When Darwin Met Ianus: Dichotomies of Expressivity

Johanna Brunar
TU Wien, Austria

Michael Pinsker
TU Wien, Austria

Moritz Schöbi
TU Wien, Austria

Abstract

The classifications of temporal and phylogeny constraint languages stand among the most seminal complexity classifications within infinite-domain Constraint Satisfaction Problems (CSPs), yet remain the most mysterious in terms of algorithms and algebraic invariants for the tractable cases. We show that those languages which do not pp-construct EVERYTHING (and thus by the classifications are solvable in polynomial time) have, in fact, very limited expressive power as measured by the graphs and hypergraphs they can pp-interpret. This limitation yields many previously unknown algebraic consequences, while also providing new, uniform proofs for known invariance properties. In particular, we show that such temporal and phylogeny constraint languages admit 4-ary pseudo-Siggers polymorphisms – a result that sustains the possibility that the existence of such polymorphisms extends to the much broader context of the Bodirsky-Pinsker conjecture. Although temporal and phylogeny constraint languages appear to follow fundamentally different algorithmic principles, our proofs reveal a common core and proceed along strikingly similar lines.

When Ianus can't express it all
He tries in vain, he hits a wall
When for $\mathbb{K}_3$ no way he knows
His face of pseudo-loops he shows.

As Darwin finds such twisted edge
To pines and vines he makes this pledge:
"Should free of pseudo-loops you shine
All finite structures shall be thine!"

2012 ACM Subject Classification Theory of computation → Logic; Theory of computation → Computational complexity and cryptography

Keywords and phrases Constraint Satisfaction Problem (CSP), Temporal Constraint language, Phylogeny Constraint language, pseudo-loop, primitive positive interpretation, polymorphism, oligomorphic permutation group, identity

Digital Object Identifier 10.4230/LIPIcs.MFCS.2026.100

Related Version *Full Version*: `https://arxiv.org/abs/2509.04347v4`

Funding Funded by the European Union (ERC, POCOCOP, 101071674). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. This research was funded in whole or in part by the Austrian Science Fund (FWF) [I5948]. For the purpose of Open Access, the authors have applied a CC BY public copyright licence to any Author Accepted Manuscript (AAM) version arising from this submission. The first author is a recipient of a DOC Fellowship of the Austrian Academy of Sciences at Technische Universität Wien.

The Roman god Janus (Latin: *Ianus*), traditionally depicted with two faces looking in opposite directions, embodies the duality of temporal perspective. With his theory of evolution, Charles Darwin (1809–1882) laid the foundation for modern phylogenetics.

1 Introduction

The *Constraint Satisfaction Problem* induced by a relational structure $\mathbb{A}$, denoted by CSP($\mathbb{A}$), is the computational problem of deciding, given a finite input structure $\mathbb{B}$ of the same signature as $\mathbb{A}$, whether there exists a homomorphism from $\mathbb{B}$ to $\mathbb{A}$, i.e. a map preserving all relations. The underlying structure $\mathbb{A}$ will often be referred to as *template* structure. This notion of fixed-template CSPs provides a uniform framework for modelling many classical computational problems such as graph-colouring problems, 3-SAT, or solving equations. In 2017, two independent confirmations of a complexity dichotomy for CSPs induced by arbitrary finite structures – conjectured already in [48, 49] – marked a breakthrough in the research programme on the complexity of finite-domain CSPs: for every finite structure $\mathbb{A}$, either CSP($\mathbb{A}$) is solvable in polynomial time, or it is NP-complete [45, 71, 72], contrasting Ladner's theorem [62]. Yet, prominent computational problems that can be phrased as the CSPs of a fixed template $\mathbb{A}$ require the domain of $\mathbb{A}$ to be of countably infinite size – for example, the problem of deciding whether a given finite digraph contains a cycle can be phrased as CSP over $(\mathbb{Q}; <)$, but cannot be modelled in this way by any finite template. There provably does not exist a complexity dichotomy for CSPs with infinite templates, not even if the template is "close to finite" in the sense of *ω-categoricity* [16, 51, 52]. For certain countably infinite ω-categorical structures, namely *first-order reducts of finitely bounded homogeneous* structures, though, a P/NP-complete complexity dichotomy has been conjectured by Bodirsky and Pinsker more than a decade ago (see [38, 10, 4, 3] for various formulations of the conjecture). The conjecture remains wide open in its generality, but has been verified for several classes of structures fitting into the conjectured framework, including temporal constraint languages [25, 26], phylogeny CSPs [22, 23], equality constraint languages [24], the universal homogeneous poset [59, 60], MMSNP [27, 28], first-order reducts of any homogeneous undirected graph [30] including the random graph [35], the universal homogeneous tournament [65, 66], and graph orientation problems with forbidden tournaments [17, 12, 50]. It is worth noting that with the exception of temporal and phylogeny constraint languages, all known complexity classifications can be obtained in terms of a general procedure from [31], which relies on *canonical polymorphisms* [39, 37] to show polynomial-time solvability of an infinite-domain CSP by reducing the problem to the CSP of a finite template. From an algorithmic perspective, temporal and phylogeny constraint languages therefore appear fundamentally different from all other classes for which the Bodirsky-Pinsker conjecture has been verified.

Temporal constraint languages are structures of the form $(\mathbb{Q}; R_1, R_2, \dots)$ where each R_i has a first-order definition in $(\mathbb{Q}; <)$ – the rational numbers with the dense linear order. Their complexity classification [25, 26], which predates the Bodirsky-Pinsker conjecture and most likely inspired it, is among the earliest within its scope. To this day, temporal constraint languages remain an active subject of study. They give rise to a natural and important class of CSPs, appearing in prominent problems from artificial intelligence such as temporal and spatial reasoning, see e.g. [56, 41, 61, 47, 20]. Yet despite their practical importance, temporal CSPs remain among the most challenging and least understood templates within the scope of the Bodirsky–Pinsker conjecture. This is not only due to the algorithmic challenges outlined above, but also because their descriptive complexity differs vastly from the finite [40]. Moreover, the algebraic properties of temporal constraint languages – crucial to the success of the so-called *algebraic approach* to CSPs – are still largely unknown. While ongoing research aims to gain deeper insight into the cases that are solvable in polynomial time through new algorithmic approaches [64], our contribution is to shed light on the algebraic aspects.

Phylogeny problems arise from evolutionary biology. In a (rooted) *phylogenetic tree*, every species has exactly one direct ancestor, and any two share a youngest common ancestor. Given three leaves a, b and c of such a tree, i.e., three species without descendants, we write $a|bc$ to denote that the youngest common ancestor of b and c lies below the youngest common ancestor of a, b and c in the tree. The set of leaves together with the ternary relation $|$ is called the *leaf structure* of the tree. The *basic phylogeny decision problem* asks whether, given a finite relational structure with one ternary relation, there exists a tree to whose leaf structure it maps to homomorphically. This problem can be modelled as the CSP over the infinite-domain *generic leaf structure* $(\mathbb{L}; |)$ [21], and is solvable in quadratic time [1]. Moreover, all CSPs induced by *phylogeny constraint languages*, i.e. structures of the form $(L; R_1, R_2, \dots)$ where each R_i has a first-order definition in $(\mathbb{L}; |)$, are known to be either solvable in polynomial time or NP-complete [32]. However, as the standard reduction to the finite via canonical polymorphisms is again inapplicable, phylogeny together with temporal constraint languages remain, in a sense, the main troublemakers in the infinite.

1.1 The algebraic approach

The complexity of finite-domain CSPs is determined by the "expressive power" of the underlying structure $\mathbb{A}$, which is encoded in the set $\mathrm{Pol}(\mathbb{A})$ of all compatible finitary operations – the *polymorphism clone* of $\mathbb{A}$. The study of the computational complexity of CSPs via the polymorphisms they have is what is now understood as the algebraic approach to CSPs, first developed in [55, 46] and refined by [9]. The finite-domain CSP dichotomy now takes the following form: as the only source of NP-completeness for $\mathrm{CSP}(\mathbb{A})$ stands the ability of $\mathbb{A}$ to *pp-construct* EVERYTHING, i.e. every finite structure. For the purpose of this paper, we call a structure that pp-constructs EVERYTHING *omni-expressive*. Every finite structure that lacks this property gives rise to a CSP solvable in polynomial time, and its polymorphism clone satisfies certain symmetries. Here, we say that the polymorphism clone $\mathrm{Pol}(\mathbb{A})$ of a structure $\mathbb{A}$ *satisfies* an identity, if the identity is witnessed by operations contained in $\mathrm{Pol}(\mathbb{A})$ for all evaluations of their arguments. For example, the polymorphism clone of a finite structure that is not omni-expressive always contains

- a 6-ary polymorphism s witnessing the 6-*ary Siggers identity* [70]

$$s(x, y, x, z, y, z) \approx s(y, x, z, x, z, y), \tag{1}$$

- a 4-ary polymorphism s witnessing the 4-*ary Siggers identity* [58]

$$s(a, r, e, a) \approx s(r, a, r, e), \tag{2}$$

- a 6-ary polymorphism o witnessing the *Olšák identities* [67]

$$o(x, x, y, y, y, x) \approx o(x, y, x, y, x, y) \approx o(y, x, x, x, y, y), \tag{3}$$

- for some $k \geq 3$, a k-ary polymorphism w witnessing the *weak near unanimity (WNU) identities* [63]

$$w(y, x, x, \dots, x) \approx w(x, y, x, \dots, x) \approx \cdots \approx w(x, x, \dots, x, y), \tag{4}$$

- for some $k \geq 3$, a k-ary polymorphism c witnessing the *cyclic identity* [6]

$$c(x_1, x_2, \dots, x_k) \approx c(x_2, \dots, x_k, x_1). \tag{5}$$

An algebraic approach to CSPs via polymorphisms is for countably infinite structures possible in the setting of *ω-categoricity*, where their automorphism group is *oligormorphic*, i.e. has only finitely many orbits in its action on tuples of any fixed finite length (although polymorphisms have also played a role for CSPs of non-omega-categorical structures, see e.g. [15, 14, 29, 18, 19]). Indeed, if $\mathbb{A}$ is ω-categorical, then the complexity of CSP($\mathbb{A}$) is again captured within the "local" algebraic structure of Pol($\mathbb{A}$) [36]. Moreover, for first-order reducts of finitely bounded homogeneous structures $\mathbb{A}$ – which are always ω-categegorical – it is conjectured that omni-expressivity remains the only source of NP-completeness of CSP($\mathbb{A}$) [38, 9, 4, 3]. Naturally, the question arises whether the identities (1)-(5) have counterparts for ω-categorical structures that are not omni-expressive. This was answered affirmatively in [10, 11] for a *pseudo-version* of identity (1): every ω-categorical structure that does not pp-construct EVERYTHING has polymorphisms s, u, and v witnessing the identity

$$u \circ s(x, y, x, z, y, z) \approx v \circ s(y, x, z, x, z, y). \tag{6}$$

The pseudo-versions of identities (2)–(5) are defined analogously, by composing each operation symbol in the original identities with a new unary operation symbol, where several occurrences of the same symbol are composed with different unary symbols. The corresponding statements for the pseudo-versions of both the cyclic identities (5) and the weaker WNU identities (4) are known to be false in general (for temporal structures that are not omni-expressive and fail to admit pseudo-cyclic polymorphisms, see e.g. [13, Proposition 12.9.1]; an example of an ω-categorical not omni-expressive structure with no pseudo-WNU polymorphisms is given in [2, Theorem 4]). It is, however, worth noting that the condition of satisfying a WNU or a cyclic identity differs fundamentally from the condition of satisfying the identities (1), (2), or (3) in that the former is in fact defined by an infinite disjunction of k-ary formulae for every $k \geq 3$, while the latter is a single formula. To the best of the authors' knowledge, no ω-categorical structures are known that are not omni-expressive and do not admit 4-ary pseudo-Siggers or pseudo-Olšák polymorphisms, leaving open the possibility that the existence of these polymorphisms may in fact characterise non-omni-expressivity for ω-categorical structures – a question highlighted in [13, Question 22]. Several indications point towards a positive answer: it is known [38, Proposition 6.6] that the pseudo-versions of all sets of identities that characterise non-omni-expressivity in the finite carry over in the case of ω-categorical not omni-expressive structures that adhere to the standard reduction from [31]. Temporal constraint languages and phylogeny CSPs are the only completely classified classes within the scope of the Bodirsky-Pinsker conjecture that do not conform to this reduction. As such, they currently represent the only known candidates for counterexamples within this range. Moreover, every *conservative* ω-categorical structure that is not omni-expressive admits 4-ary pseudo-Siggers polymorphisms [43]. Finally, in a somewhat different direction, the absence of pseudo-Olšák polymorphisms is known to imply NP-completeness [64] for ω-categorical structures (though here the NP-hardness is not known to stem from omni-expressivity).

Despite substantial progress in the general theory of CSPs since their classifications, the algebraic invariants underlying temporal and phylogeny templates that are not omni-expressive have remained poorly understood. Prior to this work, the identities known to be satisfied in these settings were those inherited from general results – the existence of 6-ary pseudo-Siggers polymorphisms [10, 11] and, assuming $P \neq NP$, pseudo-Olšák polymorphisms [64]. In addition, for temporal constraint languages, the existence of pseudo-WNU polymorphisms of all arities $k \geq 3$ [40, Proposition 7.27], and for phylogeny constraint languages, a binary pseudo-symmetric polymorphism had been established [23]. In this paper, we provide a

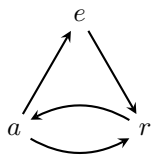

Figure 1 The 4-ary Siggers digraph.

uniform framework that not only captures the existence of the aforementioned polymorphisms, but also establishes a whole new family of identities satisfied by all temporal and all phylogeny constraint languages that are not omni-expressive. Among these, the existence of 4-ary pseudo-Siggers polymorphisms stands out as a previously unknown representative.

1.2 Loop lemmata

In the early stages of the systematic research programme on CSPs, those induced by finite graphs were among the first to be studied. Observe that the classical k-colouring problem coincides with $\mathrm{CSP}(\mathbb{K}_k)$, where $\mathbb{K}_k$ denotes the clique on k vertices. Vastly generalising the NP-completeness of the k-colouring problem for all $k \geq 3$, Hell and Nešetřil showed that every undirected non-bipartite graph is either omni-expressive (and hence its CSP is NP-complete), or has a loop (in which case it is entirely inexpressive, and in particular, its CSP trivial) [54]. This result was later extended to certain digraphs:

▶ **Theorem 1** ([8]). *Let $\mathbb{G}$ be a finite smooth digraph of algebraic length 1. Either $\mathbb{G}$ is omni-expressive or $\mathbb{G}$ contains a loop.*

As first observed in [70], loops in digraphs correspond to algebraic invariants. Given a finite digraph $\mathbb{G}$ and an enumeration $(i_1, j_1), (i_2, j_2), \ldots, (i_m, j_m)$ of its edges, the identity

$$s(x_{i_1}, \ldots, x_{i_m}) \approx s(x_{j_1}, \ldots, x_{j_m})$$

is called the $\mathbb{G}$-*loop condition*; in particular, the 6-ary Siggers identity (1) is the $\mathbb{K}_3$-loop condition, and the 4-ary Siggers identity (2) is the loop condition induced by the digraph in Figure 1. By standard techniques, Theorem 1 implies that the polymorphism clone of every not omni-expressive finite structure satisfies, in particular, both Siggers loop conditions.

▶ **Corollary 2** ([58]). *Let $\mathbb{A}$ be a finite relational structure that is not omni-expressive. If $\mathbb{G}$ is any finite smooth digraph of algebraic length 1, then $\mathrm{Pol}(\mathbb{A})$ satisfies the $\mathbb{G}$-loop condition.*

The statement of Theorem 1 fails badly for ω-categorical graphs (for example, the countably infinite complete graph has no loop, yet its CSP is solvable in polynomial time), as do in general the exact identities (1)-(5) in ω-categorical structures that are not omni-expressive. However, it has proven fruitful to consider the quotient of $\mathbb{G}$ modulo an oligomorphic subgroup Ω of its automorphism group, whose vertices are the Ω-orbits, and whose edges are induced from $\mathbb{G}$. A loop in this finite digraph comes from an edge in $\mathbb{G}$ between two vertices belonging to the same Ω-orbit – a so-called *pseudo-loop modulo* Ω. The absence of a pseudo-loop in $\mathbb{G}$ modulo its own automorphism group, or modulo an oligomorphic subgroup thereof, is a stronger condition than the absence of a loop, and has been identified as a source of computational hardness for $\mathrm{CSP}(\mathbb{G})$ in several cases [11, 2, 43]. On the other hand, while not directly implying tractability, the presence of pseudo-loops in ω-categorical digraphs can be used to derive symmetries in the form of pseudo-identities, similarly as in Corollary 2 (see [53]).

2 Our contribution

Following this direction, we prove a variant of Theorem 1 for the case of digraphs $\mathbb{G}$ that are *pp-interpretable* in a non-omni-expressive constraint language $\mathbb{A}$ that is temporal or phylogenetic. Roughly, this means that $\mathbb{G}$ is expressible in primitive positive logic over $\mathbb{A}$. Since the structures $(\mathbb{Q};<)$ and $(\mathbb{L};|)$ are ω-categorical, their automorphism groups are oligomorphic. The latter naturally act on $\mathbb{A}$ by automorphisms, and in fact also on $\mathbb{G}$ by the definition of a (pp-)interpretation. Hence, we may regard $\mathrm{Aut}((\mathbb{Q};<))$ or $\mathrm{Aut}((\mathbb{L};|))$, respectively, also as an oligomorphic subgroup of $\mathrm{Aut}(\mathbb{G})$.

▶ **Theorem 3.** *Let $\mathbb{A}$ be a temporal (phylogeny) constraint language that is not omni-expressive. If $\mathbb{G}$ is any smooth digraph that is pp-interpretable in $\mathbb{A}$ and has pseudo-algebraic length 1 modulo $\mathrm{Aut}((\mathbb{Q};<))$ (modulo $\mathrm{Aut}((\mathbb{L};|))$), then $\mathbb{G}$ contains a pseudo-loop modulo $\mathrm{Aut}((\mathbb{Q};<))$ (modulo $\mathrm{Aut}((\mathbb{L};|))$).*

Theorem 3 reveals the dichotomous nature of temporal and phylogeny constraint languages in terms of their expressivity. Namely, any such language is either omni-expressive or inexpressive based on the digraphs it can pp-interpret; in the latter case, the only pp-interpretable digraphs are those containing pseudo-loops. This loss of expressivity comes with a gain of algebraic invariants. The *$\mathbb{G}$-pseudo-loop condition* induced by a finite digraph $\mathbb{G}$ arises from the $\mathbb{G}$-loop condition by composing either side of the identity with unary function symbols u, v as before. From Theorem 3, we derive an infinite family of identities that is satisfied in every temporal or phylogeny constraint language that is not omni-expressive. Among these, we conclude the previously unknown existence of 4-ary pseudo-Siggers polymorphisms.

▶ **Corollary 4.** *Let $\mathbb{A}$ be a temporal (phylogeny) constraint language that is not omni-expressive. If $\mathbb{G}$ is any finite smooth digraph of algebraic length 1, then $\mathrm{Pol}(\mathbb{A})$ satisfies the $\mathbb{G}$-pseudo-loop condition. In particular, $\mathrm{Pol}(\mathbb{A})$ satisfies $u \circ s(a,r,e,a) \approx v \circ s(r,a,r,e)$.*

Corollary 4 confirms that the full range of loop conditions whose satisfaction is known to characterise non-omni-expressivity in the finite setting lifts, in their pseudo-versions, to all not omni-expressive temporal and to all not omni-expressive phylogeny structures. This crucially rules out these structures as counterexamples, reinforcing the broader validity of these characterisations. Moreover, Theorem 3 yields a uniform proof for the validity of all such loop conditions. We remark that the use of Theorem 3 to derive Corollary 4 is local in nature: it produces the polymorphisms on finite sets and then applies a standard compactness argument. This is necessary since, provably, no term operation over $(\mathbb{Q};<)$ and the binary operation $\ell\ell$ (whose existence as a polymorphisms obstructs omni-expressivity) can satisfy any non-trivial pseudo-identity.

As it turns out, sets of identities of the form (4) and (5) are better suited for deriving further structural properties that may, in particular, be used in algorithms [5, 7, 44, 71, 72, 65, 66, 57, 69]. They are formalised through the notion of a *$\mathbb{T}$-loop condition* for hypergraphs $\mathbb{T}$ of arity $n \geq 3$, whose hyperedges are ordered tuples. The $\mathbb{T}$-loop condition is then defined as a set of $n-1$ identities as in Section 3.3; Extending our approach to higher arities, we prove a version of Theorem 3 for the case of hypergraphs pp-interpretable in a not omni-expressive temporal or phylogeny template. A *pseudo-loop* in an n-ary hypergraph is a hyperedge all of whose coordinates belong to the same orbit; a hypergraph is *cyclic* if its hyperedges are invariant under cyclic permutations, it is *2-transitive* if it is invariant under the action of some 2-transitive subgroup of the symmetric group on n elements, and it is *symmetric* if it is invariant under all permutations.

▶ **Theorem 5.** *Let $\mathbb{T}$ be a hypergraph of arity $n \geq 2$ that is pp-interpretable in a non-omni-expressive constraint language $\mathbb{A}$.*

1. *If $\mathbb{A}$ is a temporal constraint language, $n \geq 3$, and $\mathbb{T}$ is cyclic and 2-transitive, then $\mathbb{T}$ contains a pseudo-loop modulo* $\mathrm{Aut}((\mathbb{Q}; <))$.
2. *If $\mathbb{A}$ is a phylogeny constraint language and $\mathbb{T}$ is symmetric, then $\mathbb{T}$ contains a pseudo-loop modulo* $\mathrm{Aut}((\mathbb{L}; |))$.

The *pseudo-loop condition* induced by a hypergraph is again obtained by composing all identities of the corresponding loop condition with unary function symbols. Since the hyperedges of the hypergraphs inducing the Olšák- and the WNU-identities are invariant under all permutations, they satisfy in particular the imposed symmetry conditions; hence, the existence of pseudo-Olšák polymorphisms and pseudo-WNU polymorphisms of all arities $n \geq 3$ now follows from Theorem 5 in the standard way.

Is 2-transitivity the new cyclicity?

Let us remark that the conclusion of Item 1 in Theorem 5 clearly fails in the case $n = 2$, as it would amount to the existence of a binary polymorphism that is pseudo-commutative – yet temporal constraint languages preserved by the binary operations $\ell\ell$ or mi are neither omni-expressive, nor must they admit such polymorphisms. In the phylogenetic setting, on the other hand, binary pseudo-commutative polymorphisms exist in all non-omni-expressive structures [23].

It might be of interest to point out that loop lemmata involving symmetry conditions provided by non-trivial group actions have also been established for finite hypergraphs of arity $n \geq 3$ (see [6, 73, 42, 74]). Most influentially, the study of finite cyclic hypergraphs in [6] stands as a landmark result in this context. In the temporal setting, cyclicity by itself is not a sufficiently strong condition: as mentioned before, there exist not-omni expressive temporal constraint languages that do not admit any pseudo-cyclic polymorphisms. A concrete example is the structure $(\mathbb{Q}; <, x \neq y \vee u = v)$ [13, Section 12.9.1], highlighting the necessity of the 2-transitivity condition on top of the cyclicity condition in Item 1 of Theorem 5. Observe that these symmetry conditions are proper in the sense that a hypergraph may be cyclic and 2-transitive without being fully symmetric. For example, for odd $n \geq 5$ the alternating group A_n is a proper 2-transitive subgroup of the full symmetric group S_n which contains all cyclic permutations. Whether or not cyclicity is truly a necessary condition in Item 1 of Theorem 5 remains an interesting open question for future work. For hypergraphs of arity $n = 3$, it is readily implied by 2-transitivity, whereas for all arities $n > 3$, it is not. Moreover, it remains open whether or not the symmetry condition for the phylogenetic setting in Item 2 may be relaxed to match the ones in the temporal setting.

3 Preliminaries

3.1 Model-theoretic

For $n \in \mathbb{N}$, we denote the set $\{1, \ldots, n\}$ by $[n]$. For an n-tuple $a = (a_1, \ldots, a_n)$, we write a_i or $\mathrm{pr}_i(a)$ for its i-th coordinate. By $\ker(a)$ we denote the subset of $[n] \times [n]$ consisting of all pairs (i, j) for which $a_i = a_j$. A *relation* R on a set A is a subset $R \subseteq A^n$ for some $n \in \mathbb{N}$, which is referred to as the *arity* of R. For a subset I of $[n]$, we denote by $\mathrm{pr}_I(R)$ the $|I|$-ary relation obtained by projecting R to all of its coordinates i with $i \in I$. For the purpose of this paper, we assume all relations considered to be non-empty. A *(relational) structure* is a tuple $\mathbb{A} = (A; \mathcal{R})$ consisting of a set A and a finite family $\mathcal{R}$ of relations on A

indexed by *relational symbols*. For structures $\mathbb{A} = (A; \mathcal{R})$ and $\mathbb{B} = (B; \mathcal{R}')$ indexed the same relational symbols, a *homomorphism* from $\mathbb{A}$ to $\mathbb{B}$ is a mapping $f : A \to B$ such that for every relation $R \in \mathcal{R}$ and every $(a_1, \ldots, a_n) \in R$, the tuple $(f(a_1), \ldots, f(a_n))$ is contained in the corresponding relation of $\mathcal{R}'$. The structures $\mathbb{A}$ and $\mathbb{B}$ are *homomorphically equivalent* if there exist homomorphisms both from $\mathbb{A}$ to $\mathbb{B}$, and from $\mathbb{B}$ to $\mathbb{A}$. An *automorphism* of $\mathbb{A}$ is a bijective homomorphism $f : A \to A$ whose inverse mapping f^{-1} is also a homomorphism. By $\mathrm{Aut}(\mathbb{A})$ we denote the automorphism group of $\mathbb{A}$. In the following, by componentwise application we will understand all mappings also as functions on n-tuples.

A first-order formula ϕ is called *primitive positive (pp)* over a set consisting of relational symbols $R_1, \ldots, R_n$ if it involves only the predicates R_i, equality, existential quantification, and conjunction. A relation is *pp-definable* in a structure $\mathbb{A} = (A; \mathcal{R})$ if it is definable by a pp-formula over the relational symbols from $\mathcal{R}$. A structure $\mathbb{B}$ is pp-definable in $\mathbb{A}$ if all of its relations are. We say that $\mathbb{A}$ *pp-interprets* $\mathbb{B}$ if there exist $k \geq 1$ and a partial surjective map $h : A^k \to B$ with the property that for every relation $R \subseteq B^n$ that is either a relation of $\mathbb{B}$, the equality relation on B, or B, its preimage $h^{-1}(R)$ – regarded as a relation of arity nk on A – is pp-definable in $\mathbb{A}$. If $\mathbb{A}$ pp-interprets $\mathbb{B}$, then $\mathrm{CSP}(\mathbb{B})$ is log-space reducible to $\mathrm{CSP}(\mathbb{A})$ [46]. Moreover, it is not hard to see that $\mathrm{Aut}(\mathbb{A})$ naturally acts as a subgroup of $\mathrm{Aut}(\mathbb{B})$; this action is oligomorphic if its original action is. We say that $\mathbb{A}$ *pp-constructs* $\mathbb{B}$ if $\mathbb{B}$ is homomorphically equivalent to a structure that is pp-interpretable in $\mathbb{A}$. As homomorphically equivalent structures have the same CSPs, this also implies a log-space reduction from $\mathrm{CSP}(\mathbb{B})$ to $\mathrm{CSP}(\mathbb{A})$. We say that $\mathbb{A}$ pp-constructs *EVERYTHING* if $\mathbb{A}$ pp-constructs every finite structure; we then call $\mathbb{A}$ *omni-expressive*. Clearly, the CSP induced by an omni-expressive structure is NP-hard. By the finite-domain CSP dichotomy [45, 71, 72], the CSP of a finite structure that is not omni-expressive is always solvable in polynomial time.

A permutation group Ω acting on a set A induces an equivalence relation on A: two elements $a, b \in A$ are equivalent if there is $\alpha \in \Omega$ such that $\alpha(a) = b$. The corresponding equivalence classes are called the *orbits* of $\Omega \curvearrowright A$, or simply Ω-*orbits*. By componentwise evaluation, Ω acts on A^k for every $k \in \mathbb{N}$. The group is *oligomorphic* if this action has finitely many orbits for every $k \in \mathbb{N}$. A countable structure $\mathbb{A}$ is ω-*categorical* if $\mathrm{Aut}(\mathbb{A})$ is oligomorphic. We call a structure $\mathbb{B}$ a *(first-order) reduct* of $\mathbb{A}$ if its domain is A and its relations are first-order definable over the relational symbols of $\mathbb{A}$. Note that if $\mathbb{B}$ is a reduct of $\mathbb{A}$, then $\mathrm{Aut}(\mathbb{A}) \subseteq \mathrm{Aut}(\mathbb{B})$. It follows that reducts of ω-categorical structures are ω-categorical.

3.2 Graph-theoretic

A *digraph* is a relational structure of the form $\mathbb{G} = (G; E)$, where E is binary. $\mathbb{G}$ is called *smooth* if $\mathrm{pr}_1(E) = \mathrm{pr}_2(E)$. In this case, we also write $\mathrm{supp}(\mathbb{G})$ for the set $\mathrm{pr}_1(E)$. If E is symmetric, $\mathbb{G}$ is also called a *graph*. For $m \in \mathbb{N}$, we denote by E^m the binary relation containing all tuples $(a^0, a^m) \in G \times G$ for which there exist $a^1, \ldots, a^{m-1} \in G$ such that $(a^{i-1}, a^i) \in E$ for all $i \leq m$. We write E^{-1} for the relation $\{(b, a) : (a, b) \in E\}$. An E-*walk* from a^0 to a^n is a finite sequence $p = (a^0 E_1 a^1 E_2 a^2 \ldots E_n a^n)$, where $E_i \in \{E, E^{-1}\}$ and $(a^{i-1}, a^i) \in E_i$ for all i. We say that p is *closed* if $a^0 = a^n$. The *algebraic length* of p is the number of occurrences of E minus the number of occurrences of E^{-1}. A digraph $\mathbb{G} = (G; E)$ is said to have *algebraic length* 1 if there exists a closed E-walk of algebraic length 1. A digraph $\mathbb{G}$ is *weakly connected* if for all distinct $a, b \in G$ there exists an E-walk from a to b. A *fence* in E from a^0 to a^{2n} is an E-walk of the form $(a^0 E_1 \ldots E_{2n} a^{2n})$ where $E_i = E$ for odd i and $E_i = E^{-1}$ for even i. The vertices $a^0, a^2, \ldots, a^{2n}$ are called the *lower tips* of p.

For $m \in \mathbb{N}$, a fence in E^m is also called an *m-fence* in E (see Figure 3 for an example of a 2-fence). We say that $\mathbb{G}$ is *linked* if there exists $m \in \mathbb{N}$ such that for any $a, b \in \mathrm{pr}_1(E)$ there exists an m-fence in E from a to b. A finite smooth digraph is linked if and only if it is weakly connected and has algebraic length 1 [6, Claim 3.8].

A *hypergraph* is any structure $\mathbb{T} = (T; R)$ containing only one relation R, where R is not necessarily binary. We call elements of R *hyperedges*. Let n be the arity of R, and let Σ be a subgroup of the symmetric group $\mathrm{Sym}(n)$ on n elements. We say that $\mathbb{T}$ is *Σ-invariant* if for every $a \in R$ and $\pi \in \Sigma$ also $(a_{\pi(1)}, \dots, a_{\pi(n)}) \in R$. A hypergraph is *cyclic* if it is invariant with respect to the group of cyclic shifts, and 2-*transitive* if it is Σ-invariant for some 2-transitive group Σ. A *loop* is a constant tuple $(a, \dots, a) \in R$. If Ω is a subgroup of $\mathrm{Aut}(\mathbb{T})$, then a *pseudo-loop modulo* Ω is a tuple $(a_1, \dots, a_n) \in R$ such that all elements a_i belong to the same orbit of $\Omega \curvearrowright T$. If $\mathbb{T}$ is binary, it is said to have *pseudo-algebraic length* 1 *modulo* Ω if $\mathbb{T}$ admits a walk of algebraic length 1 such that its first and last vertex are contained in the same orbit of $\Omega \curvearrowright T$.

3.3 Algebraic

A *polymorphism* of arity n of a structure $\mathbb{A} = (A; \mathcal{R})$ is a mapping $f : A^n \to A$ that *preserves* every $R \in \mathcal{R}$, i.e. for every $R \in \mathcal{R}$ and any $t^1, \dots, t^n \in R$, the tuple $f(t^1, \dots, t^n)$ is contained in R. For an ω-categorical structure $\mathbb{A}$, a relation R is pp-definable in $\mathbb{A}$ if and only if it is preserved by all polymorphisms of $\mathbb{A}$ [33]. The *polymorphism clone* of $\mathbb{A}$ is the set of all polymorphisms of $\mathbb{A}$, and it is denoted by $\mathrm{Pol}(\mathbb{A})$. Observe that it indeed forms a *clone* in the sense of universal algebra, i.e. it is closed under composition and contains all projections.

An *identity* is a formal abstract expression of the form $f \approx g$, where f and g are terms over a common functional language. A clone $\mathcal{C}$ of operations on A is said to *satisfy* a given identity $f(x_1, \dots, x_n) \approx g(y_1, \dots, y_n)$ if the function symbols appearing in the terms f and g can be interpreted as elements of $\mathcal{C}$, such that the equality $f(x_1, \dots, x_n) = g(y_1, \dots, y_n)$ holds for any evaluation of the variables in A. The satisfaction of a set of identities is defined as the simultaneous satisfaction of all identities by means of the same interpretation of function symbols. The *pseudo-version* of an identity $f \approx g$ is the identity $u \circ f \approx v \circ g$, where u and v are fresh unary function symbols.

Given a finite hypergraph $\mathbb{T}$ of some arity $n \geq 2$ and an enumeration $(x_1^1, \dots, x_n^1), \dots, (x_1^m, \dots, x_n^m)$ of its edges, the *$\mathbb{T}$-loop condition* is the $(n-1)$-element set consisting of the identities $s(x_1^1, \dots, x_1^m) \approx s(x_2^1, \dots, x_2^m) \approx \cdots \approx s(x_n^1, \dots, x_n^m)$, where s is an arbitrary m-ary function symbol. Similarly, the *$\mathbb{T}$-pseudo-loop condition* is the set containing the pseudo-identities $u_1 \circ s(x_1^1, \dots, x_1^m) \approx u_2 \circ s(x_2^1, \dots, x_2^m) \approx \cdots \approx u_n \circ s(x_n^1, \dots, x_n^m)$.

4 Temporal constraint languages

A *temporal relation* is a relation first-order definable in $(\mathbb{Q}; <)$. A *temporal constraint language* is a relational structure with domain $\mathbb{Q}$ all of whose relations are temporal. Clearly, the natural action of $\mathrm{Aut}(\mathbb{Q}) := \mathrm{Aut}((\mathbb{Q}; <))$ on $\mathbb{Q}^k$ has finitely many orbits for every $k \in \mathbb{N}$. As a consequence, all temporal constraint languages ω-categorical.

Let F be a set of operations on $\mathbb{Q}$. The clone *generated* by F is the smallest clone of operations $\mathcal{C}$ that contains $F \cup \mathrm{Aut}(\mathbb{Q})$ and is closed under *interpolation*; the latter meaning that an operation g belongs to $\mathcal{C}$ if and only if for every finite subset $A \subseteq \mathbb{Q}$, there is $f \in \mathcal{C}$ agreeing with g on A. In the case that F contains only one operation f, we say that f generates g. The clones generated by an operation f and a set F of operations are denoted by $\langle f \rangle$ and $\langle F \rangle$, respectively. For $S \subseteq \mathbb{Q}^k$ and an operation f on $\mathbb{Q}$, we denote by $\langle S \rangle_f$ the smallest subset of $\mathbb{Q}^k$ that contains S and is preserved by every operation of $\langle f \rangle$.

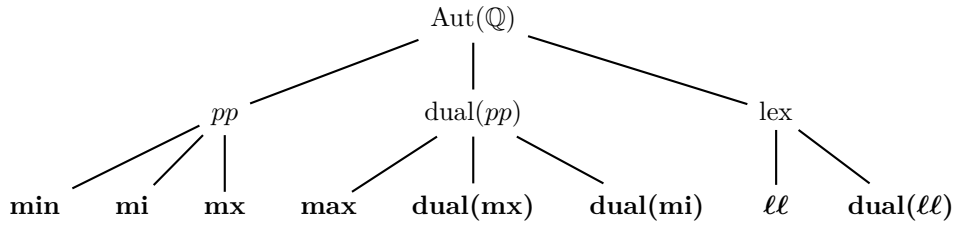

Figure 2 Polymorphisms of temporal constraint languages.

The *dual* of an n-ary operation f on $\mathbb{Q}$ is the operation defined by $\mathrm{dual}(f)(x_1, \ldots, x_n) \coloneqq -f(-x_1, \ldots, -x_n)$. For a set F of operations, $\mathrm{dual}(F)$ denotes the set containing all duals of operations in F. It is not hard to see that an operation f preserves a relation R if and only if its dual preserves the relation $-R = \{(-a_1, \ldots, -a_n) : (a_1, \ldots, a_n) \in R\}$ [26]. Moreover, as the dual of an automorphism of $(\mathbb{Q}; <)$ is again an automorphism, for every operation f we have $\langle \mathrm{dual}(f) \rangle = \mathrm{dual}(\langle f \rangle)$.

The Bodirsky-Kára classification (Theorem 6) identifies all not omni-expressive temporal constraint languages by the existence of specific polymorphisms, for whose definitions we refer to [40]. Figure 2 provides an illustration of the classification. For the proofs of Theorems 3 and 5, we will additionally draw on two binary operations lex and *pp*. As $\ell\ell$ and $\mathrm{dual}(\ell\ell)$ generate lex, and each of $\mathrm{min}, \mathrm{mi}$, and mx generates *pp*, it follows that one of lex, *pp*, and dual(*pp*) is contained in the polymorphism clone of every not omni-expressive constraint language. It is worth noting, however, that the presence of lex or *pp* in a polymorphism clone does not, on its own, imply polynomial-time solvability of the underlying template. For example, the temporal constraint language modelling the classical *Betweenness problem* is preserved by lex, yet is NP-complete [68].

▶ **Theorem 6** (Bodirsky-Kára classification [26]). *Let $\mathbb{A}$ be a temporal constraint language. If $\mathbb{A}$ is not omni-expressive, then it is preserved by one of* min, mi, mx, $\ell\ell$, *their duals, or a constant operation, and* $\mathrm{CSP}(\mathbb{A})$ *is solvable in polynomial time.*

4.1 Proof outline

The following subsection presents the key ideas and technical concepts underlying the proofs of the temporal parts of Theorems 3 and 5, with the aim of conveying the main intuition. We will mainly be interested in relations R on some finite power of $\mathbb{Q}$ that are pp-definable in a temporal constraint language $\mathbb{A}$. It follows that every polymorphism of $\mathbb{A}$ preserves R [34, 33]. Moreover, it suffices to consider relations R that are preserved by one of the operations $\mathrm{min}, \mathrm{mi}, \mathrm{mx}$, or $\ell\ell$. The general case for relations pp-interpretable in temporal constraint languages easily reduces to this setting, as discussed in the proof of Theorem 3 in Section 6. For $k \in \mathbb{N}$, we denote by $\sim_k$ the orbit-equivalence of $\mathrm{Aut}(\mathbb{Q}) \curvearrowright \mathbb{Q}^k$. In other words, for $a, b \in \mathbb{Q}^k$ we have $a \sim_k b$ if and only if $a_i \leq a_j \leftrightarrow b_i \leq b_j$ for all $i, j \leq k$. Whenever convenient, we naturally consider n-tuples on $\mathbb{Q}^k$ as tuples on $\mathbb{Q}^{nk}$.

4.1.1 Chasing orbits

In what follows, we fix $k \in \mathbb{N}$ and an n-ary relation R of $\mathbb{Q}^k$. A pseudo-loop in R modulo $\mathrm{Aut}(\mathbb{Q})$ is given by any tuple $(a_1, \ldots, a_n) \in R$ whose entries $a_1, \ldots, a_n \in \mathbb{Q}^k$ share the same relative ordering. We construct a pseudo-loop in R via a recursive procedure that, at each step, aligns the minimal elements of appropriately chosen tuples. After at most k-many steps, this recursion yields a tuple all of whose coordinates lie in the same orbit of $\mathrm{Aut}(\mathbb{Q}) \curvearrowright \mathbb{Q}^k$, i.e. a pseudo-loop modulo $\mathrm{Aut}(\mathbb{Q})$ in R.

▶ **Definition 7.**

- *For every $a \in \mathbb{Q}^k$, we denote by* $\min(a)$ *the minimal entry of* a*. By* $\operatorname{minx}(a)$ *we denote the set of all indices $i \leq k$ for which $a_i = \min(a)$.*
- *For all $a, b \in \mathbb{Q}^k$ and $I \subseteq [k]$, we write $a \sim_I b$ if $\operatorname{pr}_I(a) \sim_{|I|} \operatorname{pr}_I(b)$.*
- *For $1 \leq m \leq k$ and $a \in \mathbb{Q}^k$, we set $I_m(a)$ for the set of those indices i such that a_i is among the m smallest values that appear in a.*

We iteratively find tuples $t^m \in R$ so that, for each $m \leq k$ and for all $i, j \in [n]$,

1. $I_m(t_i^m) = I_m(t_j^m)$, i.e. the m smallest values of t_i^m and t_j^m are taken on the same set of coordinates;
2. $t_i^m \sim_{I_m(t_i^m)} t_j^m$, i.e. the subtuples of t_i^m and t_j^m containing precisely the m smallest values of t_i^m and t_j^m, respectively, are contained in the same $\sim_m$-class;
3. $\operatorname{pr}_{I_m(t_i^{m+1})}(t_i^{m+1}) \sim_m \operatorname{pr}_{I_m(t_i^m)}(t_i^m)$, i.e. the $\sim_m$-class of the subtuple of t_i^{m+1} containing the m smallest entries of t_i^{m+1} coincides with the $\sim_m$-class of the subtuple of t_i^m containing all of its m smallest entries.

At the end of this procedure, the tuple $t^k \in R$ clearly serves as the desired pseudo-loop in R.

4.1.2 Aligning minima

Finding a tuple $t^1 \in R$ satisfying conditions (1) and (2) for $m = 1$ is the same as finding a tuple $t^1 \in R$ satisfying $\operatorname{minx}(t_i^1) = \operatorname{minx}(t_j^1)$ for all $i, j \in [n]$. The construction of such a tuple relies on the following easy observation: let $a \in \mathbb{Q}^k$, $q \in \mathbb{Q}$, and define $J \coloneqq \{i \in [k] : a_i \leq q\}$. For every $b \in \mathbb{Q}^k$, the $\sim_k$-class of the k-tuple $pp_q(a, b)$ is determined only by the $\sim_{|J|}$-class of the $|J|$-tuple $\operatorname{pr}_J(a)$ and the $\sim_{(k-|J|)}$-class of the $(k - |J|)$-tuple $\operatorname{pr}_{[k]\setminus J}(b)$. The same holds true for the tuple $\ell\ell_q(a, b)$, provided that $\ker(a) = \ker(b)$. To make use of this observation, we restrict ourselves to tuples in R satisfying the following property:

▶ **Definition 8.** *Let t be an n-tuple on $\mathbb{Q}^k$. By $M(t)$ we denote the set consisting of all $i \in [n]$ for which* $\min(t_i) = \min(t)$*. We say that t is* min-clean *if for all $i, j \in M(t)$ we have* $\operatorname{minx}(t_i) = \operatorname{minx}(t_j)$*.*

Let us lay out how a min-clean tuple t in a relation R preserved by pp can be used to derive t^1 as above. Recall that a relation preserved by one of $\min, \mathrm{mi}$, or mx is always preserved by pp. The case of a relation preserved by $\ell\ell$ works similarly.

Cut at minimum. Let t be a min-clean n-tuple, $M \coloneqq \operatorname{minx}(t_i)$ for $i \in M(t)$, and set $q \coloneqq \min(t)$. Take an arbitrary n-tuple s and let $t^1 \coloneqq pp_q(t, s)$. For all $i \in M(t)$ we have $\operatorname{minx}(t_i^1) = M$, and for $j \notin M(t)$ we have $\operatorname{minx}(t_j^1) = \operatorname{minx}(s_j)$. Consequently, it suffices to take $s \in R$ with $\operatorname{minx}(s_j) = M$ for all $j \notin M(t)$. Such a tuple $s \in R$ exists by smoothness or appropriate symmetry properties of R.

4.1.3 Pseudo-algebraic length 1

The technical part of the proofs of Theorems 3 and 5 in the temporal case concerns the existence of min-clean tuples. We illustrate the representative case of a binary relation E on $\mathbb{Q}^k$ that is preserved by mi. We assume E to be smooth, weakly connected, and of pseudo-algebraic length 1 modulo $\operatorname{Aut}(\mathbb{Q})$.

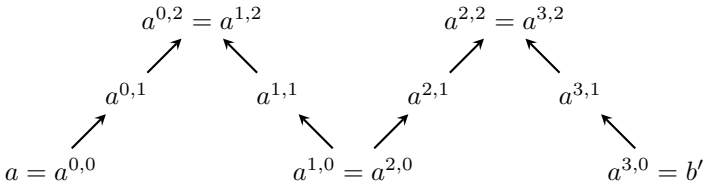

Figure 3 Transferring along fences.

Pseudo-linkedness. Recall that a finite smooth weakly connected digraph has algebraic length 1 if and only if it is m-linked for some $m \geq 1$ [6, Claim 3.8]. The k-factor of E gives rise to a finite digraph on $\sim_k$-classes that is smooth, weakly connected, and of algebraic length 1. Thus, it is m-linked for some $m \geq 1$. This means that for all $a, b \in \mathrm{supp}(E)$, there exist $b' \sim_k b$ and an m-fence in E connecting a to b'. Figure 3 depicts a 2-fence from a to b' consisting of 3 lower tips, where an edge from a vertex x to a vertex y represents $E(x, y)$. Observe that if $b \sim_k b'$, then $\mathrm{minx}(b) = \mathrm{minx}(b')$.

Controlling minimal index sets. Let $a, b \in \mathrm{supp}(E)$ be tuples sharing the same minimal entry (as it turns out, this can always be assumed by connectivity of E). Observe that if

$$\mathrm{minx}(a) \cap \mathrm{minx}(b) \neq \varnothing, \tag{7}$$

then $\mathrm{minx}(\mathrm{mi}(a, b)) = \mathrm{minx}(a) \cap \mathrm{minx}(b)$. In other words, condition (7) guarantees control over the minimal index set of $\mathrm{mi}(a, b)$.

Transferring minimal index sets along fences. There exist $b' \sim_k b$ and an m-fence in the $\sim_k$-factor of E connecting a to b' (Figure 3. We assumed $m = 2$ for simplicity). Since the two (identical) tuples $a^{0,2}$ and $a^{1,2}$ satisfy Equation (7), so do the tuples $a^{0,1}$ and $a^{1,1}$, and consequently also $a^{0,0}$ and $a^{1,0}$. Iterating this argument, we get that a and b' satisfy Equation (7). It follows that a and b do, as $\mathrm{minx}(b) = \mathrm{minx}(b')$.

Equalising minimal index sets. Using a nested composition of mi, we construct a tuple of the form $\mathrm{mi}(t, s)$, where $\mathrm{minx}(t_1) \cap \mathrm{minx}(s_1) = \mathrm{minx}(t_2) \cap \mathrm{minx}(s_2) \neq \varnothing$. In particular, this tuple is min-clean.

4.1.4 Summary

The proofs of the temporal parts of Theorems 3 and 5 split into the cases set by the classification of temporal constraint languages (Figure 2). It suffices to deal with relations R defined on some power of $\mathbb{Q}$ that are preserved by one of the operations $\min, \mathrm{mi}, \mathrm{mx}$, or $\ell\ell$. In every case, the proof consists of two parts:

1. show that R contains a min-clean tuple
2. apply a nested composition of operations to min-clean tuples to recursively construct a pseudo-loop in R.

Figure 4 provides a graphic overview of the proof's structure for the binary case.

5 Phylogeny Constraint Languages

A *tree* $\mathbb{B}$ is a finite, undirected graph that is connected and acyclic. For our purposes, it suffices to consider *binary branching rooted* trees, i.e. trees with a designated vertex r (the *root*) whose degree is 2, and whose all other vertices either have degree 3 or 1. A vertex

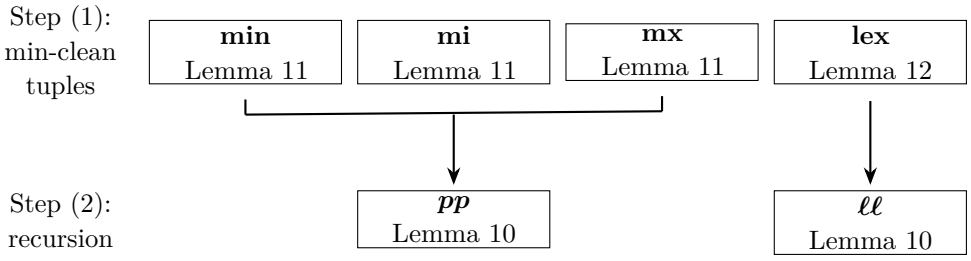

Figure 4 Proof structure of Theorem 3 for temporal constraint languages.

whose degree is 1 is called a *leaf*. We write $L(\mathbb{B})$ for the set of all leaves of $\mathbb{B}$. It is easy to see that in every tree, there is a unique shortest path between any two distinct vertices x and y. We write $x < y$ if the unique shortest path from x to r contains y. The *youngest common ancestor* of a set X of vertices is the lowest upper bound of X with respect to the partial order induced by $<$. Given three leaves a, b and c, we write $a|bc$ to denote that a is incomparable with the youngest common ancestor of the set $\{b, c\}$, i.e. there exists a vertex splitting a from b and c: a is a descendant of one of its children, while b and c are descendants of the other. We refer to $(L(\mathbb{B}); |)$ as the *leaf structure* of $\mathbb{B}$. It is a standard model-theoretic fact that there exists an – up to isomorphism unique – countably infinite structure $(\mathbb{L}; |)$ (namely, the *Fraïssé-limit* of the class of all leaf structures) with the following properties: a finite structure homomorphically maps to $(\mathbb{L}; |)$ if and only if it is the leaf structure of a binary branching rooted tree, and $(\mathbb{L}; |)$ is *homogeneous*, meaning that two k-tuples a and b over $\mathbb{L}$ are in the same orbit of $\mathrm{Aut}(\mathbb{L}; |)$ if and only if $a_i|a_j a_\ell \leftrightarrow b_i|b_j b_\ell$ for all $i, j, \ell \in [k]$. Moreover, the structure $(\mathbb{L}; |)$ is *finitely bounded* and therefore falls into the scope of the Bodirsky–Pinsker conjecture. A *phylogeny constraint language* is a relational structure with domain $\mathbb{L}$ all of whose relations are first-order definable in $(\mathbb{L}; |)$.

By [23], there exists a homogeneous expansion $(\mathbb{L}; |, \prec)$ of $(\mathbb{L}; |)$ such that $\prec$ is a linear order on $\mathbb{L}$ that is *convex*, i.e. whenever $a \prec b \prec c$, then either $a|bc$ or $c|ab$ holds.[0] We extend the ternary relation $|$ to non-empty subsets of $\mathbb{L}$ by setting $U_0|U_1$ if $(x_0|y_1 z_1)$ and $(x_1|y_0 z_0)$ hold true for all $x_i, y_i, z_i \in U_i$ for $i = 0, 1$.[1] Note that $U_0|U_1$ if and only if $U_1|U_0$. Every finite subset $U \subseteq \mathbb{L}$ with at least two elements can then be uniquely written as the union of two sets U_0, U_1 such that $U_0|U_1$ and $U_0 \prec U_1$. We call (U_0, U_1) the *generic partition of* U. The following theorem ensures the existence of a so-called *binary affine tree* polymorphism tx that is, in some precise sense, compatible with generic partitions. For its definition, we refer to [23].

▶ **Theorem 9** (cf. [23]). *Let $\mathbb{A}$ be a phylogeny constraint language. If $\mathbb{A}$ is not omni-expressive, then either it has a constant endomorphism or it has a binary affine tree polymorphism* tx.

Given finite sets $A, B \subseteq \mathbb{L}$, a binary operation f on $\mathbb{L}$ is *perfectly dominated by the first argument on* $A \times B$ if for all tuples $a \in A^3$ and $b \in B^3$, $a_1|a_2 a_3$ implies that $f(a_1, b_1)|f(a_2, b_2)f(a_3, b_3)$, and $b_1|b_2 b_3$ implies that $f(a_1, b_1)|f(a_1, b_2)f(a_1, b_3)$.[2] Perfect

[0] In fact, $(\mathbb{L}; |, \prec)$ is the Fraïssé-limit of the class of leaf structures of finite binary-branching rooted trees with an added convex linear order [23]. In this context, convexity describes that the linear order is the left-to-right ordering given by an inorder traversal of a tree, restriced to its leaves.

[1] This notation can be introduced analogously for the leaf structure of a finite binary-branching rooted tree $\mathbb{B}$. Here, $U_0|U_1$ means that some vertex $u \in \mathbb{B}$ has two children that separate the leaves, with U_0 descending from one child and U_1 from the other.

[2] Compare this with the case of a binary linear order, where the analogue of a binary function perfectly dominated by the first argument is a binary function preserving the lexicographic order.

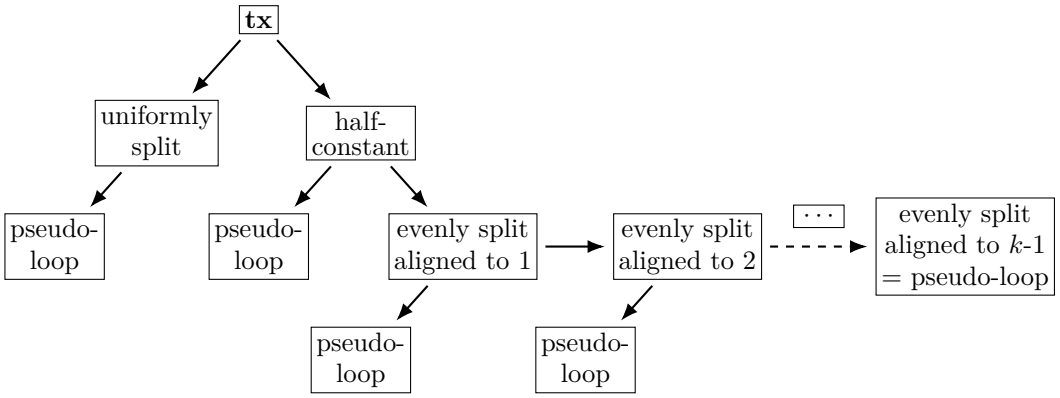

Figure 5 Structure tree of Theorem 3 for phylogeny constraint languages.

domination by the second variable is defined analogously. A binary tree operation tx then has the property that for every finite set $U \subseteq \mathbb{L}$ and every partition (U_0, U_1) of U such that $U_0|U_1$, tx is perfectly dominated by one variable on $U_0 \times U_1$, and by the other on $U_1 \times U_0$. By homogeneity of $(\mathbb{L}; |)$, for any such partition (U_0, U_1), the clone generated by any affine tree operation contains operations that are perfectly dominated by the first variable on $U_0 \times U_1$ and by the second on $U_1 \times U_0$, as well as the other way around.

5.1 Proof Outline

As for temporal constraint languages, pseudo-loops are constructed via a recursive procedure. The binary operations *pp* and $\ell\ell$ are replaced by the binary affine tree operation tx, whose domination properties guide the construction. Again, the recursion depends on the existence of tuples of a specific form.

By the above observation on generic partitions, for every non-constant $a \in \mathbb{L}^k$, there exists a unique vector $\mathrm{s}(a) \in \{0,1\}^k$ such that $\{a_i \colon s_i = 0\}|\{a_j \colon s_j = 1\}$ and $\{a_i \colon \mathrm{s}(a)_i = 0\} \prec \{a_j \colon \mathrm{s}(a)_j = 1\}$. If $a \in \mathbb{L}^k$ is constant, we set $\mathrm{s}(a) = (0, \ldots, 0)$. The vector $\mathrm{s}(a)$ is the *split vector* of a. Two k-tuples a and b over $\mathbb{L}$ are in the same orbit of $\mathrm{Aut}(\mathbb{L}; |)$ if and only if the split vectors of all corresponding 2- and 3-subtuples of a and b are identical or, since $\prec$ is not invariant under automorphisms of $(\mathbb{L}; |)$, dual in every coordinate. We say that an n-tuple t on $\mathbb{L}^k$ is *evenly split* if $\mathrm{s}(t_i) = \mathrm{s}(t_j)$ for all $i, j \in [n]$. Though pseudo-loops need not be evenly split, evenly split tuples are central to their inductive construction. Under fairly general assumptions, such tuples can always be guaranteed to exist. The coordinates $t_i \in \mathbb{L}^k$ of an evenly split n-tuple t agree precisely in which of their entries $\mathrm{pr}_j(t_i)$ are assigned to which "side" with respect to the generic partition of the set of entries of t_i, e.g. the j-th entry of every coordinate may always be assigned to the "left". We then construct a finite sequence $(U_{w0}, U_{w1})_{w \in \{0,1\}^*}$ of sub-partitions and use the domination properties of tx to recursively align both of these sides while preserving the splits. After finitely many steps, this procedure yields a pseudo-loop. Figure 5 illustrates the procedure in the case of digraphs.

6 Proof of the Main Theorems

In the following, we state the results needed to derive the temporal part of Theorem 3. The proof of the higher-arity case (Theorem 5) follows the same line of argument, with only minor modifications. Although the classes of structures under consideration are fundamentally

different, the proofs of the corresponding statements for phylogeny constraint languages rely on similar ideas. By standard methods, the existence of pseudo-loops then implies the validity of the corresponding pseudo-loop conditions.

When working with the binary polymorphism $\ell\ell$, we need to "synchronise" the kernels of the tuples we work with. Recall that $\ell\ell$ generates the binary injection lex. For any relation S on $\mathbb{Q}^k$, we define $S' := \{t \in \langle S \rangle_{\text{lex}} \mid \ker t = \bigcap_{s \in S} \ker s\}$. It is easy to see that S' is non-empty and smooth whenever S is. We now present the heart of the iterative process for constructing a pseudo-loop.

▶ **Lemma 10.** *Let E be a binary smooth relation on $\mathbb{Q}^k$ that is preserved by $\langle \ell\ell \rangle$. Let $S \subseteq E$ be smooth such that S' contains a min-clean tuple $\binom{a^1}{b^1}$. Then there exist $I \subseteq k$, $m, n \geq 1, q \in \mathbb{Q}$, and $\binom{a^2}{b^2}, \dots, \binom{a^n}{b^n} \in S'$ such that for all $\binom{a}{b} \in S'$ the tuple defined by*

$$\binom{a'}{b'} := \ell\ell_q^{[n+1]}\left(\binom{a^1}{b^1}, \binom{a^2}{b^2}, \dots, \binom{a^n}{b^n}, \binom{a}{b}\right)$$

satisfies:

- *$I_m(a') = I_m(b') = I$ and $a' \sim_I b'$;*
- *$a' \sim_{[k]\setminus I} a$ and $b' \sim_{[k]\setminus I} b$.*

The statement remains valid after replacing every occurrence of $\ell\ell$ by pp and every occurrence of S' by E.

Proof. If $\min a^1 = \min b^1$ then $\operatorname{minx} a^1 = \operatorname{minx} b^1$. In this case, we set $I := \operatorname{minx} a^1$, $m := 1$, $n := 1$, and $q := \min a^1$. Otherwise, assume without loss of generality that $\min a^1 < \min b^1$. We set $q := \min a^1$, and inductively set $b^{i+1} := a^i$, and pick a^{i+1} arbitrarily so that $\binom{a^{i+1}}{b^{i+1}} \in S'$. Set $J^1 := \operatorname{minx} a^1$, and inductively $J^{i+1} := \{j \mid a^{i+1}_j \leq q\} \setminus (J^1 \cup \dots \cup J^i)$. Finally, we pick $n \geq 2$ minimal such that J^n is empty, set I to be the union $J^1 \cup \dots \cup J^n$, and $m := |I|$.

By definition of S', for all i we have $a' \sim_{J_i} a^i$ and $b' \sim_{J_i} b^{i+1} = a^i$, hence $a' \sim_{J_i} b'$. Moreover, every element appearing in the tuple $\operatorname{pr}_{J_i}(a')$ is strictly smaller than every element appearing in the tuple $\operatorname{pr}_{J_{i+1}}(a')$. Similarly, all entries of $\operatorname{pr}_{J_i}(b')$ are strictly smaller than all entries of $\operatorname{pr}_{J_{i+1}}(b')$ for all i. It follows that $a' \sim_I b'$. Moreover, the entries of $\operatorname{pr}_I(a')$ are smaller than the entries of $\operatorname{pr}_{[k]\setminus I}(a')$, and the entries of $\operatorname{pr}_I(b')$ are smaller than the entries of $\operatorname{pr}_{[k]\setminus I}(b')$, hence $I_m(a') = I_m(b') = I$. Finally, it is easy to verify that $a' \sim_{[k]\setminus I} a$ and $b' \sim_{[k]\setminus I} b$. The same construction works for the case of *pp*. ◀

The existence of min-clean tuples within E is guaranteed whenever E is smooth, has pseudo-algebraic length 1, and is preserved by one of $\langle \min \rangle$, $\langle \text{mi} \rangle$, or $\langle \text{mx} \rangle$. As illustrated in Section 4.1.3, we make heavy use of the fact that by smoothness and pseudo-algebraic length 1 of E, there is a fence in E connecting any $a, b \in \operatorname{supp}(E)$ modulo orbit-equivalence.

▶ **Lemma 11.** *Let E be a binary relation on $\mathbb{Q}^k$ that is preserved by one of $\langle \min \rangle$, $\langle \text{mi} \rangle$, or $\langle \text{mx} \rangle$. If E is smooth and has pseudo-algebraic length 1 modulo $\operatorname{Aut}(\mathbb{Q})$, then E contains a min-clean tuple.*

The crucial difference for relations preserved by $\langle \ell\ell \rangle$ is that all tuples we work with must be contained in E'. This additional prerequisite leads to the $\ell\ell$ part of Lemma 10 being slightly weaker than its analogue for *pp*, compensated by strengthened results showing the existence of min-clean tuples even within E'.

▶ **Lemma 12.** *Let E be a binary relation on $\mathbb{Q}^k$. If E is smooth, weakly connected, and of pseudo-algebraic length 1 modulo $\operatorname{Aut}(\mathbb{Q})$, then E' contains a min-clean tuple.*

We get a pseudo-loop by iteratively applying Lemma 10 to relations defined on strictly decreasing powers k of $\mathbb{Q}$.

▶ **Lemma 13.** *Let E be a binary relation on $\mathbb{Q}^k$ that is smooth and has pseudo-algebraic length* 1 *modulo* $\mathrm{Aut}(\mathbb{Q})$*. Then each of $\langle E\rangle_{\mathrm{min}}, \langle E\rangle_{\mathrm{mi}}$, and $\langle E\rangle_{\mathrm{mx}}$ contains a pseudo-loop modulo* $\mathrm{Aut}(\mathbb{Q})$*. If, additionally, E is weakly connected, then also $\langle E'\rangle_{\ell\ell}$ contains a pseudo-loop modulo* $\mathrm{Aut}(\mathbb{Q})$*.*

With the classification of not omni-expressive temporal constraint languages in hand, it remains to combine the results established thus far to derive the temporal part of Theorem 3.

▶ **Theorem 14.** *Let $\mathbb{A}$ be a temporal constraint language that is not omni-expressive. If $\mathbb{G}$ is any smooth digraph that is pp-interpretable in $\mathbb{A}$ and has pseudo-algebraic length* 1 *modulo* $\mathrm{Aut}(\mathbb{Q})$*, then $\mathbb{G}$ contains a pseudo-loop modulo* $\mathrm{Aut}(\mathbb{Q})$*.*

Proof. Replacing $\mathbb{G}$ by the preimage $\bar{\mathbb{G}}$ of $\mathbb{G}$ under the map of the pp-interpretation in $\mathbb{A}$, we may assume that $\mathbb{G}$ is a digraph defined on a finite power of $\mathbb{Q}$ whose edge relation E is preserved by every polymorphism of $\mathbb{A}$. Indeed, by definition of a pp-interpretation, the preimage $\bar{E}$ of E is pp-definable in $\mathbb{A}$, and therefore preserved by $\mathrm{Pol}(\mathbb{A})$ [34, 33]. A pseudo-loop in $\bar{E}$ yields a pseudo-loop in E modulo $\mathrm{Aut}(\mathbb{Q})$. Moreover, $\bar{E}$ is smooth and has pseudo-algebraic length modulo $\mathrm{Aut}(\mathbb{Q})$.

By Theorem 6, $\mathbb{A}$ is preserved by one of $\ell\ell$, min, mi, mx, their duals, or a constant operation. Accordingly, $\mathbb{G}$ is preserved by one of these operations, and thus by all operations in its generated clone. Clearly, every non-empty relation preserved by a constant operation contains even a loop. Moreover, $\mathbb{G}$ contains a pseudo-loop if and only if $-\mathbb{G}$ does. Recalling that, for $f \in \{\ell\ell, \mathrm{min}, \mathrm{mi}, \mathrm{mx}\}$, $\mathbb{G}$ is preserved by $\langle \mathrm{dual}(f)\rangle$ if and only if $-\mathbb{G}$ is preserved by $\langle f\rangle$, this allows us to restrict ourselves to the cases where $\mathbb{G}$ is preserved by one of $\langle\ell\ell\rangle$, $\langle\mathrm{min}\rangle$, $\langle\mathrm{mi}\rangle$, and $\langle\mathrm{mx}\rangle$. In the first case, let $S \subseteq E$ be smooth, weakly connected, and of pseudo-algebraic length 1 modulo $\mathrm{Aut}(\mathbb{Q})$. By Lemma 13, $\langle S'\rangle_{\ell\ell} \subseteq E$ contains a pseudo-loop modulo $\mathrm{Aut}(\mathbb{Q})$. In all the other cases, we can directly apply Lemma 13, yielding a pseudo-loop modulo $\mathrm{Aut}(\mathbb{Q})$ within E. ◀

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 J. Computing*, 10(3):405–421, 1981. doi:10.1137/0210030.

2 Libor Barto, Bertalan Bodor, Marcin Kozik, Antoine Mottet, and Michael Pinsker. Symmetries of graphs and structures that fail to interpret a finite thing. In *Proceedings of the 38th Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'23*, pages 1–13. IEEE, 2023. doi:10.1109/LICS56636.2023.10175732.

3 Libor Barto, Michael Kompatscher, Miroslav Olšák, Trung Van Pham, and Michael Pinsker. Equations in oligomorphic clones and the constraint satisfaction problem for ω-categorical structures. *J. Math. Log.*, 19(02):Article 1950010, 2019. doi:10.1142/S0219061319500107.

4 Libor Barto, Michael Kompatscher, Miroslav Olšák, Trung Van Pham, and Michael Pinsker. The equivalence of two dichotomy conjectures for infinite domain constraint satisfaction problems. In *Proceedings of the 32nd Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'16*, pages 1–12. IEEE, 2017. doi:10.1109/LICS.2017.8005128.

5 Libor Barto and Marcin Kozik. Constraint satisfaction problems of bounded width. In *Proceedings of the 50th IEEE Annual Symposium on Foundations of Computer Science - FOCS'09*, pages 595–603. IEEE, 2009. doi:10.1109/FOCS.2009.32.

6 Libor Barto and Marcin Kozik. Absorbing subalgebras, cyclic terms, and the constraint satisfaction problem. *Log. Methods Comput. Sci.*, 8(1):7:1–7:27, 2012. doi:10.2168/lmcs-8(1:7)2012.

7 Libor Barto and Marcin Kozik. Constraint satisfaction problems solvable by local consistency methods. *J. ACM.*, 61(1):3:1–3:19, 2014. doi:10.1145/2556646.

8 Libor Barto, Marcin Kozik, and Todd Niven. The CSP dichotomy holds for digraphs with no sources and no sinks (a positive answer to a conjecture of Bang-Jensen and Hell). *SIAM J. Comput.*, 38(5):1782–1802, 2009. doi:10.1137/070708093.

9 Libor Barto, Jakub Opršal, and Michael Pinsker. The wonderland of reflections. *Isr. J. Math.*, 223(1):363–398, 2018. doi:10.1007/s11856-017-1621-9.

10 Libor Barto and Michael Pinsker. The algebraic dichotomy conjecture for infinite domain constraint satisfaction problems. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'16*, pages 615–622. ACM, 2016. doi:10.1145/2933575.2934544.

11 Libor Barto and Michael Pinsker. Topology Is Irrelevant (In a Dichotomy Conjecture for Infinite Domain Constraint Satisfaction Problems). *SIAM J. Comput.*, 49(2):365–393, 2020. doi:10.1137/18M1216213.

12 Zeno Bitter and Antoine Mottet. Generalized completion problems with forbidden tournaments. In *Proceedings of the 49th International Symposium on Mathematical Foundations of Computer Science - MFCS'24*, volume 306 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 28:1–28:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2024. doi:10.4230/LIPIcs.MFCS.2024.28.

13 Manuel Bodirsky. *Complexity of infinite-domain constraint satisfaction*, volume 52. Cambridge University Press, 2021. Postprint available online. doi:10.1017/9781107337534.

14 Manuel Bodirsky, Víctor Dalmau, Barnaby Martin, Antoine Mottet, and Michael Pinsker. Distance constraint satisfaction problems. *Inf. Comput.*, 247:87–105, 2016. doi:10.1016/j.ic.2015.11.010.

15 Manuel Bodirsky, Víctor Dalmau, Barnaby Martin, and Michael Pinsker. Distance constraint satisfaction problems. In *Proceedings of the 35th International Symposium on Mathematical Foundations of Computer Science - MFCS'10*, Lecture Notes in Computer Science, pages 162–173. Springer Verlag, 2010. doi:10.1007/978-3-642-15155-2_16.

16 Manuel Bodirsky and Martin Grohe. Non-dichotomies in constraint satisfaction complexity. In *Proceedings of the 35th International Colloquium on Automata, Languages, and Programming - ICALP'08*, pages 184–196. Springer, 2008. doi:10.1007/978-3-540-70583-3_16.

17 Manuel Bodirsky and Santiago Guzmán-Pro. Forbidden tournaments and the orientation completion problem. *SIAM J. Discrete Math.*, 39(1):170–205, 2025. doi:10.1137/23M1604849.

18 Manuel Bodirsky, Martin Hils, and Barnaby Martin. On the scope of the universal-algebraic approach to constraint satisfaction. In *Proceedings of the Symposium on Logic in Computer Science (LICS)*, pages 90–99. IEEE Computer Society, July 2010. doi:10.1109/LICS.2010.13.

19 Manuel Bodirsky, Martin Hils, and Barnaby Martin. On the scope of the universal-algebraic approach to constraint satisfaction. *Log. Methods Comput. Sci.*, 8(3):1–30, 2012. doi:10.2168/LMCS-8(3:13)2012.

20 Manuel Bodirsky, Peter Jonsson, Barnaby Martin, Antoine Mottet, and Žaneta Semanišinová. Complexity classification transfer for CSPs via algebraic products. *SIAM J. Comput.*, 53(5):1293–1353, 2024. doi:10.1137/22M1534304.

21 Manuel Bodirsky, Peter Jonsson, and Trung Van Pham. The reducts of the homogeneous binary branching C-relation. *J. Symb. Log.*, 81(4):1255–1297, 2016. doi:10.1017/jsl.2016.37.

22 Manuel Bodirsky, Peter Jonsson, and Trung Van Pham. The Complexity of Phylogeny Constraint Satisfaction. In *Proceedings of the 33rd Symposium on Theoretical Aspects of Computer Science - STACS'16*, volume 47 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 20:1–20:13, 2016. doi:10.4230/LIPIcs.STACS.2016.20.

23 Manuel Bodirsky, Peter Jonsson, and Trung Van Pham. The Complexity of Phylogeny Constraint Satisfaction Problems. *ACM Trans. Comput. Log.*, 18(3), 2017. An extended abstract appeared in the proceedings of the 33rd Symposium on Theoretical Aspects of Computer Science - STACS'16. doi:10.1145/3105907.

24 Manuel Bodirsky and Jan Kára. The complexity of equality constraint languages. *Theory Comput. Syst.*, 43:136–158, 2008. A conference version appeared in the proceedings of Computer Science Russia - CSR'06. doi:10.1007/s00224-007-9083-9.

25 Manuel Bodirsky and Jan Kára. The complexity of temporal constraint satisfaction problems. In *Proceedings of the 40th annual ACM symposium on Theory of computing - STOC'08*, pages 29–38. ACM, 2008. doi:10.1145/1374376.1374382.

26 Manuel Bodirsky and Jan Kára. The complexity of temporal constraint satisfaction problems. *J. ACM.*, 57(2):1–41, 2010. doi:10.1145/1667053.1667058.

27 Manuel Bodirsky, Florent Madelaine, and Antoine Mottet. A universal-algebraic proof of the complexity dichotomy for Monotone Monadic SNP. In *Proceedings of the 33st Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'18*, pages 105–114. ACM, 2018. doi:10.1145/3209108.3209156.

28 Manuel Bodirsky, Florent R. Madelaine, and Antoine Mottet. A proof of the algebraic tractability conjecture for monotone monadic SNP. *SIAM J. Comput.*, 50(4):1359–1409, 2021. doi:10.1137/19M128466X.

29 Manuel Bodirsky, Barnaby Martin, and Antoine Mottet. Discrete temporal constraint satisfaction problems. *J. ACM.*, 65(2):9:1–9:41, 2018. doi:10.1145/3154832.

30 Manuel Bodirsky, Barnaby Martin, Michael Pinsker, and András Pongrácz. Constraint satisfaction problems for reducts of homogeneous graphs. *SIAM J. Comput.*, 48(4):1224–1264, 2019. A conference version appeared in the proceedings of the 43rd International Colloquium on Automata, Languages, and Programming - ICALP'16. doi:10.1137/16M1082974.

31 Manuel Bodirsky and Antoine Mottet. Reducts of finitely bounded homogeneous structures, and lifting tractability from finite-domain constraint satisfaction. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'16*, pages 623–632. ACM, 2016. doi:10.1145/2933575.2934515.

32 Manuel Bodirsky and Jens K Mueller. The complexity of rooted phylogeny problems. *Log. Methods Comput. Sci.*, Volume 7, Issue 4, November 2011. doi:10.2168/LMCS-7(4:6)2011.

33 Manuel Bodirsky and Jaroslav Nešetřil. Constraint satisfaction with countable homogeneous templates. *J. Log. Comput*, 16(3):359–373, 2006. A conference version appeared in the proceedings of Computer Science Logic - CSL'03. doi:10.1093/logcom/exi083.

34 Manuel Bodirsky and Jaroslav Nešetřil. Constraint satisfaction with countable homogeneous templates. In *Proceedings of Computer Science Logic - CSL'03*, pages 44–57, 2003. doi:10.1007/978-3-540-45220-1_5.

35 Manuel Bodirsky and Michael Pinsker. Schaefer's theorem for graphs. *J. ACM.*, 62(3):19:1–19:52, 2015. A conference version appeared in the proceedings of the 43rd ACM Symposium on Theory of Computing - STOC'11. doi:10.1145/2764899.

36 Manuel Bodirsky and Michael Pinsker. Topological Birkhoff. *Trans. Amer. Math. Soc.*, 367(4):2527–2549, 2015. doi:10.1090/S0002-9947-2014-05975-8.

37 Manuel Bodirsky and Michael Pinsker. Canonical functions: a proof via topological dynamics. *Contrib. Discrete Math.*, 16(2):36–45, 2021. doi:10.11575/cdm.v16i2.71724.

38 Manuel Bodirsky, Michael Pinsker, and András Pongrácz. Projective clone homomorphisms. *J. Symb. Log.*, 86(1):148–161, 2021. doi:10.1017/jsl.2019.23.

39 Manuel Bodirsky, Michael Pinsker, and Todor Tsankovs. Decidability of definability. *J. Symb. Log.*, 78(4):1036–1054, 2013. A conference version appeared in the proceedings of the 26th Annual IEEE Symposium on Logic in Computer Science - LICS'11. doi:10.2178/jsl.7804020.

40 Manuel Bodirsky and Jakub Rydval. On the descriptive complexity of temporal constraint satisfaction problems. *J. ACM*, 70(1):2:1–2:58, 2022. A conference version under the title "Temporal Constraint Satisfaction Problems in Fixed-Point Logic" appeared in the proceedings of the 35th Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'20. doi:10.1145/3566051.

41 Mathias Broxvall and Peter Jonsson. Point algebras for temporal reasoning: Algorithms and complexity. *Artif. Intell.*, 149(2):179–220, 2003. doi:10.1016/S0004-3702(03)00075-4.

42 Johanna Brunar. Identities in finite Taylor algebras. Diploma thesis, 2023. doi:10.34726/hss.2023.107802.

43 Johanna Brunar, Marcin Kozik, Tomás Nagy, and Michael Pinsker. The sorrows of a smooth digraph: the first hardness criterion for infinite directed graph-colouring problems. In *Proceedings of the 40th Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'25*, pages 1–13. IEEE, 2025. doi:10.1109/LICS65433.2025.00037.

44 Andrei A. Bulatov. A dichotomy theorem for constraint satisfaction problems on a 3-element set. *J. ACM.*, 53(1):66–120, 2006. doi:10.1145/1120582.1120584.

45 Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs. In *Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science - FOCS'17*, pages 319–330. IEEE, 2017. doi:10.1109/FOCS.2017.37.

46 Andrei A. Bulatov, Peter Jeavons, and Andrei A. Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, 2005. doi:10.1137/S0097539700376676.

47 Thomas Drakengren and Peter Jonsson. Computational complexity of temporal constraint problems. In *Handbook of Temporal Reasoning in Artificial Intelligence*, volume 1 of *Foundations of Artificial Intelligence*, pages 197–218. Elsevier, 2005. doi:10.1016/S1574-6526(05)80008-2.

48 Tomás Feder and Moshe Y. Vardi. Monotone monadic SNP and constraint satisfaction. In *Proceedings of the 25th annual ACM symposium on Theory of Computing - STOC'93*, pages 612–622. ACM, 1993. doi:10.1145/167088.167245.

49 Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1):57–104, 1998. doi:10.1137/S0097539794266766.

50 Roman Feller and Michael Pinsker. An algebraic proof of the dichotomy for graph orientation problems with forbidden tournaments. *SIAM J. Discrete Math.*, 40(1):1–31, 2026. doi:10.1137/25M1726157.

51 Pierre Gillibert, Julius Jonušas, Michael Kompatscher, Antoine Mottet, and Michael Pinsker. Hrushovski's encoding and ω-categorical CSP monsters. In *Proceedings of the 47th International Colloquium on Automata, Languages, and Programming- ICALP'20*, volume 168 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 131:1–131:17. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2020. doi:10.4230/LIPIcs.ICALP.2020.131.

52 Pierre Gillibert, Julius Jonušas, Michael Kompatscher, Antoine Mottet, and Michael Pinsker. When symmetries are not enough: a hierarchy of hard Constraint Satisfaction Problems. *SIAM J. Computing*, 51(2), 2022. doi:10.1137/20M1383471.

53 Pierre Gillibert, Julius Jonušas, and Michael Pinsker. Pseudo-loop conditions. *Bull. Lond. Math. Soc.*, 51(5):917–936, 2019. doi:10.1112/blms.12286.

54 Pavol Hell and Jaroslav Nešetřil. On the complexity of H-coloring. *Journal of Combinatorial Theory, Series B*, 48:92–110, 1990. doi:10.1016/0095-8956(90)90132-J.

55 Peter Jeavons. On the algebraic structure of combinatorial problems. *Theor. Comput. Sci.*, 200(1-2):185–204, 1998. doi:10.1016/S0304-3975(97)00230-2.

56 Peter Jonsson and Thomas Drakengren. A complete classification of tractability in RCC-5. *J. Artif. Intell. Res.*, 6:211–221, 1997. doi:10.1613/jair.379.

57 Tamás Kátay, László Márton Tóth, and Zoltán Vidnyánszky. The CSP dichotomy, the axiom of choice, and cyclic polymorphisms. Preprint arXiv:2310.00514, 2024. doi:10.48550/arXiv.2310.00514.

58 Keith A. Kearnes, Petar Marković, and Ralph McKenzie. Optimal strong Mal'cev conditions for omitting type 1 in locally finite varieties. *Algebra Univers.*, 72(1):91–100, 2015. doi:10.1007/s00012-014-0289-9.

59 Michael Kompatscher and Trung Van Pham. A complexity dichotomy for poset constraint satisfaction. In *Proceedings of the 34th Symposium on Theoretical Aspects of Computer Science - STACS'17*, volume 66 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 47:1–47:12. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2017. doi:10.4230/LIPIcs.STACS.2017.47.

60 Michael Kompatscher and Trung Van Pham. A complexity dichotomy for poset constraint satisfaction. *IfCoLoG J. Log. their Appl.*, 5(8):1663–1696, 2018. URL: http://collegepublications.co.uk/ifcolog/?00028.

61 Andrei A. Krokhin, Peter Jeavons, and Peter Jonsson. Reasoning about temporal relations: The tractable subalgebras of Allen's interval algebra. *J. ACM.*, 50(5):591–640, 2003. doi:10.1145/876638.876639.

62 Richard E. Ladner. On the structure of polynomial time reducibility. *J. ACM.*, 22(1):155–171, 1975. doi:10.1145/321864.321877.

63 Miklós Maróti and Ralph McKenzie. Existence theorems for weakly symmetric operations. *Algebra Univers.*, 59(3):463–489, 2008. doi:10.1007/s00012-008-2122-9.

64 Antoine Mottet. Algebraic and algorithmic synergies between promise and infinite-domain CSPs. In *Proceedings of the 40th Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'25*, pages 1–13. IEEE, 2025. doi:10.1109/LICS65433.2025.00034.

65 Antoine Mottet and Michael Pinsker. Smooth approximations and csps over finitely bounded homogeneous structures. In *Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science - LICS'22*, pages 36:1–36:13, 2022. doi:10.1145/3531130.3533353.

66 Antoine Mottet and Michael Pinsker. Smooth approximations: An algebraic approach to CSPs over finitely bounded homogeneous structures. *J. ACM.*, 71(5):36:1–36:47, 2024. doi:10.1145/3689207.

67 Miroslav Olšák. The weakest nontrivial idempotent equations. *Bull. Lond. Math. Soc.*, 49(6):1028–1047, 2017. doi:10.1112/blms.12097.

68 Jaroslav Opatrny. Total ordering problem. *SIAM J. Comput.*, 8(1):111–114, 1979. doi:10.1137/0208008.

69 Michael Pinsker, Jakub Rydval, Moritz Schöbi, and Christoph Spiess. Three fundamental questions in modern infinite-domain constraint satisfaction. In *Proceedings of the 50th International Symposium on Mathematical Foundations of Computer Science - MFCS'25*, volume 345, pages 83:1–83:20, Dagstuhl, Germany, 2025. Leibniz International Proceedings in Informatics (LIPIcs). doi:10.4230/LIPIcs.MFCS.2025.83.

70 Mark H. Siggers. A strong Mal'cev condition for locally finite varieties omitting the unary type. *Algebra Univers.*, 64:15–20, 2010. doi:10.1007/s00012-010-0082-3.

71 Dmitriy Zhuk. A proof of CSP dichotomy conjecture. In *Proceedings of the 58th IEEE Annual Symposium on Foundations of Computer Science - FOCS'17*, pages 331–342. IEEE, 2017. doi:10.1109/FOCS.2017.38.

72 Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM.*, 67(5):30:1–30:78, 2020. doi:10.1145/3402029.

73 Dmitriy Zhuk. Strong subalgebras and the constraint satisfaction problem. *J. Multiple Valued Log. Soft Comput.*, 36(4-5):455–504, 2020. URL: https://www.oldcitypublishing.com/journals/mvlsc-home/mvlsc-issue-contents/mvlsc-volume-36-number-4-5-2021/mvlsc-36-4-5-p-455-504/.

74 Dmitriy Zhuk. A simplified proof of the CSP dichotomy conjecture and XY-symmetric operations. Preprint ArXiv: 10.48550/arXiv.2404.01080, 2024. doi:10.48550/arXiv.2404.01080.