.e., all countable structures that satisfy the same first-order sentences as Γ are isomorphic (see e.g. Cameron [18] or Hodges [21]).

When working with relational structures, it is often convenient to not distinguish between a relation and its relation symbol. For instance, when we write $(L(T), C)$ for a leaf structure (Definition 1), the letter C stands both for the relation symbol, and for the relation itself. This should never cause confusion.

▶ Proposition 7 (Proposition 7 in Bodirsky, Jonsson, & Van Pham [5]). There exists an (up to isomorphism unique) homogeneous structure $(\mathbb{L}; C)$ with the property that all its finite substructures are isomorphic to leaf structures of finite rooted binary trees.

The structure $(\mathbb{L}; C)$ is well-studied in the literature, and the relation C is commonly referred to as the *binary branching homogeneous C-relation*.

▶ **Definition 8.** Let Δ be a structure. Then a relational structure Γ with the same domain as Δ is called a *reduct* of Δ if all relations of Γ have a first-order definition in Δ (using conjunction, disjunction, negation, universal and existential quantification, as usual).

It is well-known that all structures with a first-order definition in an ω-categorical structures are again ω-categorical (we refer once again to Hodges [21], Theorem 7.3.8; the analogous statement for homogeneity is false.) Furthermore, an ω-categorical structure is homogeneous if and only if it has *quantifier-elimination*, that is, every first-order formula is over Γ equivalent to a quantifier-free formula; see Hodges [21].

Proof of Proposition 6. Let Φ be a finite set of phylogeny formulas. Let Γ_Φ be the reduct of $(\mathbb{L}; C)$ defined as follows. For every $\phi \in \Phi$ with free variables $x_1, \ldots, x_k$, we have the k-ary relation R_ϕ in Γ_ϕ which is defined by the formula ϕ over $(\mathbb{L}; C)$. It follows straightforwardly from the definitions that this structure has the properties required in the statement of Proposition 6.

Conversely, every CSP for a reduct $\Gamma = (\mathbb{L}; R_1, \ldots, R_n)$ of $(\mathbb{L}; C)$ corresponds to a phylogeny problem. Let ϕ_i be a quantifier-free first-order definition of R_i in $(\mathbb{L}; C)$. When Δ is an instance of CSP(Γ), consider the instance Ψ of Phyl($\{\phi_1, \ldots, \phi_n\}$) where the variables V are the vertices of Δ, and where Ψ contains for every tuple $(v_1, \ldots, v_n) \in R_i^\Delta$ the formula $\phi_i(v_1, \ldots, v_n)$. It is again straightforward to verify that Δ homomorphically maps to Γ if and only if Ψ is a satisfiable instance of Phyl($\{\phi_1, \ldots, \phi_n\}$). Therefore, the class of phylogeny problems corresponds precisely to the class of CSPs whose template is a reduct of $(\mathbb{L}; C)$. ◀

3 The universal-algebraic approach

We apply the so-called *universal-algebraic approach* to obtain our results. For a more detailed introduction to this approach, see Bodirsky [3]. We discuss some important concepts and present certain results in the following three subsections.

3.1 Primitive positive definability and interpretability

A first-order formula ϕ with free variables $z_1, \ldots, z_k$ over the signature τ is *primitive positive* if it is of the form $\exists x_1, \ldots, x_n\, (\psi_1 \wedge \cdots \wedge \psi_m)$, where $\psi_1, \ldots, \psi_m$ are *atomic*, that is, of the form $R(y_1, \ldots, y_k)$ or of the form $y_1 = y_2$, for $R \in \tau$ and $y_1, \ldots, y_k \in \{x_1, \ldots, x_n, z_1, \ldots, z_k\}$. When Γ is a τ-structure, then ϕ defines over Γ a k-ary relation, namely the set of all k-tuples that satisfy ϕ in Γ. We let $\langle\Gamma\rangle$ denote the set of all finitary relations that are primitive positive definable in Γ. The following result motivates why we are interested in positive primitive definability in connection with the complexity of CSPs.

▶ **Lemma 9** (Jeavons [22])**.** *Let Γ be a constraint language, and let Γ' be the structure obtained from Γ by adding the relation R. If R is primitive positive definable in Γ, then* CSP(Γ) *and* CSP(Γ') *are polynomial-time equivalent.*

This result was originally proved for finite-domains CSPs but the proof extends immediately to infinite-domain CSPs.

Primitive positive interpretations are a generalisation of primitive positive definitions, and are often used for proving NP-hardness results; we refer the reader to Bodirsky [3] for more information about this. We will consider the relation $\mathrm{NAE} = \{0,1\}^3 \setminus \{(0,0,0),(1,1,1)\}$

in connection with primitive positive interpretations. Clearly $\mathrm{CSP}(\{0,1\};\mathrm{NAE})$ is NP-complete.

▶ **Definition 10.** A relational σ-structure Δ has a *(first-order) interpretation* I in a τ-structure Γ if there exists a natural number d, called the *dimension* of I, and

- a τ-formula $\delta_I(x_1,\dots,x_d)$ – called the *domain formula*,
- for each atomic σ-formula $\phi(y_1,\dots,y_k)$ a τ-formula $\phi_I(\overline{x}_1,\dots,\overline{x}_k)$ where the $\overline{x}_i$ denote disjoint d-tuples of distinct variables – called the *defining formulas*,
- a surjective map h from all d-tuples of elements of Γ that satisfy δ_I to Δ – called the *coordinate map*,

such that for all atomic σ-formulas ϕ and all tuples in the domain of h, $\Delta \models \phi(h(\overline{a}_1),\dots,h(\overline{a}_k))$ if and only if $\Gamma \models \phi_I(\overline{a}_1,\dots,\overline{a}_k)$.

If the formulas δ_I and ϕ_I are all primitive positive, we say that the interpretation I is *primitive positive*. We say that Δ is *pp interpretable with parameters in* Γ is Δ has an interpretation I where the formulas δ_I and ϕ_I might involve elements from Γ (the *parameters*). That is, interpretations with parameters in Γ interpretations in the expansion of Γ by finitely many constants. The importance of primitive positive interpretations follows from the following lemma.

▶ **Lemma 11.** *Let Γ and Δ be structures with finite relational signature. Suppose that Γ is ω-categorical and that Δ has a primitive positive interpretation in Γ. Then there is a polynomial-time reduction from $\mathrm{CSP}(\Delta)$ to $\mathrm{CSP}(\Gamma)$. If Γ is a model-complete core, then the interpretation might even be with parameters and the conclusion of the lemma still holds.*

3.2 Polymorphisms

Primitive positive definability can be characterised by preservation under so-called *polymorphisms* – this is the starting point of the universal-algebraic approach to constraint satisfaction (see, for instance, Bulatov, Jeavons, and Krokhin [17] for this approach over finite domains). The *(direct–, categorical–, or cross–) product* $\Gamma_1 \times \Gamma_2$ of two relational τ-structures Γ_1 and Γ_2 is a τ-structure on the domain $D_{\Gamma_1} \times D_{\Gamma_2}$. For all relations $R \in \tau$ the relation $R\big((x_1,y_1),\ \dots,\ (x_k,y_k)\big)$ holds in $\Gamma_1 \times \Gamma_2$ iff $R(x_1,\dots,x_k)$ holds in Γ_1 and $R(y_1,\dots,y_k)$ holds in Γ_2. Homomorphisms from $\Gamma^k = \Gamma \times \cdots \times \Gamma$ to Γ are called *polymorphisms* of Γ. When R is a relation over the domain D, then we say that f *preserves* R (or that R is *closed under* f) if f is a polymorphism of $(D;R)$. Note that unary polymorphisms of Γ are endomorphisms of Γ. When ϕ is a first-order formula that defines R, and f preserves R, then we also say that f *preserves* ϕ.

The set of all polymorphisms $\mathrm{Pol}(\Gamma)$ of a relational structure forms an algebraic object called *clone* [26], which is a set of operations defined on a set D that is closed under composition and that contains all projections. The set $\mathrm{Pol}(\Gamma)$ is also locally closed, in the following sense. A set of functions $\mathcal{F}$ with domain D is *locally closed* if every function f with the following property belongs to $\mathcal{F}$: for every finite subset A of D there is some operation $g \in \mathcal{F}$ such that $f(a) = g(a)$ for all $a \in A^k$. We write $\overline{F}$ for the smallest set that is locally closed and contains F.

Polymorphism clones can be used to characterise primitive positive definability over a finite structure; this follows from results by Bodnarčuk, Kalužnin, Kotov, and Romov [15] and Geiger [20]. The characterisation remains true if the structure is ω-categorical.

▶ **Theorem 12** (Bodirsky & Nešetřil [9]). *Let Γ be an ω-categorical structure. Then the primitive positive definable relations in Γ are precisely the relations preserved by the polymorphisms of Γ.*

3.3 Model-complete cores

Let Γ and Δ be structures with relational signature τ. A homomorphism of Γ to Δ is said to be an *embedding* if it is injective and preserves $\neg R$ for all $R \in \tau$. The structure Γ is a *core* if all its endomorphisms are embeddings. Note that endomorphisms preserve existential positive formulas, and embeddings preserve existential formulas. The structure $(\mathbb{L}; C)$ for example is a core.

A first-order theory T is said to be *model-complete* if every embedding between models of T preserves all first-order formulas. A structure is called *model-complete* if its first-order theory is model-complete. The structure $(\mathbb{L}; C)$ is model-complete since it is even homogeneous. We say that two structures Γ and Δ are *homomorphically equivalent* if there exists a homomorphism from Γ to Δ, and one from Δ to Γ. Clearly, homomorphically equivalent structures have identical CSPs.

▶ **Theorem 13** (Bodirsky [2]). *Let Γ be an ω-categorical structure. Then Γ is homomorphically equivalent to an ω-categorical model-complete core Δ. The structure Δ is unique up to isomorphism, and again ω-categorical.*

Hence, we speak in the following of *the* model-complete core of an ω-categorical structure. The model-complete cores of reducts of $(\mathbb{L}; C)$ have been classified recently [5].

▶ **Theorem 14** (Bodirsky, Jonsson, & Pham [5]). *Let Γ be a reduct of $(\mathbb{L}; C)$, and Δ its model-complete core. Then one of the following applies.*

1. *Δ has just one element.*
2. *Δ is isomorphic to a reduct of $(\mathbb{L}; =)$.*
3. *$\Delta = \Gamma$ has the same endomorphisms as $(\mathbb{L}; Q)$ where Q is the relation defined by the formula given in Example 4 (the 'unrooted' situation).*
4. *$\Delta = \Gamma$ has the same endomorphisms as $(\mathbb{L}; C)$ (the 'rooted' situation).*

Define the relations

$$C_d := \{(x, y, z) \in \mathbb{L}^3 \ : \ x | yz \wedge y \neq z\}, \text{ and}$$
$$Q_d := \{(x, y, u, v) \in \mathbb{L}^4 \ : \ Q(x, y, u, v) \wedge x \neq y \wedge u \neq v\}.$$

The following result is a consequence of Theorem 14.

▶ **Lemma 15.** *Let Γ be a reduct of $(\mathbb{L}; C)$ which does not have a constant endomorphism and which is not homomorphically equivalent to an equality constraint language. Then Γ is a model-complete core, and C_d or Q_d are primitive positive definable in Γ.*

4 Affine Horn formulas

Recall that a Boolean relation R is called *affine* if it can be defined by a system of linear equation systems over the 2-element field. It is well-known (see e.g. [19]) that a Boolean relation is affine if and only if it is preserved by the function $(x, y, z) \mapsto x + y + z \pmod 2$.

▶ **Definition 16.** Let $B \subseteq \{0,1\}^n$ be a Boolean relation. Then $\phi_B(z_1, \ldots, z_n)$ stands for the formula

$$z_1 = \cdots = z_n \vee \bigvee_{t \in B \setminus \{(0,0,\ldots,0),(1,1,\ldots,1)\}} \{z_i : t_i = 0\} | \{z_i : t_i = 1\} \; .$$

The formula ϕ_B is called *affine* if $B \cup \{(0,0,\ldots,0),(1,1,\ldots,1)\}$ is affine.

▶ **Definition 17.** An *affine Horn clause* is a formula of the form $x_1 \neq y_1 \vee \cdots \vee x_n \neq y_n$ or of the form $x_1 \neq y_1 \vee \cdots \vee x_n \neq y_n \vee \phi(z_1, \ldots, z_k)$ where ϕ is an affine formula. An *affine Horn formula* is a conjunction of affine Horn clauses. A relation $R \subseteq \mathbb{L}^k$ is called *affine Horn* if it can be defined by an affine Horn formula over $(\mathbb{L}; C)$. A phylogeny constraint language is called *affine Horn* if all its relations are affine Horn.

▶ **Example 18.** The relation $\{(z_1, z_2, z_3, z_4) \in \mathbb{L}^4 : z_1 z_2 | z_3 z_4$ and $z_1 = z_2 \Leftrightarrow z_3 = z_4\}$ is affine Horn. To see this, first note that it can equivalently be defined by the formula

$$(z_1 z_2 | z_3 z_4 \vee z_1 = z_2 = z_3 = z_4) \wedge (z_1 \neq z_2 \vee z_3 = z_4) \wedge (z_3 \neq z_4 \vee z_1 = z_2) \wedge z_1 \neq z_3.$$

It is sufficient to verify that each conjunct is an affine Horn clause. We do this here for the first conjunct. Consider the relation $R = \{(0,0,0,0),(1,1,0,0),(0,0,1,1),(1,1,1,1)\}$, which is affine since $(z_1, z_2, z_3, z_4) \in R$ if and only if $z_1 + z_2 = 0 \pmod 2$ and $z_3 + z_4 = 0 \pmod 2$. We see that $\phi_R(z_1, z_2, z_3, z_4)$ is equivalent to $z_1 = z_2 = z_3 = z_4 \vee z_1 z_2 | z_3 z_4$.

The relation $N := \{(x, y, z) \in \mathbb{L}^3 : (xy|z \vee x|yz)\}$ has been called the *forbidden triple relation* by Bryant [16] and it plays an important role in the classification. Bryant showed that $\mathrm{CSP}(\mathbb{L}; N)$ is NP-complete. We are therefore particularly interested in those reducts Γ of $(\mathbb{L}; C)$ where $N \notin \langle\Gamma\rangle$. We will prove later that when Γ is a reduct of $(\mathbb{L}; C)$ with finite relational signature such that $C \in \langle\Gamma\rangle$ and $N \notin \langle\Gamma\rangle$, then $\mathrm{CSP}(\Gamma)$ is in P. The following result is the combinatorial heart of this paper.

▶ **Theorem 19.** *Let Γ be a reduct of $(\mathbb{L}; C)$ such that $C \in \langle\Gamma\rangle$ and $N \notin \langle\Gamma\rangle$. Then all relations in $\langle\Gamma\rangle$ are affine Horn.*

In the proof of Theorem 19 we use the algebraic approach in combination with a Ramsey theorem for trees, due to Leeb [23]; also see Bodirsky [4]. The outline is as follows: if the relation N is not primitive positive definable in Γ, there must be a polymorphism of Γ that does not preserve it, by Theorem 12. We apply Ramsey theory to prove that polymorphisms of Γ must behave *canonically* on large parts of the domain; the technique we use here is developed in a larger context by Bodirsky, Pinsker, and Tsankov [14]. The obtained polymorphisms in turn imply strong structural properties on the relations they preserve which can then be used to prove that all relations that are primitive positive definable in Γ are affine Horn.

▶ **Theorem 20.** *There is a polynomial-time algorithm that decides whether a given affine Horn formula is satisfiable over $(\mathbb{L}; C)$.*

We give a sketch of how the algorithm works. The key is a procedure which does the following: either it returns a solution where all variables take different values in $\mathbb{L}$ or it returns a set of variables that must take equal value in all solutions. If variables that are syntactically forced to be different are contracted, the algorithm rejects, and otherwise we find a solution after a linear number of variable contractions. The idea for the key procedure is as follows: we solve a particular affine Boolean equation system in order to

determine which variables will be mapped below the same child of the root in a solution to the instance. This can be done in polynomial time by Gaussian elimination. If there is no solution, the procedure returns all variables, and if there is a solution, it recursively proceeds with two sub-instances induced by the solution to the equation system.

▶ **Corollary 21.** *Let* Γ *be a reduct of* $(\mathbb{L}; C)$ *which is affine Horn and has a finite signature. Then* $\mathrm{CSP}(\Gamma)$ *can be solved in polynomial time.*

We can now prove Theorem 5.

Proof of Theorem 5. Let Γ be a reduct of $(\mathbb{L}; C)$ with finite relational signature and let Δ be the model-complete core of Γ. The structure Δ is homomorphically equivalent to Γ by Theorem 13 so $\mathrm{CSP}(\Gamma)$ and $\mathrm{CSP}(\Delta)$ have the same complexity. We need to consider four cases by Lemma 15.

(1) Δ has just one element and $\mathrm{CSP}(\Delta)$ is trivially in P.

(2) Δ is isomorphic to a reduct of $(\mathbb{L}; =)$ and $\mathrm{CSP}(\Delta)$ is either in P or NP-hard by Bodirsky & Kára [6].

(3) $C_d \in \langle\Delta\rangle$. It is easy to show that if $C_d \in \langle\Delta\rangle$, then $C \in \langle\Delta\rangle$, too. In this case, the complexity of $\mathrm{CSP}(\Delta)$ depends on whether $N \in \langle\Delta\rangle$ or not. If $N \in \langle\Delta\rangle$ then $\mathrm{CSP}(\Delta)$ is NP-hard (Bryant [16]) as discussed in Section 4. Otherwise, Theorem 19 implies that all relations in Γ are affine Horn and $\mathrm{CSP}(\Gamma)$ is in P by Corollary 21.

(4) $Q_d \in \langle\Delta\rangle$ and $\mathrm{CSP}(\Delta)$ is NP-hard due to Steel [25]. ◀

5 Affine tree operations

The border between NP-hardness and tractability for phylogeny problems can be stated in terms of polymorphisms. To characterize such polymorphisms, we introduce a certain kind of binary operations over $\mathbb{L}$ which we call *affine tree operations*. The syntactic characterization of affine Horn constraint languages (from Section 4) is convenient to work with when, for instance, constructing algorithms. However, it is not very convenient when studying polymorphisms. In this section, we construct an operation, called tx, such that every relation that is first order definable in $(\mathbb{L}; C)$ is preserved by tx if and only if it can be defined by an affine Horn formula. The operation tx is constructed as follows.

Let U, V be two finite subsets of $\mathbb{L}$. A function $f\colon \mathbb{L}^2 \to \mathbb{L}$ is called *perfectly dominated (by the first argument) on* $U \times V$ if the following conditions holds.

- For all $u_1, u_2, u_3 \in U$ and $v_1, v_2, v_3 \in V$ if $u_1|u_2u_3$ then $f(u_1, v_1)|f(u_2, v_2)f(u_3, v_3)$ and
- for all $u \in U$ and $v_1, v_2, v_3 \in V$ if $v_1|v_2v_3$ then $f(u, v_1)|f(u, v_2)f(u, v_3)$.

Let $f\colon \mathbb{L}^2 \to \mathbb{L}$ be an injective function, and U be a finite subset of $\mathbb{L}$. We inductively define whether f is *semidominated on* U^2 as follows. If $U = \emptyset$ or $|U| = 1$ then f is semidominated on $U \times U$. Otherwise, f is semidominated on $U \times U$ if there are $U_1, U_2 \subseteq U$ such that $U = U_1 \cup U_2$, $U_1|U_2$, and the following conditions hold.

- f is semidominated on $U_1 \times U_1$ and $U_2 \times U_2$;
- $f(U_1 \times U_1)|f(U_2 \times U_2)$ and $f(U_1 \times U_2)|f(U_2 \times U_1)$;
- $f((U_1 \times U_1) \cup (U_2 \times U_2))|f((U_1 \times U_2) \cup (U_2 \times U_1))$;
- $f(x, y)$ is perfectly dominated on $U_1 \times U_2$ and $f(y, x)$ is perfectly dominated on $U_2 \times U_1$.

We say that an operation $f\colon \mathbb{L}^2 \to \mathbb{L}$ is an *affine tree operation* if f is semidominated on $U \times U$ for every finite subset U of $\mathbb{L}$. We are now ready for the main result of this section.

▶ **Theorem 22.** *There exists an affine tree operation, which we call* tx*, and endomorphisms* e_1, e_2 *of* $(\mathbb{L}; C)$ *such that* $e_1(\mathrm{tx}(x,y)) = e_2(\mathrm{tx}(y,x))$ *and for every reduct* Γ *of* $(\mathbb{L}; C)$*, the following are equivalent:*

(1) Γ *is preserved by* tx*.*
(2) all relations in $\langle\Gamma\rangle$ *are affine Horn.*
(3) all relations in Γ *are affine Horn.*

The above theorem can be proved by the idea of the following lemma that comes from the proof of Proposition 6.6 in Bodirsky, Pinsker and Pongracz [13].

▶ **Lemma 23.** *Let* Δ *be* ω*-categorical, and* $f \in \mathrm{Pol}^{(2)}(\Delta)$*. Suppose that for every finite subset* A *of the domain* D *of* Δ *there exists an* $\alpha \in \mathrm{Aut}(\Delta)$ *such that* $f(x,y) = \alpha(f(y,x))$ *for all* $x, y \in A$*. Then there are* $e_1, e_2 \in \mathrm{Aut}(\Delta)$ *such that* $e_1(f(x,y)) = e_2(f(y,x))$ *for all* $x, y \in D$*.*

6 Main result

Our results are much stronger than the complexity classification from Theorem 5, though. We have a dichotomy for reducts of $(\mathbb{L}; C)$ which remains interesting even if P=NP, and which we view as a fundamental result not just in the context of constraint satisfaction. Our dichotomy can be phrased in various different but equivalent ways, using terminology from universal algebra and topology; we mention that there is also an equivalent formulation using primitive positive interpretability. We first introduce the necessary concepts, and then state how they are linked together in the strongest formulation of our results.

Let $\mathcal{C}$ and $\mathcal{D}$ denote two clones as defined in Section 3. A function $\xi\colon \mathcal{C} \to \mathcal{D}$ is called a *clone homomorphism* if it sends every projection in $\mathcal{C}$ to the corresponding projection in $\mathcal{D}$, and it satisfies the identity $\xi(f(g_1, \ldots, g_n)) = \xi(f)(\xi(g_1), \ldots, \xi(g_n))$ for all n-ary $f \in \mathcal{C}$ and all m-ary $g_1, \ldots, g_n \in \mathcal{C}$. Such a homomorphism ξ is *continuous* if the map ξ is continuous with respect to the topology of pointwise convergence, where the closed sets are precisely the sets that are locally closed as defined in Section 3. We write $\mathbf{1}$ for the clone on the set $\{0,1\}$ that only contains the projections and carries the discrete topology.

A binary polymorphism of Γ is called *symmetric modulo endomorphisms* if there are endomorphisms e_1 and e_2 of Γ such that $\forall x, y.\, e_1(f(x,y)) = e_2(f(y,x))$.

▶ **Theorem 24.** *Let* Γ *be a reduct of* $(\mathbb{L}; C)$*, and let* Δ *be the model-complete core of* Γ*. Then the following are equivalent.*

1. Δ *has a symmetric polymorphism modulo endomorphisms.*
2. *For all elements* $a_1, \ldots, a_n$ *of* Δ *there is no clone homomorphism from* $\mathrm{Pol}(\Delta, a_1, \ldots, a_n)$ *to* $\mathbf{1}$*.*
3. *For all elements* $a_1, \ldots, a_n$ *of* Δ *there is no continuous clone homomorphism from* $\mathrm{Pol}(\Delta, a_1, \ldots, a_n)$ *to* $\mathbf{1}$*.*
4. *For all elements* $a_1, a_2, \ldots, a_n$ *of* Δ *there is no primitive positive interpretation of* NAE *in* $(\Delta, a_1, a_2, \ldots, a_n)$*.*

If these conditions apply, $\mathrm{CSP}(\Gamma)$ *is in P, otherwise* $\mathrm{CSP}(\Gamma)$ *is NP-complete.*

Proof sketch. The implication (1) ⇒ (2) can be shown to hold in general for ω-categorical model-complete cores Δ and the implication (2) ⇒ (3) is trivial. The implication (3) ⇒ (4) follows from Theorem 28 in [12]. For the implication (4) ⇒ (1), we use the classification of Δ into four types from Theorem 14. For the first type, Δ has just one element and hence satisfies item 1. For the second type, the statement follows from results by Bodirsky

and Kára [6]; in fact, tx is a suitable polymorphism. For the third type, $Q_d \in \langle\Gamma\rangle$ by Lemma 15 and one can show that $Q \in \langle\Gamma\rangle$, too. Furthermore, NAE has a primitive positive definition in $(\mathbb{L}; Q, a_1, a_2, a_3)$ for arbitrary pairwise distinct constants $a_1, a_2, a_3 \in \mathbb{L}$ so NAE has a primitive positive definition in (Γ, a_1, a_2, a_3). By Theorem 28 in [12], there is a continuous clone homomorphism from $\mathrm{Pol}(\Gamma, a_1, a_2, a_3)$ to $\mathbf{1}$. We can disregard this case since it contradicts our basic assumption.

We now focus on the fourth type. It can be shown that in this case $C \in \langle\Gamma\rangle$ since $C_d \in \langle\Gamma\rangle$ by Lemma 15. If $N \in \langle\Gamma\rangle$, then NAE has a primitive positive definition in (N, a_1, a_2) where $a_1, a_2 \in \mathbb{L}$ are distinct constants. This contradicts (4). If $N \notin \langle\Gamma\rangle$, then Theorem 19 implies that every relation in $\langle\Gamma\rangle$ is affine Horn. By Theorem 22, tx is a binary commutative polymorphism modulo endomorphisms.

If items 1.—4. hold, then Δ has only one element or tx is a binary polymorphism of Γ. If the former holds, then $\mathrm{CSP}(\Gamma)$ is trivially in P. If the latter holds, then it follows from Theorems 20 and 22 that $\mathrm{CSP}(\Gamma)$ is in P. If items 1.—4. do not hold, then there is a clone homomorphisms from $(\Delta, a_1, \ldots, a_n)$ to $\mathbf{1}$ for some elements $a_1, \ldots, a_n \in \mathbb{L}$ and NAE has a primitive positive interpretation in an expansion of Γ with a finite number of constants. Thus, $\mathrm{CSP}(\Gamma)$ is NP-complete. ◀

The fact that the continuity condition in item 3 of Theorem 24 can simply be dropped in item 2 is remarkable. Indeed, we do not know whether there is an ω-categorical structure whose polymorphism clone homomorphically maps to $\mathbf{1}$, but not via a continuous clone homomorphism (see the discussion in [13]).

Suppose that Γ is a reduct of $(\mathbb{L}; C)$ with finite relational signature such that $C \in \langle\Gamma\rangle$. Then one might ask whether the *meta-problem* of deciding the complexity of $\mathrm{CSP}(\Gamma)$ is effective. Here we assume that Γ is given via quantifier-free first-order definitions of its relations in $(\mathbb{L}; C)$. We can then use the techniques developed by Bodirsky, Pinsker, and Tsankov [14] to effectively test whether the relation N is in $\langle\Gamma\rangle$. Thus, the meta-problem for phylogeny problems is decidable.

Acknowledgements

The first and third author have received funding from the European Research Council under the European Community's Seventh Framework Programme (FP7/2007-2013 Grant Agreement no. 257039.) The second author is partially supported by the Swedish Research Council (VR) under grant 621-2012-3239.

References

1 Alfred V. Aho, Yehoshua Sagiv, Thomas G. Szymanski, and Jeffrey D. Ullman. Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions. *SIAM Journal on Computing*, 10(3):405–421, 1981.

2 Manuel Bodirsky. Cores of countably categorical structures. *Logical Methods in Computer Science*, 3(1):1–16, 2007.

3 Manuel Bodirsky. Complexity classification in infinite-domain constraint satisfaction. Mémoire d'habilitation à diriger des recherches, Université Diderot – Paris 7. Available at arXiv:1201.0856, 2012.

4 Manuel Bodirsky. Ramsey classes: Examples and constructions. To appear in the Proceedings of the 25th British Combinatorial Conference; arXiv:1502.05146, 2015.

5 Manuel Bodirsky, Peter Jonsson, and Trung Van Pham. The reducts of the homogeneous binary branching C-relation. Preprint arXiv:1408.2554, 2014.

6 Manuel Bodirsky and Jan Kára. The complexity of equality constraint languages. *Theory of Computing Systems*, 3(2):136–158, 2008. A conference version appeared in the proceedings of Computer Science Russia (CSR'06).

7 Manuel Bodirsky and Jan Kára. The complexity of temporal constraint satisfaction problems. *Journal of the ACM*, 57(2):1–41, 2009. An extended abstract appeared in the Proceedings of the Symposium on Theory of Computing (STOC'08).

8 Manuel Bodirsky and Jens K. Mueller. Rooted phylogeny problems. *Logical Methods in Computer Science*, 7(4), 2011. An extended abstract appeared in the proceedings of ICDT'10.

9 Manuel Bodirsky and Jaroslav Nešetřil. Constraint satisfaction with countable homogeneous templates. *Journal of Logic and Computation*, 16(3):359–373, 2006.

10 Manuel Bodirsky and Michael Pinsker. Reducts of Ramsey structures. *AMS Contemporary Mathematics, vol. 558 (Model Theoretic Methods in Finite Combinatorics)*, pages 489–519, 2011.

11 Manuel Bodirsky and Michael Pinsker. Schaefer's theorem for graphs. In *Proceedings of the Annual Symposium on Theory of Computing (STOC)*, pages 655–664, 2011. Preprint of the long version available at arxiv.org/abs/1011.2894.

12 Manuel Bodirsky and Michael Pinsker. Topological Birkhoff. *Transactions of the American Mathematical Society*, 367:2527–2549, 2015.

13 Manuel Bodirsky, Michael Pinsker, and András Pongrácz. Projective clone homomorphisms. Preprint, available at ArXiv:1409.4601, 2014.

14 Manuel Bodirsky, Michael Pinsker, and Todor Tsankov. Decidability of definability. *Journal of Symbolic Logic*, 78(4):1036–1054, 2013. A conference version appeared in the Proceedings of LICS 2011.

15 V. G. Bodnarčuk, Lev A. Kalužnin, Victor Kotov, and Boris A. Romov. Galois theory for Post algebras, part I and II. *Cybernetics*, 5:243–539, 1969.

16 David Bryant. Building trees, hunting for trees, and comparing trees. PhD-thesis at the University of Canterbury, 1997.

17 Andrei A. Bulatov, Andrei A. Krokhin, and Peter G. Jeavons. Classifying the complexity of constraints using finite algebras. *SIAM Journal on Computing*, 34:720–742, 2005.

18 Peter J. Cameron. *Oligomorphic permutation groups*. Cambridge University Press, Cambridge, 1990.

19 Hubie Chen. A rendezvous of logic, complexity, and algebra. *SIGACT News*, 37(4):85–114, 2006.

20 David Geiger. Closed systems of functions and predicates. *Pacific Journal of Mathematics*, 27:95–100, 1968.

21 Wilfrid Hodges. *Model theory*. Cambridge University Press, 1993.

22 Peter Jeavons. On the algebraic structure of combinatorial problems. *Theoretical Computer Science*, 200:185–204, 1998.

23 Klaus Leeb. *Vorlesungen über Pascaltheorie*, volume 6 of *Arbeitsberichte des Instituts für Mathematische Maschinen und Datenverarbeitung*. Friedrich-Alexander-Universität Erlangen-Nürnberg, 1973.

24 Meei Pyng Ng, Mike Steel, and Nicholas C. Wormald. The difficulty of constructing a leaf-labelled tree including or avoiding given subtrees. *Discrete Applied Mathematics*, 98:227–235, 2000.

25 Michael Steel. The complexity of reconstructing trees from qualitative charaters and subtrees. *Journal of Classification*, 9:91–116, 1992.

26 Ágnes Szendrei. *Clones in universal algebra*. Séminaire de Mathématiques Supérieures. Les Presses de l'Université de Montréal, 1986.