


Restricted CSPs and F-Free Digraph Algorithmics

Santiago Guzmán-Pro 

Institut für Algebra, TU Dresden, Germany

Barnaby Martin 

Department of Computer Science, Durham University, UK

Abstract

In recent years, much attention has been placed on the complexity of graph homomorphism problems when the input is restricted to \mathbb{P}_k -free and \mathbb{P}_k -subgraph-free graphs. We consider the directed version of this research line, by addressing the question *is it true that digraph homomorphism problems $\text{CSP}(\mathbb{H})$ have a P versus NP-complete dichotomy when the input is restricted to $\tilde{\mathbb{P}}_k$ -free (resp. $\tilde{\mathbb{P}}_k$ -subgraph-free) digraphs?* Our main contribution in this direction shows that if $\text{CSP}(\mathbb{H})$ is NP-complete, then there is a positive integer N such that $\text{CSP}(\mathbb{H})$ remains NP-hard even for $\tilde{\mathbb{P}}_N$ -subgraph-free digraphs. Moreover, $\text{CSP}(\mathbb{H})$ becomes polynomial-time solvable for $\tilde{\mathbb{P}}_{N-1}$ -subgraph-free *acyclic* digraphs. We then verify the questions above for digraphs on three vertices and a family of smooth tournaments. We prove these results by establishing a connection between \mathbb{F} -(subgraph)-free algorithmics and constraint satisfaction theory. On the way, we introduce *restricted CSPs*, i.e., problems of the form $\text{CSP}(\mathbb{H})$ restricted to yes-instances of $\text{CSP}(\mathbb{H}')$ – these were called restricted homomorphism problems by Hell and Nešetřil. Another main result of this paper presents a P versus NP-complete dichotomy for these problems. Moreover, this complexity dichotomy is accompanied by an algebraic dichotomy in the spirit of the finite domain CSP dichotomy.

2012 ACM Subject Classification Mathematics of computing → Graph theory; Theory of computation → Graph algorithms analysis; Theory of computation → Logic

Keywords and phrases Digraph homomorphisms, constraint satisfaction problems, subgraph-free algorithmics

Digital Object Identifier 10.4230/LIPIcs.ICALP.2025.158

Category Track B: Automata, Logic, Semantics, and Theory of Programming

Related Version *Full Version:* <https://arxiv.org/abs/2502.17596> [19]

Funding *Santiago Guzmán-Pro:* Funded by the European Research Council (Project POCOCOP, ERC Synergy Grant 101071674). Views and opinions expressed are however those of the authors 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.

Barnaby Martin: Funded by EPSRC Grant EP/X03190X/1.

*As little as a few years ago, most graph theorists, while passively aware of a few classical results on graph homomorphisms, would not include homomorphisms among the topics of central interest in graph theory. We believe that this perception is changing, principally because of the usefulness of the homomorphism perspective... At the same time, the homomorphism framework strengthens the link between graph theory and other parts of mathematics, making graph theory more attractive, and understandable, to other mathematicians. Pavol Hell and Jaroslav Nešetřil, *Graphs and Homomorphisms*, 2004 [21].*



© Santiago Guzmán-Pro and Barnaby Martin;
licensed under Creative Commons License CC-BY 4.0

52nd International Colloquium on Automata, Languages, and Programming (ICALP 2025).

Editors: Keren Censor-Hillel, Fabrizio Grandoni, Joël Ouaknine, and Gabriele Puppis

Article No. 158; pp. 158:1–158:21



Leibniz International Proceedings in Informatics

LIPIcs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



1 Introduction

Story of the main questions

The Hell-Nešetřil theorem asserts that if \mathbb{H} is a finite undirected graph, then $\text{CSP}(\mathbb{H})$ is polynomial-time solvable whenever \mathbb{H} is either a bipartite graph or contains a loop, and otherwise $\text{CSP}(\mathbb{H})$ is NP-complete. In recent years, much attention has been placed on the complexity of graph homomorphism problems when the input is restricted to \mathbb{F} -free and \mathbb{F} -subgraph-free graphs, i.e., to graphs avoiding \mathbb{F} as an induced subgraph, and as a subgraph, respectively. In [17], the authors show that $\text{CSP}(\mathbb{C}_5)$ is polynomial-time solvable when the input is restricted to \mathbb{P}_8 -free graphs. Moreover, they show that there are finitely many \mathbb{P}_8 -free obstructions to $\text{CSP}(\mathbb{C}_5)$. Also, in [8] the authors prove that $\text{CSP}(\mathbb{K}_3)$ becomes tractable if the input is restricted to \mathbb{P}_7 -free graphs, while Huang [22] proved that $\text{CSP}(\mathbb{K}_4)$ remains NP-hard even for \mathbb{P}_7 -free graphs. In general, several complexity classifications are known for $\text{CSP}(\mathbb{K}_n)$ with input restriction to \mathbb{P}_k -free graphs (see, e.g., [18, Theorem 7]), however, a complete complexity classification of graph colouring problems with input restriction to \mathbb{F} -(subgraph)-free graphs remains wide open.

The finite domain CSP dichotomy [28] (announced independently by Bulatov [11] and Zhuk [27]) asserts that digraph colouring problems also exhibit a P versus NP-complete dichotomy. In this paper we consider the directed version of the research line described above, where we consider the following to be the long-term question of this variant: Is there a P versus NP-complete dichotomy of $\text{CSP}(\mathbb{H})$ where the input is restricted (1) to \mathbb{F} -free digraphs? and (2) to \mathbb{F} -subgraph-free digraphs?¹

The Sparse Incomparability Lemma [24] asserts that if $\text{CSP}(\mathbb{H})$ is NP-hard, then it remains NP-hard even for high-girth digraphs. Hence, both questions above have a positive answer whenever \mathbb{F} is not an oriented forest. Motivated by the literature on graph colouring problems restricted to \mathbb{P}_k -free and \mathbb{P}_k -subgraph-free digraphs, we consider the restriction of these questions to the case when F is a directed path $\vec{\mathbb{P}}_k$.

► **Question 1.** *Is there a P versus NP-complete dichotomy of $\text{CSP}(\mathbb{H})$ where the input is restricted*

1. *to $\vec{\mathbb{P}}_k$ -free digraphs?*
2. *to $\vec{\mathbb{P}}_k$ -subgraph-free digraphs?*

In our effort to settle Question 1 we stumble into three more questions which we also address in this paper. Allow us to elaborate. Clearly, if $\text{CSP}(\mathbb{H})$ is NP-hard even for $\vec{\mathbb{P}}_k$ -subgraph-free digraphs, then $\text{CSP}(\mathbb{H})$ is NP-hard for $\vec{\mathbb{P}}_k$ -free digraphs. In turn, if $\text{CSP}(\mathbb{H})$ restricted to $\vec{\mathbb{P}}_k$ -homomorphism-free digraphs is NP-hard, then $\text{CSP}(\mathbb{H})$ restricted to $\vec{\mathbb{P}}_k$ -subgraph-free digraphs is NP-hard as well. It is well-known that a digraph \mathbb{D} is $\vec{\mathbb{P}}_k$ -homomorphism-free if and only if \mathbb{D} homomorphically maps to the transitive tournament in $k - 1$ vertices \mathbb{T}_{k-1} (see, e.g, Observation 6). Hence, a simple way to find complexity lower bounds to the problems in Question 1 is to consider the complexity of $\text{CSP}(\mathbb{H})$ restricted to $\text{CSP}(\mathbb{T}_k)$.

The last problems have the following natural generalization: decide $\text{CSP}(\mathbb{H})$ restricted to input digraphs \mathbb{D} in $\text{CSP}(\mathbb{H}')$. We denote this problem by $\text{RCSP}(\mathbb{H}, \mathbb{H}')$. These problems have been called restricted homomorphism problems [10, 9] and it was conjectured by Hell

¹ Note that a negative answer to any of these questions implies that if $P \neq NP$, then there are finite digraphs \mathbb{F} and \mathbb{H} such that $\text{CSP}(\mathbb{H})$ restricted to \mathbb{F} -free digraphs is NP-intermediate – which we believe to be a more natural problem than the current NP-intermediate problems constructed in the literature. At this point we do not conjecture either a positive or a negative answer to the previous questions.

and Nešetřil that when \mathbb{H} and \mathbb{H}' are undirected graphs, then $\mathbb{H}' \rightarrow \mathbb{H}$, or \mathbb{H} is bipartite, and in these cases $\text{CSP}(\mathbb{H})$ restricted to $\text{CSP}(\mathbb{H}')$ is in P; and otherwise it is NP-hard. This was later confirmed by Brewster and Graves [9], where they actually propose a hardness condition for a broader family of digraphs \mathbb{H}' (see Theorem 5 below). It is then natural to ask our second main question.

► **Question 2.** *Is there a P versus NP-hard dichotomy of problems $\text{RCSP}(\mathbb{H}, \mathbb{H}')$ parametrized by finite digraphs (structures) \mathbb{H} and \mathbb{H}' ?*

Clearly, every digraph \mathbb{D} that admits a homomorphism to some transitive tournament must be an acyclic digraph. A natural question of a digraph \mathbb{H} , for which $\text{CSP}(\mathbb{H})$ is NP-complete, is to ask if this problem remains NP-complete on acyclic instances. Notice that this is the same problem as $\text{RCSP}(\mathbb{H}, (\mathbb{Q}, <))$. If \mathbb{H} is an undirected graph, then of course this will be true (choose any total order of the vertex set and orient the edges according to this ordering). However, when \mathbb{H} is an arbitrary digraph, the situation is not a priori clear. Indeed, it is addressed for some small digraphs by Hell and Mishra in [20]. They prove, for example, that $\text{CSP}(\vec{\mathbb{C}}_3^+)$ – where $\vec{\mathbb{C}}_3^+$ is drawn in the forthcoming Figure 4 – does indeed remain NP-complete on acyclic inputs. The general question is not posed in [20], but it is perfectly natural, and we address it here.

► **Question 3.** *If $\text{CSP}(\mathbb{H})$ is NP-hard for a finite graph \mathbb{H} , does $\text{CSP}(\mathbb{H})$ remain NP-hard for acyclic instances? Equivalently, is $\text{RCSP}(\mathbb{H}, (\mathbb{Q}, <))$ NP-hard whenever $\text{CSP}(\mathbb{H})$ is NP-hard?*

Some readers might have already noticed that there is a third natural question motivated by the previous paragraph: is there a P versus NP-complete dichotomy of $\text{CSP}(\mathbb{H})$ restricted to $\vec{\mathbb{P}}_k$ -homomorphism-free digraphs? Or more generally: is there a P versus NP-hard dichotomy of $\text{CSP}(\mathbb{H})$ where the input is restricted to \mathcal{F} -homomorphism-free digraphs? (where \mathcal{F} is a fixed finite set of digraphs). This question can already be settled from results in the literature: every such problem can be coded into *monotone monadic strict NP* (MMSNP), and Feder and Vardi [14] proved that if finite domain CSPs have a P versus NP-complete dichotomy, then MMSNP exhibits the same dichotomy. However, we provide an alternative prove of this fact in Section 5.

Story of the paper

Our paper splits into two parts. The first part focuses on relational structures, of which digraphs are somewhat canonical examples, while the second part focuses on digraphs specifically. In Figure 1 we depict the flow of ideas and results of this paper.

Within the first part (Sections 3–5), we begin by introducing *restricted CSPs* (RCSPs) which have been studied as restricted homomorphism problems in [10, 9]. In particular, we elaborate a connection with promise CSPs (PCSPs) that enables one to view RCSPs as PCSPs.

Our first main result shows that, for every pair of finite digraphs (structures), \mathbb{A} and \mathbb{B} the problem $\text{CSP}(\mathbb{B})$ with input restricted to $\text{CSP}(\mathbb{A})$ is either in P or NP-hard (settling Question 2). Moreover, this complexity classification is accompanied with an algebraic dichotomy (Theorem 18) which stems from the finite domain CSP dichotomy result. We push the previous complexity dichotomy to finite domain CSPs with restrictions in GMSNP (*guarded monotone strict NP*) [19, Section 4.4]. Another contribution of this work builds again on the finite domain CSP dichotomy to present a new proof of the complexity dichotomy for finite domain CSP restricted to \mathcal{F} -homomorphism-free digraphs (structures): we avoid going via MMSNP to the infinite domain CSP setting, and stay in the finite domain world via Lemma 21 (Section 5).

In the second part of the paper (Sections 6–9), we leverage results from the first part, in order to study digraph CSPs, where the ultimate focus will be on \mathbb{F} -free and \mathbb{F} -subgraph-free algorithmics.

Our second main result shows that, if \mathbb{H} is a digraph and $\text{CSP}(\mathbb{H})$ is NP-complete, then there is a positive integer N such that $\text{CSP}(\mathbb{H})$ remains NP-complete even for $\vec{\mathbb{P}}_N$ -subgraph-free acyclic inputs (settling Question 3). Moreover, N can be chosen so that $\text{CSP}(\mathbb{H})$ is polynomial-time solvable for $\vec{\mathbb{P}}_{N-1}$ -subgraph-free acyclic inputs. This also yields a partial answer to Question 1: for every digraph \mathbb{H} there is a positive integer $N \leq 4^{|H|}$ such that Question 1 has a positive answer restricted to $k \geq N$. We complement this general partial answer by settling Question 1 for digraphs \mathbb{H} on three vertices (Theorems 31 and 32), and for a family of smooth tournaments TC_n (Theorems 33 and Theorem 34). We note that eventual hardness on $\vec{\mathbb{P}}_N$ -subgraph-free instances does not hold in general for $\text{CSP}(\mathbb{H})$, if \mathbb{H} is an infinite digraph. We provide a counterexample (Example 27) which is otherwise well-behaved (for example, by being ω -categorical). As byproducts of our work we see that there are finitely many minimal $\vec{\mathbb{P}}_3$ -obstructions to $\text{CSP}(\text{TC}_n)$ for each positive integer n [19, Theorem 55], and if \mathbb{F} is not an oriented path, then $\text{CSP}(\text{TC}_n)$ (and $\text{CSP}(\vec{\mathbb{C}}_3^+)$) is NP-hard even for \mathbb{F} -subgraph-free instances (Theorem 36).

2 Preliminaries

2.1 Relational structures and digraphs

For a finite relational signature $\tau = (R_1, \dots, R_k)$, a *relational* (τ -)structure \mathbb{A} on domain A consists of k relations $R_1 \subseteq A^{a_1}, \dots, R_k \subseteq A^{a_k}$, where a_i is the arity of R_i . We denote the cardinality of A as $|A|$. We tend to conflate the relation symbol and actual relation since this will not introduce confusion.

For some signature τ , the *loop* \mathbb{L} is the structure on one element a all of whose relations are maximally full, that is, contains one tuple (a, \dots, a) – the structure \mathbb{L} clearly depends on the signature τ , but in this work τ will always be clear from context.

Directed graphs (digraphs) \mathbb{D} are relational structures on the signature $\{E\}$ where E is a binary relation. A digraph is a *graph* if E is symmetric, i.e. $(x, y) \in E$ iff $(y, x) \in E$. We will use the same blackboard font notation for digraphs as we do for relational structures.

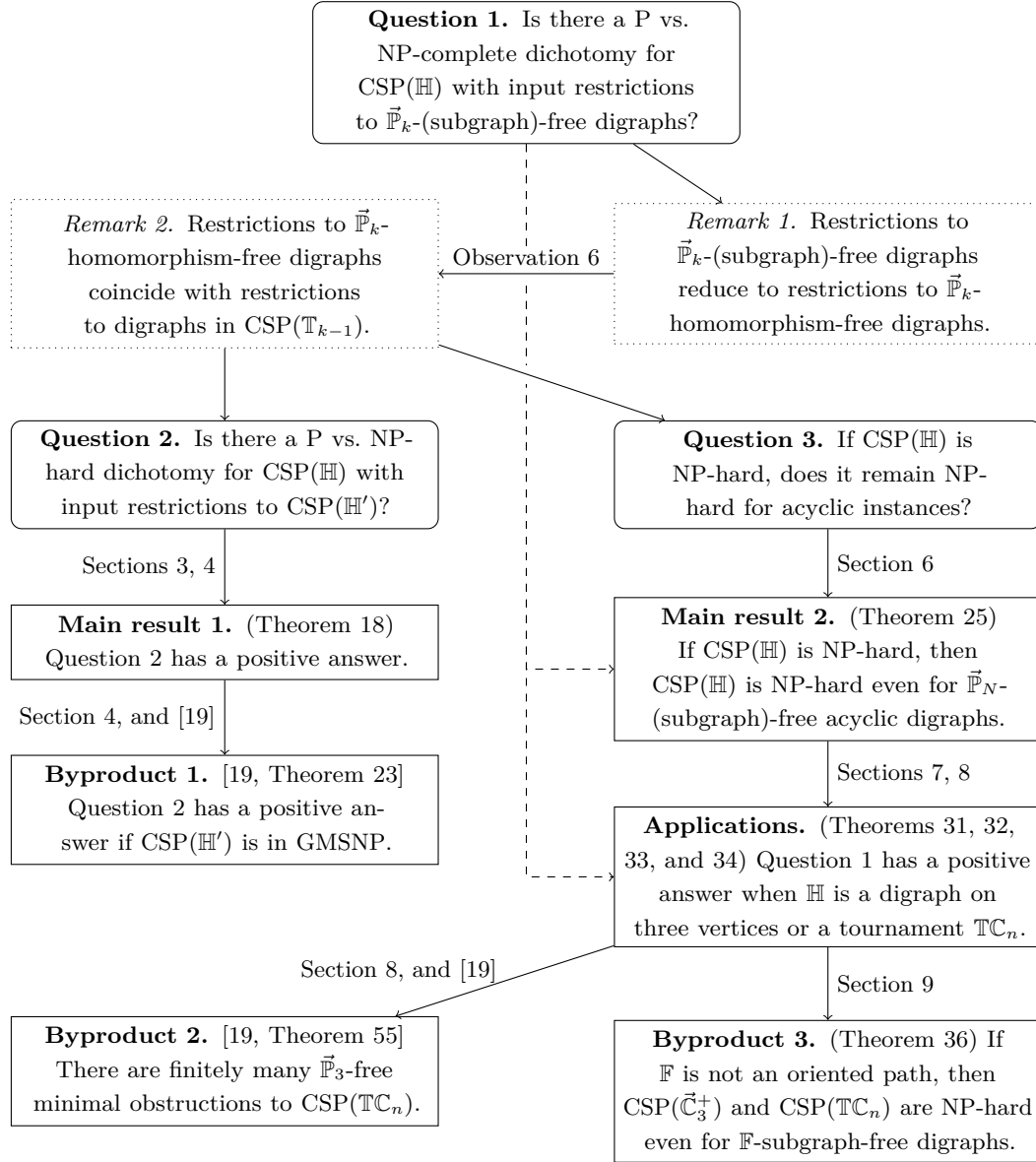
Given a positive integer k we denote by $\vec{\mathbb{C}}_k$ the directed cycle on k vertices, by $\vec{\mathbb{P}}_k$ the directed path on k vertices, by \mathbb{T}_k the transitive tournament on k vertices. Similarly we denote by \mathbb{K}_k the complete graph on k vertices, and we think of it as a digraph with edges (i, j) for every $i \neq j$.

A (*directed*) *walk* on a (directed) graph \mathbb{D} is a sequence of vertices x_1, \dots, x_k such that for every $i \in [k-1]$ there is a (directed) edge $(x_i, x_{i+1}) \in E$. An *oriented graph* is a digraph with no pair of symmetric edges, i.e., a loopless \mathbb{K}_2 -free digraph.

A *tree* is an oriented digraph whose underlying graph has no cycle (equivalently, is an orientation of a traditional undirected tree). It follows that trees are \mathbb{K}_2 -free. A *forest* is a disjoint union of trees. A structure \mathbb{A} is a *core* if all of its endomorphisms are self-embeddings. If \mathbb{A} is finite then this is equivalent to all endomorphisms being automorphisms.

2.2 Constraint satisfaction problems

Given a pair of digraphs \mathbb{D} and \mathbb{H} a *homomorphism* $f: \mathbb{D} \rightarrow \mathbb{H}$ is a vertex mapping such that, for every (u, v) that is an edge of \mathbb{D} , the image $(f(u), f(v))$ is an edge of \mathbb{H} . If such a homomorphism exists, we write $\mathbb{D} \rightarrow \mathbb{H}$, and otherwise $\mathbb{D} \not\rightarrow \mathbb{H}$. We follow the notation



■ **Figure 1** A depiction of the flow between main results and main questions addressed in this paper. Rectangles with rounded corners mark the main questions considered here. Dotted squares indicate the (simple) remarks connecting the \mathcal{F} -subgraph-free CSPs to the theory of digraph homomorphisms. Solid rectangles with straight corners indicate the main results of the present paper. A solid (resp. dashed) edge between a question and a result indicates that the result provides an answer (resp. a partial answer) to the corresponding question. Finally, edges between results represent that the result at the head is proved using tools introduced while proving the result at the tail of the corresponding edge.

of constraint satisfaction theory, and denote by $\text{CSP}(\mathbb{H})$ the class of finite digraphs such that $\mathbb{D} \rightarrow \mathbb{H}$. We also denote by $\text{CSP}(\mathbb{H})$ the computational problem of deciding if an input digraph \mathbb{D} belongs to $\text{CSP}(\mathbb{H})$.

The finite-domain CSP dichotomy is commonly stated in terms of *polymorphisms*. A *polymorphism* of a structure \mathbb{A} is a homomorphism $f: \mathbb{A}^n \rightarrow \mathbb{A}$, and in this case we say that f is an *n-ary* polymorphism. A 4-ary polymorphism $f: \mathbb{A}^4 \rightarrow \mathbb{A}$ satisfies the *Sigger's identity* if

$$f(x_1, x_2, x_3, x_1) = f(x_2, x_1, x_2, x_3) \text{ for every } x_1, x_2, x_3, x_4 \in A.$$

The Sigger's identity is one of several identities that equivalently describe the tractability frontier for finite domain CSPs.

► **Theorem 4** (Equivalent to Theorem 1.4 in [28]). *For every finite structure \mathbb{A} one of the following holds.*

- *Either \mathbb{A} has a polymorphism $f: \mathbb{A}^4 \rightarrow \mathbb{A}$ that satisfies the Sigger's identity, and in this case $\text{CSP}(\mathbb{A})$ is polynomial-time solvable, or*
- *otherwise, \mathbb{A} pp-constructs \mathbb{K}_3 , and in this case $\text{CSP}(\mathbb{A})$ is NP-complete.*

2.3 Smooth digraphs

A digraph \mathbb{D} is *smooth* if it has no sources nor sinks. A digraph \mathbb{H} is *hereditarily hard* if $\text{CSP}(\mathbb{H}')$ is NP-complete for every loopless digraph \mathbb{H}' such that $\mathbb{H} \rightarrow \mathbb{H}'$. Bang-Jensen, Hell, and Niven conjectured that a smooth digraph \mathbb{H} is hereditarily hard whenever \mathbb{H} does not homomorphically map to a disjoint union of directed cycles [2, Conjecture 2.5], and this was later confirmed by Barto, Kozik, and Niven proved in [4]. As far as we are aware, the most general result regarding hardness of restricted CSP problems is Theorem 3 in [9].

► **Theorem 5** (Theorem 3 in [9]). *If \mathbb{H} is a hereditarily hard digraph and \mathbb{H}' is a finite digraph such that $\mathbb{H}' \not\rightarrow \mathbb{H}$ then $\text{RCSP}(\mathbb{H}, \mathbb{H}')$ is NP-hard.*

2.4 Duality pairs

A pair of digraphs (relational structure) $(\mathbb{T}, \mathbb{D}_{\mathbb{T}})$ are called a *duality pair* if for every digraph \mathbb{D} it is the case that $\mathbb{D} \rightarrow \mathbb{D}_{\mathbb{T}}$ if and only if $\mathbb{T} \not\rightarrow \mathbb{D}$. It was proved in [25] that for every tree \mathbb{T} there is a digraph $\mathbb{D}_{\mathbb{T}}$ such that $(\mathbb{T}, \mathbb{D}_{\mathbb{T}})$ is a duality pair. Moreover, they also proved that if $(\mathbb{T}, \mathbb{D}_{\mathbb{T}})$ is a duality pair, then \mathbb{T} is homomorphically equivalent to a tree. A well-known example of a family of duality pairs is the following one.

► **Observation 6.** *For every positive integer k the directed path $\vec{\mathbb{P}}_{k+1}$ together with the transitive tournament \mathbb{T}_k are duality pair.*

Given a set of digraphs \mathcal{F} , we denote by $\text{Forb}(\mathcal{F})$ the set of digraphs \mathbb{D} such that $\mathbb{F} \not\rightarrow \mathbb{D}$ for every $\mathbb{F} \in \mathcal{F}$. It is straightforward to observe that for any such set \mathcal{F} , there is a (possibly infinite) digraph \mathbb{D} such that $\text{Forb}(\mathcal{F}) = \text{CSP}(\mathbb{D})$. A *generalized duality* is a pair $(\mathcal{F}, \mathcal{D})$ of finite sets of digraphs such that $\text{Forb}(\mathcal{F}) = \bigcup_{\mathbb{D} \in \mathcal{D}} \text{CSP}(\mathbb{D})$. When $\mathcal{D} = \{\mathbb{D}\}$ we simply write $(\mathcal{F}, \mathbb{D})$. In particular, for every set of trees \mathcal{F} there is a digraph \mathbb{D} such that $(\mathcal{F}, \mathbb{D})$ is a generalized duality [16, Theorem 11].

2.5 Large girth

A well-known result from Erdős [13] about k -colourability states that for every pair of positive integers l, k there is a graph G with girth strictly larger than l and such that G does not admit a proper k -colouring. This result generalizes to arbitrary relational structures, and it is known as the Sparse Incomprability Lemma [24] – in order to stay within the scope of this paper, we state it for digraphs.

Given a digraph (structure) \mathbb{D} , the *incidence graph* of \mathbb{D} is the undirected bipartite graph $\mathbb{I}(\mathbb{D})$ with vertex set $V \cup E$, for $v \in D$ a vertex of D and $e = (x, y) \in E$ an edge of D . There is an (undirected) edge (v, e) in $\mathbb{I}(\mathbb{D})$ if and only if $v \in \{x, y\}$. The *girth* of \mathbb{D} is half the length of the shortest cycle in $\mathbb{I}(\mathbb{D})$. Notice that if \mathbb{D} has a pair of symmetric arcs, its girth is 2, and otherwise, it is the graph theoretic girth of the underlying graph of \mathbb{D} .

► **Theorem 7** (Sparse Incomparability Lemma [24]). *For every digraph \mathbb{D} and every pair of positive integers k and ℓ there is a digraph \mathbb{D}' with the following properties:*

- $\mathbb{D}' \rightarrow \mathbb{D}$,
- the girth of \mathbb{D}' is larger than ℓ ,
- $\mathbb{D} \rightarrow \mathbb{H}$ if and only if $\mathbb{D}' \rightarrow \mathbb{H}$ for every digraph \mathbb{H} on at most k vertices,
- \mathbb{D}' can be constructed in polynomial time (from \mathbb{D}).

3 Restricted constraint satisfaction problems

Promise problems (not to be confused with Promise CSPs) can be thought as decision problems with input restrictions. Formally [26], a *promise problem* is a pair $(\mathcal{P}, \mathcal{C})$ of decidable sets. A solution to $(\mathcal{P}, \mathcal{C})$ is a decidable set \mathcal{S} such that $\mathcal{S} \cap \mathcal{P} = \mathcal{C} \cap \mathcal{P}$. We say that the promise problem $(\mathcal{P}, \mathcal{C})$ is polynomial-time solvable if it has a solution in P, and if every solution is NP-hard, we say that $(\mathcal{P}, \mathcal{C})$ is NP-hard.

Given a pair of (possibly infinite) structures \mathbb{A} and \mathbb{B} (with the same finite signature) whose CSPs are decidable, the *restricted CSP* $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is the promise problem $(\text{CSP}(\mathbb{B}), \text{CSP}(\mathbb{A}))$. In this case, we call (\mathbb{A}, \mathbb{B}) the *template* of the restricted CSP, \mathbb{A} is called the *domain* and \mathbb{B} the *restriction*. In particular, if \mathbb{A} is finite we say that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is a finite domain RCSP, and if \mathbb{B} is finite, we say that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is an RCSP with finite restriction.

Informally, the promise problem $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is $\text{CSP}(\mathbb{A})$ where the input is promised to belong to $\text{CSP}(\mathbb{B})$. For instance, $\text{RCSP}(\mathbb{K}_3, \mathbb{K}_4)$ is essentially the problem of deciding whether an input 4-colourable graph is 3-colourable.

Notice that for any digraph (structure) \mathbb{A} the problems $\text{CSP}(\mathbb{A})$ and $\text{RCSP}(\mathbb{A}, \mathbb{L})$ are the same problems where \mathbb{L} is the loop. So every decidable CSP is captured by an RCSP with finite restriction. One of the main results of this work shows that every RCSP with finite restriction is log-space equivalent to a CSP. It follows from the proof that every finite domain RCSP with finite restriction is log-space equivalent to a finite domain CSP, and thus, finite domain restricted CSPs with finite restrictions have a P versus NP-hard dichotomy. Moreover, the reduction mentioned in this paragraph are obtained by *restricted primitive positive construction* (*rpp-constructions*), which are the natural cousins of pp-constructions for CSPs [5] and for PCSPs [3].

3.1 Restricted primitive positive constructions

Several reductions in graph algorithmics arise from gadget reductions. For instance, a standard way of proving that $\text{CSP}(\mathbb{C}_5)$ is NP-complete can be done with the following gadget reduction from $\text{CSP}(\mathbb{K}_5)$. On input \mathbb{G} to $\text{CSP}(\mathbb{K}_5)$, consider the graph obtained from \mathbb{G} by replacing every edge $e := (x, y)$ by a path x, u_e, v_e, y (see, e.g., [21] where this is called an indicator construction). So in this case, the path on four vertices is the gadget associated to this reduction). The algebraic approach to CSPs proposes a general framework encompassing gadget reductions between constraint satisfaction problems. In this section we introduce primitive positive constructions, and *restricted* primitive positive constructions.

Primitive positive constructions

Given a pair of finite relational signature τ and σ , a *primitive positive definition* (of τ in σ)² of dimension $d \in \mathbb{Z}^+$ is a finite set Δ of primitive positive formulas $\delta_R(\bar{x})$ indexed by σ , and for each $R \in \sigma$ of arity r the formula $\delta_R(\bar{x})$ has $r \cdot d$ free variables.

For every primitive positive definition of σ in τ we associate a mapping Π_Δ from τ -structures to σ -structures as follows. Given a τ -structure \mathbb{A} the *pp power* $\Pi_\Delta(\mathbb{A})$ of \mathbb{A} is the structure

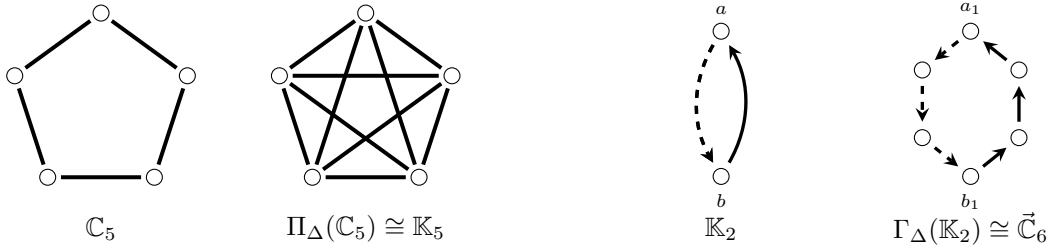
- with domain A^d , and
- for each $R \in \sigma$ of arity r the interpretation of R in $\Pi_\Delta(\mathbb{A})$ consists of the tuples $(\bar{a}_1, \dots, \bar{a}_r) \in (A^d)^r$ such that $\mathbb{A} \models \delta_R(\bar{a}_1, \dots, \bar{a}_r)$.

We say that a structure \mathbb{A} *pp-constructs* a structure \mathbb{B} if there is a primitive positive definition Δ such that $\Pi_\Delta(\mathbb{A}) \rightarrow \mathbb{B} \rightarrow \Pi_\Delta(\mathbb{A})$, i.e., $\Pi_\Delta(\mathbb{A})$ is homomorphically equivalent to \mathbb{B} . For instance, if Δ is the 1-dimensional primitive positive definition consisting of

$$\delta_E(x, y) := \exists z_1, z_2. E(x, z_1) \wedge E(z_1, z_2) \wedge E(z_2, y),$$

then the pp power $\Pi_\Delta(\mathbb{C}_5)$ of the 5-cycle is the complete graph \mathbb{K}_5 (see Figure 2).

► **Remark 8.** For every fixed primitive positive definition Δ , the pp-power Π_Δ is a monotone construction with respect to the homomorphism order, i.e., if $\mathbb{A} \rightarrow \mathbb{B}$, then $\Pi_\Delta(\mathbb{A}) \rightarrow \Pi_\Delta(\mathbb{B})$ (because existential positive formulas are preserved by homomorphism [6, Theorem 2.5.2]).



■ **Figure 2** Let Δ be the primitive positive definition (of $\{E\}$ in $\{E\}$) where $\delta_E(x, y) := \exists z_1, z_2. E(x, z_1) \wedge E(z_1, z_2) \wedge E(z_2, y)$. On the left, we depict \mathbb{C}_5 and its pp power $\Pi_\Delta(\mathbb{C}_5) \cong \mathbb{K}_5$ (an undirected edge from x to y represents (x, y) and (y, x)), and on the right, we depict \mathbb{K}_2 and its gadget replacement $\Gamma_\Delta(\mathbb{K}_2) \cong \vec{\mathbb{C}}_6$ (dashed edges and solid edges indicate the respective edge replacements).

Gadget replacements

For every primitive positive formula $\delta(\bar{x})$ without equalities³ we construct a structure $\mathbb{D}_\delta(\bar{d})$ with a distinguished tuples of vertices \bar{d} as follows.

- The domain D of $\mathbb{D}_\delta(\bar{d})$ consists of a vertex v_y for every (free or bounded) variable y of δ .
- The distinguished vertices d_1, \dots, d_m are the vertices v_{x_1}, \dots, v_{x_m} .
- For every relation symbol R there is a tuple $\bar{v} \in R^{\mathbb{D}_\delta(\bar{d})}$ if and only if δ contains the conjunct $R(\bar{y})$ where \bar{v} is the tuple of vertices corresponding to the tuple of variables \bar{y} .

² When the signatures are clear from context we will simply say talk about a primitive positive definition.

³ If a primitive positive formula contains a conjunct $x = y$, we obtain an equivalent formula δ' by deleting the conjunct $x = y$ and replacing each occurrence of the variable y by the variable x .

The structure $\mathbb{D}_\delta(\bar{d})$ is sometimes called the *canonical database* of δ . For instance, if δ_E is the formula considered above, the canonical database of δ_E is the directed with vertices 1, 2, 3, 4, and the distinguished vertices are 1 and 4.

The canonical database $\mathbb{D}_\delta(\bar{d})$ and the primitive positive formula δ are closely related: for every structure \mathbb{A} and a tuple \bar{a} of elements of A

$$\mathbb{A} \models \delta(\bar{a}) \text{ if and only if there is a homomorphism } f: \mathbb{D}_\delta(\bar{d}) \rightarrow \mathbb{A} \text{ such that } f(\bar{d}) = \bar{a}$$

(see, e.g., [6, Proposition 1.2.5]).

Building on canonical databases, for every d -dimensional primitive positive definition Δ of σ in τ we associate a mapping Γ_Δ from σ -structures to τ -structures as follows. Given a σ -structure \mathbb{B} the *gadget replacement* $\Gamma_\Delta(\mathbb{B})$ is the τ -structure obtained from \mathbb{B} where

- for each vertex $b \in B$ we introduce d vertices b_1, \dots, b_d , and
- for each $R \in \sigma$ of arity r , and every tuple $(b^1, \dots, b^r) \in R^\mathbb{B}$ we introduce a fresh copy of $\mathbb{D}_{\delta_R}(\bar{d}^1, \dots, \bar{d}^r)$ and we identify each d -tuple \bar{d}^i with (b^i_1, \dots, b^i_d) .

Going back to our on going example $\Delta := \{\delta_E\}$ and taking $\mathbb{B} := \mathbb{K}_2$, the gadget replacement $\Gamma_\Delta(\mathbb{B})$ is isomorphic to the directed 6-cycle \mathbb{C}_6 (see Figure 2 for a depiction).

It is straightforward to observe that, in general, for every primitive positive definition Δ , the gadget replacement $\Gamma_\Delta(\mathbb{B})$ can be constructed in logarithmic space from \mathbb{B} . Also, these mapping behave like adjoints (see, e.g., Observation 4.4 in [23]).

Log-space reductions and rpp-constructions

We say that a (not necessarily finite) restricted CSP template (\mathbb{A}, \mathbb{B}) *rpp-constructs* a restricted CSP template $(\mathbb{A}', \mathbb{B}')$ if there is a primitive positive definition Δ such that

$$(\Pi_\Delta(\mathbb{A}) \times \mathbb{B}') \leftrightarrow (\mathbb{A}' \times \mathbb{B}') \text{ and } \mathbb{B}' \rightarrow \Pi_\Delta(\mathbb{B}).$$

► **Remark 9.** Every primitive positive formula is satisfiable in some finite structure, e.g., in the loop \mathbb{L} . This implies that for every primitive positive definition of σ in τ the pp-power $\Pi_\Delta(\mathbb{L}_\tau)$ of the (τ) -loop is homomorphically equivalent to the (σ) -loop \mathbb{L}_σ . In particular, this implies that the following statements are equivalent for every τ -structure \mathbb{A} and every σ -structure \mathbb{A}' .

- \mathbb{A} pp-constructs \mathbb{A}' .
- $(\mathbb{A}, \mathbb{L}_\tau)$ rpp-constructs $(\mathbb{A}', \mathbb{L}_\sigma)$.
- $(\mathbb{A}, \mathbb{L}_\tau)$ rpp-constructs $(\mathbb{A}', \mathbb{B}')$ for ever structure \mathbb{B}' .

As mentioned above, pp-constructions compose, and building on this fact, it is straightforward to observe that rpp-constructions compose (see also [19, Lemma 13]).

► **Lemma 10.** *Consider three restricted CSP templates $(\mathbb{A}_1, \mathbb{B}_1)$, $(\mathbb{A}_2, \mathbb{B}_2)$, and $(\mathbb{A}_3, \mathbb{B}_3)$. If $(\mathbb{A}_1, \mathbb{B}_1)$ rpp-constructs $(\mathbb{A}_2, \mathbb{A}_2)$, and $(\mathbb{A}_2, \mathbb{B}_2)$ rpp-constructs $(\mathbb{A}_3, \mathbb{B}_3)$, then $(\mathbb{A}_1, \mathbb{B}_1)$ rpp-constructs $(\mathbb{A}_3, \mathbb{B}_3)$.*

Also, rpp-constructions yield log-space reductions between RCSPs as pp-constructions do for CSPs (the reader can find a proof in [19, Lemma 14]).

► **Lemma 11.** *Consider two (not necessarily finite) restricted CSP templates (\mathbb{A}, \mathbb{B}) and $(\mathbb{A}', \mathbb{B}')$. If (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{A}', \mathbb{B}')$, then there is a log-space reduction from $\text{RCSP}(\mathbb{A}', \mathbb{B}')$ to $\text{RCSP}(\mathbb{A}, \mathbb{B})$.*

► **Example 12.** Observe that the restricted CSP template $(\mathbb{C}_5, \mathbb{K}_3)$ rpp-constructs the template $(\mathbb{K}_5, \mathbb{L})$ via the formula

$$\delta_E(x, y) := \exists z_1, z_2. E(x, z_1) \wedge E(z_1, z_2) \wedge E(z_2, y).$$

Indeed, we already noticed that the pp power $\Pi_\Delta(\mathbb{C}_5)$ is isomorphic to \mathbb{K}_5 . It is also not hard to observe that the pp power $\Pi_\Delta(\mathbb{K}_5)$ is homomorphically equivalent to the loop \mathbb{L} . Hence, by Lemma 11 we conclude that $\text{RCSP}(\mathbb{C}_5, \mathbb{K}_3)$ is NP-hard, i.e., deciding if a 3-colourable graph \mathbb{G} admits a homomorphism to \mathbb{C}_5 is NP-hard.

4 Finite domain restrictions

In this section we show that for every restricted CSP template (\mathbb{A}, \mathbb{B}) with finite restriction, there is a structure \mathbb{C} such that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ and $\text{CSP}(\mathbb{C})$ are log-space equivalent. Moreover, if \mathbb{A} is also a finite structure, then \mathbb{C} is a finite structure. It thus follows from the finite domain dichotomy [11, 27] that restricted CSPs with finite domain and finite restriction, have a P versus NP-complete dichotomy.

4.1 Exponential structures

Given a pair of τ -structures \mathbb{A} and \mathbb{B} the *exponential* $\mathbb{A}^\mathbb{B}$ is the τ -structure

- with vertex set all functions $f: B \rightarrow A$, and
- for each $R \in \tau$ of arity r there is an r -tuple (f_1, \dots, f_r) belongs to the interpretation of R in $\mathbb{A}^\mathbb{B}$ if and only if $(f_1(b_1), \dots, f_r(b_r)) \in R^\mathbb{A}$ whenever $(b_1, \dots, b_r) \in R^\mathbb{B}$.

In Figure 3 we depict an exponential digraph construction. This image is also found in [21], where the reader can also find further discussion and properties; we list two of them (see also [15, Corollary 1.5.12]).

► **Lemma 13.** *The following statements hold for all τ -structures \mathbb{A}, \mathbb{B} and \mathbb{C} .*

- $\mathbb{A}^\mathbb{L}$ is isomorphic to \mathbb{A} .
- $\mathbb{C} \rightarrow \mathbb{A}^\mathbb{B}$ if and only if $\mathbb{C} \times \mathbb{B} \rightarrow \mathbb{A}$.

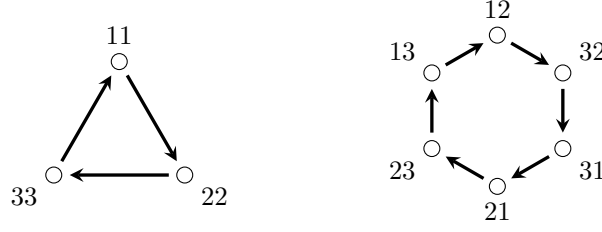
► **Corollary 14.** *For every pair of (possibly infinite) τ -structures \mathbb{A} and \mathbb{B} , the restricted CSP template $(\mathbb{A}^\mathbb{B}, \mathbb{L})$ rpp-constructs the template (\mathbb{A}, \mathbb{B}) .*

Proof. Let Δ be the trivial primitive positive definition of τ in τ , i.e., for each $R \in \tau$ or arity r let $\delta_R(x_1, \dots, x_r) := R(x_1, \dots, x_r)$. So for every structure \mathbb{A} the equality $\Pi_\Delta(\mathbb{A}) = \mathbb{A}$ holds. Clearly, $\mathbb{B} \rightarrow \mathbb{L} = \Pi_\Delta(\mathbb{L})$, and notice that $\mathbb{A}^\mathbb{B} \times \mathbb{B}$ is homomorphically equivalent to $\mathbb{A} \times \mathbb{B}$: since $\mathbb{A}^\mathbb{B} \rightarrow \mathbb{A}^\mathbb{B}$, it follows from the second item in Lemma 13 that $\mathbb{A}^\mathbb{B} \times \mathbb{B} \rightarrow \mathbb{A}$, and clearly $\mathbb{A}^\mathbb{B} \times \mathbb{B} \rightarrow \mathbb{B}$, so $\mathbb{A}^\mathbb{B} \times \mathbb{B} \rightarrow \mathbb{A} \times \mathbb{B}$; conversely, $(\mathbb{A} \times \mathbb{B}) \times \mathbb{B} \rightarrow \mathbb{A}$, so by Lemma 13, $\mathbb{A} \times \mathbb{B} \rightarrow \mathbb{A}^\mathbb{B}$, and since $\mathbb{A} \times \mathbb{B} \rightarrow \mathbb{B}$, it follows that $\mathbb{A} \times \mathbb{B} \rightarrow \mathbb{A}^\mathbb{B} \times \mathbb{B}$. Therefore, Δ is a primitive positive definition witnessing that the claim of the lemma holds. ◀

► **Lemma 15.** *Let τ be a finite relational signature, and \mathbb{A} be a (possibly infinite) τ -structure. If \mathbb{B} is a finite τ -structure, then the restricted RCSP template (\mathbb{A}, \mathbb{B}) rpp-constructs the RCSP template $(\mathbb{A}^\mathbb{B}, \mathbb{L})$.*

Proof. We first show that (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{A}^\mathbb{B}, \mathbb{L})$, and we consider the case of digraphs but it naturally generalizes to τ -structures for finite τ . Let $B = \{b_1, \dots, b_m\}$ and consider the following m -dimensional primitive positive definition $\Delta := \{\delta_E\}$, where

$$\delta_E(x_1, \dots, x_m, y_1, \dots, y_m) := \bigwedge_{(b_i, b_j) \in E^\mathbb{B}} E(x_i, y_j).$$



■ **Figure 3** The exponential $\vec{\mathbb{C}}_3^{\mathbb{K}_2}$ where a label ij represents the function $f: \{1, 2\} \rightarrow \{1, 2, 3\}$ defined by $1 \mapsto i$ and $2 \mapsto j$.

Notice that for any structure \mathbb{A}' the pp-power Π_Δ is isomorphic to $\mathbb{A}'^{\mathbb{B}}$. Indeed, identify a tuple (a_1, \dots, a_m) of A with the function defined by $b_i \mapsto a_i$, it now follows from the definition of the exponential structure and of δ_E that this mapping is an isomorphism. In particular, notice that $\Pi_\Delta(\mathbb{B}) = \mathbb{B}^{\mathbb{B}}$ contains the loop \mathbb{L} (the identity function $I: B \rightarrow B$ is a loop on $\mathbb{B}^{\mathbb{B}}$). Hence,

$$\Pi_\Delta(\mathbb{A}) \cong \mathbb{A}^{\mathbb{B}} \text{ and } \mathbb{L} \rightarrow \Pi_\Delta(\mathbb{B}),$$

i.e., Δ is a witness to the fact that (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{A}^{\mathbb{B}}, \mathbb{L})$. ◀

We discuss two applications of this lemma. The first one being that rpp-constructions between RCSP templates with finite restrictions are captured by (standard) pp-constructions.

► **Theorem 16.** *The following statements are equivalent for every pair of restricted CSP templates $(\mathbb{A}_1, \mathbb{B}_1)$ and $(\mathbb{A}_2, \mathbb{B}_2)$ with finite restrictions.*

- $(\mathbb{A}_1, \mathbb{B}_1)$ rpp-constructs $(\mathbb{A}_2, \mathbb{B}_2)$ and,
- $\mathbb{A}_1^{\mathbb{B}_1}$ pp-constructs $\mathbb{A}_2^{\mathbb{B}_2}$.

In particular, $(\mathbb{A}_1, \mathbb{B}_1)$ and $(\mathbb{A}_1^{\mathbb{B}_1}, \mathbb{L})$ are mutually rpp-constructible (and so $\text{RCSP}(\mathbb{A}_1, \mathbb{B}_1)$ and $\text{CSP}(\mathbb{A}_1^{\mathbb{B}_2})$ are log-space Turing-equivalent).

Proof. Suppose that $\mathbb{A}_1^{\mathbb{B}_1}$ pp-constructs $\mathbb{A}_2^{\mathbb{B}_2}$, so by Remark 9 the template $(\mathbb{A}_1^{\mathbb{B}_1}, \mathbb{L})$ rpp-constructs $(\mathbb{A}_2^{\mathbb{B}_2}, \mathbb{L})$. By Lemma 15, $(\mathbb{A}_1, \mathbb{B}_1)$ rpp-constructs $(\mathbb{A}_1^{\mathbb{B}_1}, \mathbb{L})$, and by Corollary 14 $(\mathbb{A}_2^{\mathbb{B}_2}, \mathbb{L})$ rpp-constructs $(\mathbb{A}_2, \mathbb{B}_2)$. Hence, by composing rpp-constructions (Lemma 10), we conclude that $(\mathbb{A}_1, \mathbb{B}_1)$ rpp-constructs $(\mathbb{A}_2, \mathbb{B}_2)$. The converse implication follows with similar arguments: $(\mathbb{A}_2, \mathbb{B}_2)$ rpp-constructs $(\mathbb{A}_2^{\mathbb{B}_2}, \mathbb{L})$ (Lemma 15), and $(\mathbb{A}_1^{\mathbb{B}_1}, \mathbb{L})$ rpp-constructs $(\mathbb{A}_1, \mathbb{B}_1)$ (Corollary 14); by composing rpp-constructions we see that $(\mathbb{A}_1^{\mathbb{B}_1}, \mathbb{L})$ rpp-constructs $(\mathbb{A}_2^{\mathbb{B}_2}, \mathbb{L})$, and it thus follows that $\mathbb{A}_1^{\mathbb{B}_1}$ pp-constructs $\mathbb{A}_2^{\mathbb{B}_2}$ (Remark 9). ◀

The second application of Lemma 15 is the following statement which is analogous to the case of CSPs and pp-constructions (see, e.g., [6, Theorem 3.2.2]).

► **Corollary 17.** *The following statements are equivalent for every (possibly infinite) structure \mathbb{A} and a finite structure \mathbb{B} with a finite signature.*

- The restricted CSP template (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{K}_3, \mathbb{L})$.
- The restricted CSP template (\mathbb{A}, \mathbb{B}) rpp-constructs every restricted CSP template $(\mathbb{A}', \mathbb{B}')$ where \mathbb{A}' is a finite structure (and \mathbb{B}' a possibly infinite structure).
- The structure $\mathbb{A}^{\mathbb{B}}$ pp-constructs \mathbb{K}_3 .

If any of these equivalent statements hold, then $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is NP-hard.

Proof. By the first item of Lemma 13 we know that $\mathbb{K}_3^{\mathbb{L}}$ is isomorphic to \mathbb{K}_3 , and so $\mathbb{A}^{\mathbb{B}}$ pp-constructs \mathbb{K}_3 if and only if $\mathbb{A}^{\mathbb{B}}$ pp-constructs $\mathbb{K}_3^{\mathbb{L}}$. Hence, the equivalence between the first and third itemized statement follows from Theorem 16. Also, the second statement clearly implies the third one. Finally, we show that the first item implies the second one. It is well-known that \mathbb{K}_3 pp-constructs every finite structure \mathbb{A}' (see, e.g., [6, Corollary 3.2.1]). So, by Remark 9 $(\mathbb{K}_3, \mathbb{L})$ rpp-constructs every finite domain RCSP template $(\mathbb{A}', \mathbb{B}')$ (where \mathbb{B}' is a possibly infinite structure). Again, by composing rpp-constructions, we conclude that (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{A}', \mathbb{B}')$. The equivalence between the three itemized statement is now settled. The final statement holds because pp-construction yield log-space reduction (see, e.g., Corollary 3.5 in [5]), and because $\text{RCSP}(\mathbb{K}_3, \mathbb{L}) = \text{CSP}(\mathbb{K}_3)$ is NP-hard. ◀

4.2 A dichotomy for finite domain RCSPs

The dichotomy for finite domain CSPs asserts that if \mathbb{A} is a finite structure, then either \mathbb{A} pp-constructs \mathbb{K}_3 , and in this case $\text{CSP}(\mathbb{A})$ is NP-complete; or otherwise, $\text{CSP}(\mathbb{A})$ is polynomial-time solvable. We apply the results of this section to obtain an analogous (actually, equivalent) statement for finite domain CSPs with finite restrictions.

- **Theorem 18.** *For every pair of finite structures \mathbb{A}, \mathbb{B} one of the following statements hold.*
- *Either (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{K}_3, \mathbb{L})$ (and consequently, $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is NP-hard), or*
 - *$\text{RCSP}(\mathbb{A}, \mathbb{B})$ is polynomial-time solvable.*

Proof. Suppose that the first itemized statement does not hold, and assume $P \neq NP$ (otherwise, $\text{CSP}(\mathbb{A})$ is in P, and thus $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is polynomial-time solvable). Since (\mathbb{A}, \mathbb{B}) does not rpp-construct $(\mathbb{K}_3, \mathbb{L})$, it follows from Corollary 17 that $\mathbb{A}^{\mathbb{B}}$ does not pp-construct \mathbb{K}_3 . Hence, the finite domain dichotomy implies that $\text{CSP}(\mathbb{A}^{\mathbb{B}})$ is in P. By the “in particular” statement of Theorem 16 we know that $\text{CSP}(\mathbb{A}^{\mathbb{B}})$ and $\text{RCSP}(\mathbb{A}, \mathbb{B})$ are polynomial-time equivalent, and we thus conclude that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is polynomial-time solvable. ◀

RCSPs also make sense in the infinite setting, but this diverges from the main axis of this paper. Regarding finite domain RCSPs with infinite restrictions, the dichotomy from Theorem 18 can be leveraged into a dichotomy for finite-domain with certain tame infinite restrictions [19, Theorem 23]. Regarding infinite domain RCSPs we have little hope since infinite domain CSPs remain far from being understood. However, the tractability conjecture is a generalization of the Feder-Vardi conjecture to a broad class of “well-behaved” infinite structures. In the full version of this paper, we see that the tractability conjecture has an equivalent statement in terms of RCSPs of “well-behaved” infinite structures with finite restrictions [19, Theorem 24].

4.3 A small remark regarding PCSPs

Theorem 16 asserts in particular that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is log-space equivalent to $\text{CSP}(\mathbb{A}^{\mathbb{B}})$. The following statement strengthens this connection by showing that these two problems are further log-space equivalent to $\text{PCSP}(\mathbb{A}, \mathbb{A}^{\mathbb{B}})$ – the proofs are quite straightforward, and we leave them for the full version [19, Section 4.6].

- **Lemma 19.** *For every possibly infinite structure \mathbb{A} and a (possibly infinite) structure and \mathbb{B} a finite structure, the following problems are polynomial-time equivalent.*
- *the constraint satisfaction problem $\text{CSP}(\mathbb{A}^{\mathbb{B}})$,*
 - *the promise constraint satisfaction problem $\text{PCSP}(\mathbb{A}, \mathbb{A}^{\mathbb{B}})$, and*
 - *the restricted constraint satisfaction problem $\text{RCSP}(\mathbb{A}, \mathbb{B})$.*

► **Corollary 20.** *For every non-bipartite graph \mathbb{H} and every finite digraph \mathbb{D} one of the following holds.*

- *Either $\mathbb{D} \rightarrow \mathbb{H}$, and in this case $\text{PCSP}(\mathbb{H}, \mathbb{H}^{\mathbb{D}})$ is polynomial-time solvable,*
- *or $\mathbb{D} \not\rightarrow \mathbb{H}$, and in this case $\text{PCSP}(\mathbb{H}, \mathbb{H}^{\mathbb{D}})$ is NP-hard.*

5 Homomorphism-free restrictions

As mentioned above, the reader familiar with monotone monadic strict NP (MMSNP) can notice that digraph (finite domain) CSPs with input restriction to \mathcal{F} -homomorphism-free digraphs (structures) can be solved by infinite domain CSPs expressible in MMSNP. Thus, these problems exhibit a P versus NP-complete dichotomy (see, e.g., [7, 14]). Feder and Vardi reduce problems in MMSNP to finite domain CSPs. Their reduction changes the input signature, and then uses the Sparse Incomparability Lemma. Here, we prove the following lemma which allows us to preserve the signature by using duality pairs, and then, as Feder and Vardi, we proceed via Sparse Incomparability (see Section 5 in [19]).

► **Lemma 21.** *For every finite set of structures \mathcal{F} and every finite digraph (structure) \mathbb{B} , there is a finite set of structures \mathcal{C} such that the following problems are polynomial-time equivalent:*

- *deciding if an input structure \mathbb{A} belongs to $\text{CSP}(\mathcal{C})$ for some $\mathbb{C} \in \mathcal{C}$, and*
- *$\text{CSP}(\mathbb{B})$ restricted to \mathcal{F} -homomorphism-free structures.*

The next theorem now follows directly from Lemma 21 and the finite domain CSP dichotomy [28].

► **Theorem 22.** *For every finite digraph (structure) \mathbb{H} and every finite set of digraphs (structures) \mathcal{F} , $\text{CSP}(\mathbb{H})$ restricted to \mathcal{F} -homomorphism-free digraphs (structures) is either in P or NP-complete.*

6 Acyclic digraphs and bounded paths

In this section we apply our results above, and the theory of constraint satisfaction to obtain results in the context of \mathcal{F} -(subgraph)-free algorithmics. The main result of this section settles Question 3 (Theorem 25).

Consider a finite digraph \mathbb{H} , and let k be a positive integer. For a given function $f: (H \times [k])^n \rightarrow (H \times [k])$ and for each $i \in [k]$ we define a function

$$f_i: H^n \rightarrow H \text{ by } (h_1, \dots, h_n) \mapsto \pi_H(f((h_1, i), \dots, (h_n, i))),$$

where π_H is the projection of $H \times [k]$ onto H .

► **Lemma 23.** *Let \mathbb{H} be a digraph, k a positive integer, and $f: (\mathbb{H} \times \mathbb{T}_k)^n \rightarrow (\mathbb{H} \times \mathbb{T}_k)$ and n -ary polymorphism of $\mathbb{H} \times \mathbb{T}_k$. If there are $i, j \in [k]$ with $i < j$ such that $f_i = f_j$, then f_i is an n -ary polymorphism of \mathbb{H} , and if f satisfies the equalities*

$$f(x_{\sigma(1)}, \dots, x_{\sigma(n)}) = f(x_{\rho(1)}, \dots, x_{\rho(n)}) \text{ for all } x_1, \dots, x_n \in H \times [k]$$

for some $\sigma, \rho: [n] \rightarrow [m]$, then f_i satisfies the equalities

$$f_i(h_{\sigma(1)}, \dots, h_{\sigma(n)}) = f_i(h_{\rho(1)}, \dots, h_{\rho(n)}) \text{ for all } h_1, \dots, h_n \in H.$$

Proof. We first see that f_i is an n -ary polymorphism of \mathbb{H} . Let $((h_1, \dots, h_n), (h'_1, \dots, h'_n))$ be an edge of \mathbb{H}^n , i.e., (h_m, h'_m) is an edge for \mathbb{H} for each $m \in [n]$. Since $i < j$, it follows that $((h_1, i), \dots, (h_n, i)), ((h'_1, j), \dots, (h'_n, j))$ is an edge of $\mathbb{H} \times \mathbb{T}_n$. Since f is a polymorphism, it must be the case that $(f((h_1, i), \dots, (h_n, i)), f((h'_1, j), \dots, (h'_n, j)))$ is an edge of $\mathbb{H} \times \mathbb{T}_n$. Recall that the projection π_H is a homomorphism from $\mathbb{H} \times \mathbb{T}_n$ onto \mathbb{H} , and so

$$(f_i(h_1, \dots, h_n), f_j(h'_1, \dots, h'_n)) = (\pi_H f((h_1, i), \dots, (h_n, i)), \pi_H f((h'_1, j), \dots, (h'_n, j))) \in E(\mathbb{H}).$$

Finally, since $f_i = f_j$, it follows that $(f_i(h_1, \dots, h_n), f_j(h_1, \dots, h_n))$ is an edge of \mathbb{H} , and thus $f_i: \mathbb{H}^n \rightarrow \mathbb{H}$ is an n -ary polymorphism of \mathbb{H} .

Now, let $h_1, \dots, h_m \in H$, and for each $l \in [n]$ let $\bar{h}_l := (h_l, i)$, so

$$f_i(h_{\sigma(1)}, \dots, h_{\sigma(n)}) = \pi_H(f(\bar{h}_{\sigma(1)}, \dots, \bar{h}_{\sigma(n)})).$$

Since f satisfies the loop condition for σ and ρ , it follows that

$$\pi_H(f(\bar{h}_{\sigma(1)}, \dots, \bar{h}_{\sigma(n)})) = \pi_H(f(\bar{h}_{\rho(1)}, \dots, \bar{h}_{\rho(n)})) = f_i(h_{\rho(1)}, \dots, h_{\rho(n)}).$$

And thus, $f_i(h_{\sigma(1)}, \dots, h_{\sigma(n)}) = f_i(h_{\rho(1)}, \dots, h_{\rho(n)})$, and since the choice of $h_1, \dots, h_m \in H$ was arbitrary, the claim of this lemma follows. \blacktriangleleft

► **Theorem 24.** *For every finite digraph \mathbb{H} one of the following statements holds.*

- *Either \mathbb{H} has a polymorphism satisfying the Sigger's identity, and $\text{RCSP}(\mathbb{H}, \mathbb{T}_k)$ is in P for every positive integer k , or*
- *otherwise, $\text{RCSP}(\mathbb{H}, \mathbb{T}_k)$ is NP-hard for every large enough k .*

Proof. The first item holds by Theorem 4, and because if $\text{CSP}(\mathbb{H})$ is in P, then $\text{RCSP}(\mathbb{H}, \mathbb{T}_k)$ is in P for every positive integer k . Now, suppose that \mathbb{H} does not have a polymorphism satisfying the Sigger's identity. Since $\text{CSP}(\mathbb{T}_k)$ can be verified in polynomial-time (see, e.g., Observation 6), it follows that $\text{RCSP}(\mathbb{H}, \mathbb{T}_k)$ and $\text{CSP}(\mathbb{H} \times \mathbb{T}_k)$ are polynomial-time equivalent. Let $M = |H|^{4|H|} + 1$, and consider a polymorphism $f: (\mathbb{H} \times \mathbb{T}_M)^4 \rightarrow \mathbb{H}$. Clearly, there are $M - 1$ functions from H^4 to H , and since for each $i \in [M]$ every f_i is a 4-ary function of H , it follows from the choice of M that there are $i < j \leq M$ such that $f_i = f_j$. So, by Lemma 23 if f satisfies the Sigger's identity, then f_i satisfies the Sigger's identity. Since \mathbb{H} does not have such a polymorphism, then $\mathbb{H} \times \mathbb{T}_M$ does not have such a polymorphism, and hence $\text{CSP}(\mathbb{H} \times \mathbb{T}_k)$ is NP-hard (see, e.g., Theorem 4). Moreover, since $\text{CSP}(\mathbb{H} \times \mathbb{T}_M)$ is polynomial-time equivalent to $\text{RCSP}(\mathbb{H}, \mathbb{T}_M)$, it follows that $\text{RCSP}(\mathbb{H}, \mathbb{T}_M)$ is NP-hard. Finally, it is straightforward to observe that if $\text{RCSP}(\mathbb{H}, \mathbb{T}_n)$ is at least as hard as $\text{RCSP}(\mathbb{H}, \mathbb{T}_{n+1})$, and so, if N is the smallest integer such that $\mathbb{H} \times \mathbb{T}_N$ does not have a Sigger's polymorphism, then N witnesses that the second itemized statement holds. \blacktriangleleft

As promised, we apply the framework of RCSPs to the context of \mathcal{F} -(subgraph)-free algorithmics – recall that the class of $\vec{\mathbb{P}}_k$ -subgraph-free acyclic digraphs equals the class of $\vec{\mathbb{P}}_k$ -homomorphism-free digraphs.

► **Theorem 25.** *For every digraph \mathbb{H} such that $\text{CSP}(\mathbb{H})$ is NP-hard, there is a positive integer $N \leq |H|^{4|H|} + 1$ such that $\text{CSP}(\mathbb{H})$ remains NP-complete even for $\vec{\mathbb{P}}_N$ -subgraph-free acyclic digraphs. Moreover, there is such an N such that $\text{CSP}(\mathbb{H})$ is polynomial-time solvable when the input is a $\vec{\mathbb{P}}_{N-1}$ -subgraph-free acyclic digraphs.*

Proof. If $\text{P} = \text{NP}$ the claim is trivial, so we assume that $\text{P} \neq \text{NP}$. Notice that a digraph \mathbb{D} is acyclic and has no directed path on N vertices if and only if there is no homomorphism

$\vec{\mathbb{P}}_N \rightarrow \mathbb{D}$. Also, since $\text{CSP}(\mathbb{H})$ is NP-complete and we are assuming that $P \neq NP$, it follows from Theorem 4 that \mathbb{H} has no Siggers polymorphism. The claim of this theorem now follows from these simple arguments and Theorem 24. \blacktriangleleft

► **Corollary 26.** *Let \mathbb{H} be a digraph such that $\text{CSP}(\mathbb{H})$ is NP-hard. Then, there are positive integers N and M such that*

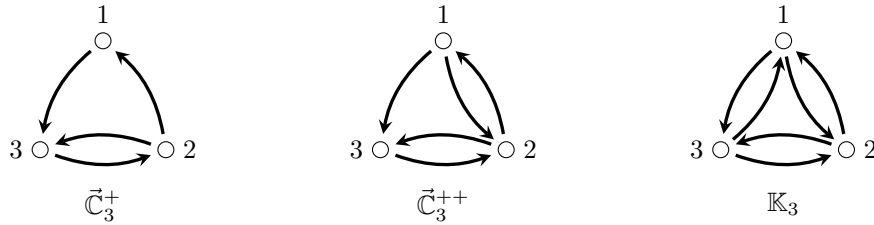
- $\text{CSP}(\mathbb{H})$ is NP-hard for $\vec{\mathbb{P}}_k$ -free digraphs whenever $k \geq N$, and
- $\text{CSP}(\mathbb{H})$ is NP-hard for $\vec{\mathbb{P}}_k$ -subgraph-free digraphs whenever $k \geq M$.

The following example shows this corollary fails in the infinite (ω -categorical) setting. A structure \mathbb{B} is ω -categorical if for every positive integer k the automorphism group of \mathbb{B} defines finitely many orbits of k -tuples.

► **Example 27.** Let \mathbb{H} be the disjoint union of \mathbb{K}_3 and the rational numbers with the strict linear order. It is straightforward to observe that 3-COLOURING reduces to $\text{CSP}(\mathbb{H})$, and that every acyclic digraph \mathbb{D} is a yes-instance of $\text{CSP}(\mathbb{H})$. Hence, $\text{CSP}(\mathbb{H})$ is NP-complete, but it is polynomial-time solvable on acyclic instances. The fact that \mathbb{H} is ω -categorical follows from its definition – it is the disjoint union of two ω -categorical digraphs.

7 Digraphs on three vertices

In this section we answer Question 1 for digraphs on three vertices. A loopless digraph on three vertices either contains two directed cycles and its CSP is NP-complete, or otherwise its CSP is polynomial-time solvable. Hence, we focus on the three loopless digraphs on three vertices: $\vec{\mathbb{C}}_3^+$ (obtained from the directed cycle by adding one edge), $\vec{\mathbb{C}}_3^{++}$ (obtained from the directed cycle by adding two edges), and \mathbb{K}_3 – see also Figure 4.



■ **Figure 4** The three digraphs on three vertices whose CSP is NP-complete.

Also note that these three digraphs are hereditarily hard (see, e.g., Conjecture 2.5 in [2] proved in [4]), and it thus follows from Theorem 5 that $\text{RCSP}(\mathbb{H}, \mathbb{T}_4)$ is NP-hard for $\mathbb{H} \in \{\vec{\mathbb{C}}_3^+, \vec{\mathbb{C}}_3^{++}, \mathbb{K}_3\}$. In turn, this implies that for such digraphs \mathbb{H} the problem $\text{CSP}(\mathbb{H})$ is NP-hard even restricted to $\vec{\mathbb{P}}_5$ -(subgraph)-free digraphs.

► **Remark 28.** If a $\vec{\mathbb{P}}_4$ -subgraph-free weakly connected digraph \mathbb{D} contains (as a subgraph) a directed 3-cycle v_1, v_2, v_3 , then $D = \{v_1, v_2, v_3\}$. Indeed, since \mathbb{D} is $\vec{\mathbb{P}}_4$ -subgraph-free and $(v_1, v_2), (v_2, v_3), (v_3, v_1) \in E$, it must be the case that the out- and in-neighbourhood of each v_i is a subset of $\{v_1, v_2, v_3\}$ (and so, the claim follows because \mathbb{D} is weakly connected).

We now show that $\text{CSP}(\vec{\mathbb{C}}_3^+)$ is tractable when the input is restricted to $\vec{\mathbb{P}}_4$ -subgraph-free digraphs. To do so, we reduce this problem to the list version of $\text{CSP}(\mathbb{G})$ where \mathbb{G} is the digraph depicted in Figure 5, and we briefly argue that this problem is polynomial-time solvable.

► **Lemma 29.** *The digraph \mathbb{G} from Figure 5 has a conservative majority polymorphism. In particular, $\text{CSP}(\mathbb{G}, U)$ can be solved in polynomial time.*

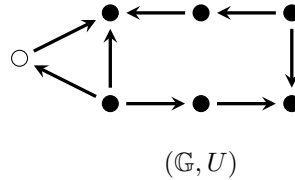
Proof. We define a conservative majority function $f: G^3 \rightarrow G$, and we then argue that it is a polymorphism. We denote by π_1 the projection onto the first coordinate, and by $\text{maj}(x, y, z)$ the majority operation when $|\{x, y, z\}| \leq 2$. Also, we simplify our writing by implicitly assuming that in the n -th itemized case of the following definition, neither of the first $n - 1$ cases holds.

$$f(x, y, z) = \begin{cases} \text{maj}(x, y, z) & \text{if } |\{x, y, z\}| \leq 2, \\ 1 & \text{if } 1 \in \{x, y, z\}, \text{ and } \{3, 7\} \cap \{x, y, z\} \neq \emptyset, \\ 2 & \text{if } 2 \in \{x, y, z\}, \text{ and } \{6, 7\} \cap \{x, y, z\} \neq \emptyset, \\ 3 & \text{if } 3 \in \{x, y, z\}, \text{ and } \{5, 7\} \cap \{x, y, z\} \neq \emptyset, \\ 4 & \text{if } 4, 6 \in \{x, y, z\}, \\ 6 & \text{if } \{x, y, z\} = \{5, 6, 7\}, \\ \pi_1(x, y, z) & \text{otherwise.} \end{cases}$$

It is clear that f is a symmetric conservative majority function. It is routine verifying that f is indeed a polymorphism. The fact that $\text{CSP}(\mathbb{G}, U)$ can be solved in polynomial time now follows from [12] because the same function f defines a majority polymorphism of (\mathbb{G}, U) . ◀

► **Lemma 30.** *$\text{CSP}(\vec{\mathbb{C}}_3^+)$ can be solved in polynomial-time when the input is restricted to $\bar{\mathbb{P}}_4$ -subgraph-free digraphs.*

Proof. Consider a loopless weakly connected digraph \mathbb{D} with no directed 3-cycles. Let B be the subset of vertices of \mathbb{D} incident to some symmetric edge, and let \mathbb{D}^* be the digraph obtained from \mathbb{D} after removing exactly one edge from each symmetric pair of edges. It is straightforward to observe there is no homomorphism from the directed path on four vertices to \mathbb{D}^* . Hence, by Observation 6, (\mathbb{D}^*, B) homomorphically maps to $(\mathbb{T}_3, \{1, 2, 3\})$ considered as a $\{E, U\}$ -structure where E is a binary predicate and U a unary predicate (this simply means that there is a homomorphism $\mathbb{D}^* \rightarrow \mathbb{T}_3$ such that every vertex in B is mapped to a vertex in $\{1, 2, 3\}$). Also note that $\mathbb{D} \rightarrow \vec{\mathbb{C}}_3^+$ if and only if there is a homomorphism $f: \mathbb{D}^* \rightarrow \vec{\mathbb{C}}_3^+$ where $f(B) \subseteq \{2, 3\}$, i.e., $\mathbb{D} \rightarrow \vec{\mathbb{C}}_3^+$ if and only if $(\mathbb{D}^*, B) \rightarrow (\vec{\mathbb{C}}_3^+, \{2, 3\})$ (as $\{E, U\}$ -structures). It follows from these arguments that $\text{CSP}(\vec{\mathbb{C}}_3^+)$ restricted to $\{\vec{\mathbb{C}}_3, \bar{\mathbb{P}}_4\}$ -subgraph-free digraphs reduces in polynomial time to $\text{CSP}((\vec{\mathbb{C}}_3^+, \{2, 3\}) \times (\mathbb{T}_3, \{1, 2, 3\}))$. It is not hard to observe that the core of $(\vec{\mathbb{C}}_3^+, \{2, 3\}) \times (\mathbb{T}_3, \{1, 2, 3\})$ is the structure (\mathbb{G}, U) depicted in Figure 5, whose CSP can be solved in polynomial time (Lemma 29).



■ **Figure 5** A $\{E, U\}$ -structure where E is a binary relation represented by arcs, and U is a unary relation represented by black vertices.

An algorithm that solves $\text{CSP}(\vec{\mathbb{C}}_3^+)$ in polynomial time when the input is restricted to $\bar{\mathbb{P}}_4$ -subgraph-free digraphs processes each weakly connected component of the input \mathbb{D} and accepts if and only if the following subroutine accepts each weakly connected component.

On a weakly connected component \mathbb{D}' of \mathbb{D} , the subroutine distinguishes whether \mathbb{D}' is $\vec{\mathbb{C}}_3$ -subgraph-free: if not, it accepts if \mathbb{D}' is either $\vec{\mathbb{C}}_3$ or $\vec{\mathbb{C}}_3^+$, and rejects otherwise – this step is consistent and correct because $|D'| = 3$ (see Remark 28); if \mathbb{D}' is $\vec{\mathbb{C}}_3$ -free, then the subroutine accepts or rejects \mathbb{D}' according to the reduction to $\text{CSP}(\mathbb{G}, U)$ explained above. Given the arguments in the previous paragraph, it is clear that this algorithm is consistent, correct, and runs in polynomial time. ◀

► **Theorem 31.** *The following statements hold for every positive integer k .*

- *If $k \leq 4$, then $\text{CSP}(\vec{\mathbb{C}}_3^+)$, $\text{CSP}(\vec{\mathbb{C}}_3^{++})$, and $\text{CSP}(\mathbb{K}_3)$ are solvable in polynomial-time when the input is restricted to $\vec{\mathbb{P}}_k$ -subgraph-free digraphs.*
- *If $k \geq 5$, then $\text{CSP}(\vec{\mathbb{C}}_3^+)$, $\text{CSP}(\vec{\mathbb{C}}_3^{++})$, and $\text{CSP}(\mathbb{K}_3)$ are NP-hard even if the input is restricted to $\vec{\mathbb{P}}_k$ -subgraph-free digraphs.*

Proof. The second item follows from the arguments above. The first item holds for \mathbb{K}_3 because every loopless $\vec{\mathbb{P}}_4$ -subgraph-free digraph is 3-colourable [19, Observation 38]. Similarly, every loopless $\{\mathbb{K}_3, \vec{\mathbb{P}}_4\}$ -subgraph-free digraph \mathbb{D} admits a homomorphism to $\vec{\mathbb{C}}_3^{++}$ [19, Lemma 39]. Hence, the first itemized statement holds for $\vec{\mathbb{C}}_3^{++}$. Finally, the tractability of $\text{CSP}(\vec{\mathbb{C}}_3^+)$ restricted to $\vec{\mathbb{P}}_4$ -subgraph-free digraphs follows by Lemma 30. ◀

Contrary to the $\vec{\mathbb{P}}_k$ -subgraph-free case, the CSPs of $\text{CSP}(\vec{\mathbb{C}}_3^+)$, $\text{CSP}(\vec{\mathbb{C}}_3^{++})$, and $\text{CSP}(\mathbb{K}_3)$ have different behaviours with respect to $\vec{\mathbb{P}}_k$ -free input restrictions. Due to space restrictions, we limit ourselves to present a proof idea, and refer the reader to Section 7.2 in the full version for a complete proof.

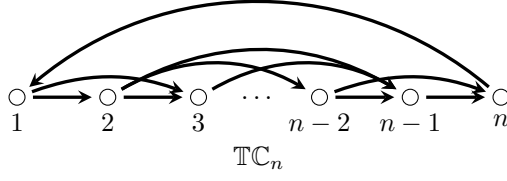
► **Theorem 32.** *The following statements hold for every positive integer $k \geq 2$.*

- *$\text{CSP}(\vec{\mathbb{C}}_3^+)$ is solvable in quadratic time for $\vec{\mathbb{P}}_k$ -free digraphs if $k \leq 3$, and NP-hard even for $\vec{\mathbb{P}}_k$ -free digraphs if $k \geq 4$,*
- *$\text{CSP}(\vec{\mathbb{C}}_3^{++})$ is solvable in linear time for $\vec{\mathbb{P}}_k$ -free digraphs if $k = 2$, and NP-hard even for $\vec{\mathbb{P}}_k$ -free digraphs if $k \geq 3$, and*
- *$\text{CSP}(\mathbb{K}_3)$ is NP-hard even for $\vec{\mathbb{P}}_k$ -free digraphs.*

Proof idea. The case of \mathbb{K}_3 is straightforward since $\text{CSP}(\mathbb{K}_3)$ is NP-complete for symmetric digraphs. The hard cases from the first and second item follow with standard gadget reductions; see [19, Lemma 43] for the former, and [19, Lemma 45] for the latter. Now, notice that $\text{CSP}(\vec{\mathbb{C}}_3^{++})$ restricted to $\vec{\mathbb{P}}_2$ -free digraphs reduces to $\text{CSP}(\mathbb{K}_2)$. Indeed, a symmetric digraph \mathbb{D} maps to $\vec{\mathbb{C}}_3^{++}$ if and only if \mathbb{D} is bipartite. Finally, the tractability of $\text{CSP}(\vec{\mathbb{C}}_3^+)$ restricted to $\vec{\mathbb{P}}_3$ -free digraphs is proved in [19, Lemma 44], where on a given $\vec{\mathbb{P}}_3$ -free digraph \mathbb{D} , the idea of the algorithm is the following. At each step there are three sets $X_1, X_2, X_3 \subseteq D$ such that the mapping $x \mapsto i$ if $x \in X_i$ defines a partial homomorphism from \mathbb{D} to $\vec{\mathbb{C}}_3^+$. We extend these sets to X'_1, X'_2, X'_3 in such a way that the partial homomorphism defined by X_1, X_2, X_3 extends to a homomorphism from $\mathbb{D} \rightarrow \vec{\mathbb{C}}_3^+$ if and only if the partial homomorphism defined by X'_1, X'_2, X'_3 extends to a homomorphism from $\mathbb{D} \rightarrow \vec{\mathbb{C}}_3^+$. ◀

8 A family of smooth tournaments

In this section we answer Question 1 for a natural family of smooth tournaments TC_n . Given a positive integer n , we denote by TC_n the tournament obtained from \mathbb{T}_n by reversing the edge from the source to the sink (see Figure 6). In particular, $\text{TC}_2 \cong \mathbb{T}_2$, and $\text{TC}_3 \cong \vec{\mathbb{C}}_3$, so $\text{CSP}(\text{TC}_n)$ is polynomial-time solvable for $n \leq 3$, and NP-complete for $n \geq 4$ (see, e.g., [1]).



■ **Figure 6** A depiction of the \mathbb{TC}_n – for a cleaner picture we omit the edges $(1, n-2)$, $(1, n-1)$, $(2, n)$, and $(3, n)$.

Since \mathbb{TC}_n is a hereditarily hard digraph ([2, Conjecture 2.5] proved in [4]) and $\mathbb{T}_n \not\rightarrow \mathbb{TC}_n$, it follows that $\text{RCSP}(\mathbb{TC}_n, \mathbb{T}_n)$ is NP-hard (Theorem 5). Equivalently, $\text{CSP}(\mathbb{TC}_n)$ is NP-hard for digraphs with no directed walk on $n+1$ vertices, and since $\mathbb{T}_{n-1} \rightarrow \mathbb{TC}_n$, $\text{CSP}(\mathbb{TC}_n)$ is polynomial-time solvable for digraphs with no directed walk on n vertices. However, if we only forbid $\vec{\mathbb{P}}_k$ as a subgraph (and not homomorphically), it turns out that $\text{CSP}(\mathbb{TC}_n)$ is NP-hard even for $\vec{\mathbb{P}}_5$ -subgraph-free digraphs. This hardness proof uses a similar gadget reduction as in the case of $\text{CSP}(\vec{\mathbb{C}}_3^+)$ [19, Lemma 47].

► **Theorem 33.** *For every pair of positive integers n and k the following statements hold.*

- *If $n \leq 3$ or $k \leq 4$, then $\text{CSP}(\mathbb{TC}_n)$ is in P for $\vec{\mathbb{P}}_k$ -subgraph-free digraphs.*
- *If $n \geq 4$ and $k \geq 5$, then $\text{CSP}(\mathbb{TC}_n)$ is NP-hard even for $\vec{\mathbb{P}}_k$ -subgraph-free digraphs.*

Proof. The hard cases follow from [19, Lemma 47]. The polynomial-time cases when $n \leq 3$ follow because $\mathbb{TC}_n \cong \mathbb{T}_n$ when $n \leq 3$. For, $n \geq 4$, the polynomial-time cases follow by Lemma 48 in [19] that $\text{CSP}(\mathbb{TC}_n)$ is in P for $\vec{\mathbb{P}}_4$ -subgraph-free digraphs because \mathbb{TC}_n is an oriented graph and contains \mathbb{T}_3 . ◀

The idea of deterministically extending a partial homomorphism to solve $\text{CSP}(\vec{\mathbb{C}}_3^+)$ on $\vec{\mathbb{P}}_3$ -free instances can also be used to see that $\text{CSP}(\mathbb{TC}_n)$ is polynomial-time solvable on $\vec{\mathbb{P}}_3$ -free instances. Moreover, in this case it also follows that there are finitely many minimal $\vec{\mathbb{P}}_3$ -free obstruction to $\text{CSP}(\mathbb{TC}_n)$, and with a little extra work, we can even spell out these minimal obstructions. To streamline this paper, we present these lists in the full version of this paper [19, Section 8.2]. For now, we answer Question 1.2 for \mathbb{TC}_n , where the hard cases are proved in Lemma 47 from [19], and the tractable ones are an immediate consequence of Theorem 55 in [19].

► **Theorem 34.** *For every positive integer k and $n \geq 4$ one of the following holds*

- *$k \leq 3$ and in this case $\text{CSP}(\mathbb{TC}_n)$ is tractable where the input is restricted to $\vec{\mathbb{P}}_k$ -free digraphs, or*
- *$k \geq 4$ and in this case $\text{CSP}(\mathbb{TC}_n)$ is NP-complete where the input is restricted to $\vec{\mathbb{P}}_k$ -free digraphs.*

9 Omitting a single digraph

As mentioned in the introduction, the long-term question of the research line introduced in this paper is the following.

► **Question 35.** *Is there a P versus NP-complete dichotomy of $\text{CSP}(\mathbb{H})$ where the input is restricted*

1. *to \mathbb{F} -free digraphs?*
2. *to \mathbb{F} -subgraph-free digraphs?*

Having settled Question 1 for the digraphs on three vertices and for the family of tournaments \mathbb{TC}_n , a natural next step is tackling Question 35 for these digraphs. In [19, Section 9], we discuss several results in this direction which follow from our previous work. We present the current landscape in Tables 1 and 2 in [19]. Here, we merge the main results of Section 9 in [19] (Theorem 57 and Corollary 60) in the following statement.

► **Theorem 36.** *For every positive integer n the problems $\text{CSP}(\vec{\mathbb{C}}_3^+)$ and $\text{CSP}(\mathbb{TC}_n)$ are NP-hard even when the input is restricted to \mathbb{F} -subgraph-free digraphs, whenever \mathbb{F} is not a disjoint union of oriented paths, or when \mathbb{F} is a connected digraph with $|F| \geq 12$.*

10 Conclusion and outlook

In this paper we have brought together homomorphisms, digraphs and \mathbb{H} -(subgraph-)free algorithmics. In doing so, we have uncovered a series of results concerning not only restricted CSPs, but also hardness of digraph CSPs under natural restrictions such as acyclicity. Our work raises numerous open problems, which we believe deserve attention in the future. Besides Questions 1 and 35, we ask the following.

- Is it true that for every finite structure \mathbb{A} and every (possibly infinite) \mathbb{B} , if (\mathbb{A}, \mathbb{B}) does not rpp-construct $(\mathbb{K}_3, \mathbb{L})$, then $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is polynomial-time solvable?
- Do finite-domain RCSPs (with possibly infinite restriction) exhibit a P vs. NP-hard dichotomy? (Assuming $P \neq NP$)
- Characterize the class of finite digraphs (structures) \mathbb{A} such that $\text{RCSP}(\mathbb{A}, \mathbb{B})$ is NP-hard for every finite \mathbb{B} such that $\mathbb{B} \not\rightarrow \mathbb{A}$ (assuming $P \neq NP$).
- Characterize the class of finite digraphs (structures) \mathbb{A} that *persistently construct* \mathbb{K}_3 , i.e., the template (\mathbb{A}, \mathbb{B}) rpp-constructs $(\mathbb{K}_3, \mathbb{L})$ whenever $\mathbb{B} \not\rightarrow \mathbb{A}$. See also [19, Observation 63].
- Characterize the class of finite digraphs (structures) \mathbb{H} such that $\text{RCSP}(\mathbb{H}, \mathbb{H}')$ is NP-hard whenever $\mathbb{H}' \not\rightarrow \mathbb{H}$ (assuming $P \neq NP$).
- Since $\text{CSP}(\mathbb{H})$ reduces to $\text{CSP}(\mathbb{H})$ on acyclic instances (Theorem 25), and the latter is equivalent to $\text{CSP}(\mathbb{H} \times \mathbb{Q})$ we ask: is it true that for every finite digraph \mathbb{H} the (infinite) digraph $\mathbb{Q} \times \mathbb{H}$ pp-constructs \mathbb{H} ?
- Is it true that for every tournament \mathbb{H} there a finitely many $\vec{\mathbb{P}}_3$ -free minimal obstructions to $\text{CSP}(\mathbb{H})$? (Compare to [19, Theorem 55]).

References

- 1 Jørgen Bang-Jensen, Pavol Hell, and Gary MacGillivray. The complexity of colouring by semicomplete digraphs. *SIAM Journal on Discrete Mathematics*, 1(3):281–298, 1988. doi:10.1137/0401029.
- 2 Jorgen Bang-Jensen, Pavol Hell, and Gary MacGillivray. Hereditarily hard H-colouring problems. *Discrete Mathematics*, 138:75–92, 1995. doi:10.1016/0012-365X(94)00189-P.
- 3 Libor Barto, Jakub Bulín, Andrei A. Krokhin, and Jakub Opršal. Algebraic approach to promise constraint satisfaction. *J. ACM*, 68(4):28:1–28:66, 2021. doi:10.1145/3457606.
- 4 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 Journal on Computing*, 38(5), 2009. doi:10.1137/070708093.
- 5 Libor Barto, Jakub Opršal, and Michael Pinsker. The wonderland of reflections. *Israel Journal of Mathematics*, 223(1):363–398, 2018. doi:10.1007/s11856-017-1621-9.
- 6 Manuel Bodirsky. *Complexity of Infinite-Domain Constraint Satisfaction*. Lecture Notes in Logic (52). Cambridge University Press, Cambridge, United Kingdom; New York, NY, 2021. doi:10.1017/9781107337534.

- 7 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.
- 8 Flavia Bonomo, Maria Chudnovsky, Peter Maceli, Oliver Schaudt, Maya Stein, and Mingxian Zhong. Three-Coloring and List Three-Coloring of Graphs Without Induced Paths on Seven Vertices. *Combinatorica*, 38:779–801, 2018. doi:10.1007/s00493-017-3553-8.
- 9 Richard C. Brewster and Timothy Graves. On the restricted homomorphism problem. *Discret. Appl. Math.*, 156(14):2823–2826, 2008. doi:10.1016/j.dam.2007.03.028.
- 10 Richard C. Brewster, Pavol Hell, and Gary MacGillivray. The complexity of restricted graph homomorphisms. *Discret. Math.*, 167-168:145–154, 1997. Selected Papers 15th British Combinatorial Conference. doi:10.1016/S0012-365X(96)00223-3.
- 11 Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs. In *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17*, pages 319–330, 2017. doi:10.1109/FOCS.2017.37.
- 12 Víctor Dalmau and Andrei A. Krokhin. Majority constraints have bounded pathwidth duality. *Eur. J. Comb.*, 29(4):821–837, 2008. doi:10.1016/j.ejc.2007.11.020.
- 13 P. Erdős. Graph theory and probability. *Canadian Journal of Mathematics*, 11:34–38, 1959.
- 14 Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: a study through Datalog and group theory. *SIAM Journal on Computing*, 28:57–104, 1999. doi:10.1137/S0097539794266766.
- 15 Jan Foniok. Homomorphisms and structural properties of relational systems. Doctoral Thesis, Charles University Prague, Faculty of Mathematics and Physics, 2007.
- 16 Jan Foniok, Jaroslav Nešetřil, and Claude Tardif. Generalised dualities and maximal finite antichains in the homomorphism order of relational structures. *European J. Combin.*, 29(4):881–899, 2008. doi:10.1016/j.ejc.2007.11.017.
- 17 Jan Goedgebeur, Jorik Jooken, Karolina Okrasa, Pawel Rzazewski, and Oliver Schaudt. Minimal obstructions to C_5 -coloring in hereditary graph classes. In Rastislav Kráľovic and Antonín Kucera, editors, *49th International Symposium on Mathematical Foundations of Computer Science, MFCS 2024, August 26-30, 2024, Bratislava, Slovakia*, volume 306 of *LIPIcs*, pages 55:1–55:15, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.MFCS.2024.55.
- 18 Petr A. Golovach, Matthew Johnson, Daniël Paulusma, and Jian Song. A survey on the computational complexity of coloring graphs with forbidden subgraphs. *J. Graph Theory*, 84(4):331–363, 2017. doi:10.1002/jgt.22028.
- 19 Santiago Guzmán-Pro and Barnaby Martin. Restricted CSPs and F-free Digraph Algorithmics. Preprint arXiv:2502.17596, 2025. doi:10.48550/arXiv.2502.17596.
- 20 Pavol Hell and Aurosh Mishra. H-coloring degree-bounded (acyclic) digraphs. *Theor. Comput. Sci.*, 554:40–49, 2014. doi:10.1016/j.tcs.2014.06.014.
- 21 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.
- 22 S. Huang. Improved complexity results on k-coloring. *European J. Combin.*, 51:336–346, 2016. doi:10.1016/j.ejc.2015.06.005.
- 23 Andrei Krokhin, Jakub Opršal, Marcin Wrochna, and Stanislav Živný. Topology and adjunction in promise constraint satisfaction. *SIAM J. Comput.*, 52(1):38–79, 2023. doi:10.1137/20m1378223.
- 24 Gábor Kun. Constraints, MMSNP, and expander relational structures. *Combinatorica*, 33(3):335–347, 2013. doi:10.1007/s00493-013-2405-4.
- 25 Jaroslav Nešetřil and Claude Tardif. Duality theorems for finite structures (characterising gaps and good characterisations). *J. Comb. Theory B*, 80(1):80–97, 2000. doi:10.1006/jctb.2000.1970.
- 26 Alan L. Selman. Promise problems complete for complexity classes. *Inf. Comput.*, 78(2):87–97, 1988. doi:10.1016/0890-5401(88)90030-2.

- 27 Dmitriy Zhuk. A proof of CSP dichotomy conjecture. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 331–342. IEEE Computer Society, 2017. <https://arxiv.org/abs/1704.01914>. doi:10.1109/FOCS.2017.38.
- 28 Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM*, 67(5):30:1–30:78, 2020. doi:10.1145/3402029.