Algebra and the Complexity of Digraph CSPs: a Survey

Benoît Larose

LACIM, Université du Québec à Montréal, Montréal, Canada
`blarose@lacim.ca`

—— **Abstract** ——————————————————————————————

We present a brief survey of some of the key results on the interplay between algebraic and graph-theoretic methods in the study of the complexity of digraph-based constraint satisfaction problems.

1 Introduction

At the expense of elegance and tradition, and since in all likelyhood the reader is already acquainted with some or all aspects of the topic at hand, we shall spare her the customary high-level introductory paragraphs, and refer to [10, 24, 75, 51] for more detailed accounts on motivation, background, and history of the field; in particular, we highly recommend the paper [60] for an excellent overview of algebraic methods in the study of digraph CSPs, as well as a wealth of interesting examples. In brief: the constraint satisfaction problem is a natural, flexible framework which encompasses several decision problems which are ubiquitous and fundamental in computer science; the introduction of powerful algebraic techniques in the pioneering work of [46], [61], [62] and [20] has led to a much deeper understanding of the algorithmic complexity of fixed-template CSPs; precise conjectures have been formulated linking the algorithmic and descriptive complexity of these CSPs to the algebraic properties of the fixed template. Very roughly, the paradigm underlying this theory is the following: if the template supports structure-preserving operations (*polymorphisms*) that obey "nice enough" identities, then the associated decision problem should be well-behaved from an algorithmic point of view; and if no such operations are present, then the problem is hard for some well-known complexity class. The present paper, concerned with the interplay of algebraic and graph-theoretic techniques in the study of these conjectures, focuses on CSPs whose fixed template is a digraph, possibly with some extra unary constraints. These are known in the literature under various names, such as *graph* or *digraph homomorphism problems*, *digraph list homomorphism problems*, *digraph with constants problem*, *digraph retraction problem*, *one-or-all list homomorphism problems*, and so on.

Why digraphs? Obviously these structures offer a good source of examples to test conjectures because they are simple, natural and we have powerful representation techniques for digraphs (in other words, we can draw them). One can sometimes hope to explicitly describe combinatorial properties that turn out to characterise the complexity. On the other hand, digraphs are flexible enough to encode complex problems. Secondly, various natural conditions can be imposed on digraphs to obtain subfamilies such as simple graphs, graphs

with loops, posets, tournaments, acyclic digraphs, etc. as to make testing some difficult conjectures more amenable; in this respect, the considerable literature on graph theory is a powerful tool. Thirdly, and perhaps more interestingly, the proofs of some strong general theorems on CSPs rely on results specifically on digraphs, see for instance [12]. It should also be noted that the complexity of CSPs on digraphs has been studied well before algebraic tools were introduced; in fact the concept of graph homomorphism can be traced back to the mid 20th century, and its complexity-theoretic aspects at least as far back as the late 70's.

To streamline the presentation, and in order to make this survey accessible to both mathematician and computer scientist alike, we shall avoid more involved universal algebraic concepts and results concerning varieties, tame congruence theory and such, and try to rephrase the relevant results only in terms of polymorphisms of structures whenever possible; for a more detailed account we refer the gentle reader to [10, 24]. The small price to pay for this approach is that some of the terminology we use is slightly non-standard (namely Definitions 6 and 7). Similarly, we avoid most technicalities concerning complexity issues and refer the reader to [2, 86] for these.

We make no claim at comprehensiveness, and obviously certain editorial choices have been made as to the inclusion or not of certain results. The literature on the complexity of digraph homomorphism is quite vast, so we shall focus mainly on those results involving algebraic techniques. Fixed-template constraint satisfaction problem involving digraphs that appear in the literature can be roughly classified into one of four different categories: (i) the "straightforward" $CSP(\mathbb{H})$ where $\mathbb{H}$ is a digraph; (ii) the so-called CSP with constants, or retraction problem, or one-or-all list homomorphism problem $CSP(\mathbb{H}^{+c})$ where $\mathbb{H}^{+c}$ is the structure consisting of the digraph $\mathbb{H}$ together with all unary singleton relations $\{h\}$ with $h \in H$; (iii) the list homomorphism or conservative constraint satisfaction problem $CSP(\mathbb{H}^{+l})$ where $\mathbb{H}^{+l}$ is the structure consisting of the digraph $\mathbb{H}$ together with all non-empty unary relations S with $S \subseteq H$; (iv) constraint satisfaction problems of one of the above forms but with various restrictions on the inputs (bounded degree, connected lists, bipartite inputs, etc.) Since this last case has not been much investigated with the use of algebraic tools we focus mainly on cases (i), (ii), (iii): in the sequel we shall refer to CSPs of one of these forms as *digraph CSPs*.

Some of the questions we wish to address in this paper are the following: are the dichotomy conjectures proved for a particular class of digraphs? Is there a combinatorial characterisation of the digraphs admitting such and such polymorphisms? Is there a combinatorial description of the digraphs whose CSP is solvable with such and such restriction on a given resource? Is there some collapse (of Mal'tsev conditions or complexity) for such and such class of digraphs?

Here is a brief outline of the paper: in section 2 we first introduce the basic notation and concepts in algebra, (descriptive) complexity and graphs that we require in the sequel. We shall then state in section 3 the important general results on CSPs we need later on, as well as the conjectures that will orient our presentation. In section 4 we present results on CSPs on digraphs and consider various subfamilies of digraphs; sections 5 and 6 deal with the variants of the CSP alluded to earlier, namely digraphs with constants and the list homomorphism problem. Section 7 closes the paper with a series of open problems.

2   Preliminaries

Where we set our notation and define our terms.

2.1 Relational Structures and Digraphs

A (finite) relational structure is a tuple $\mathbb{H} = \langle H; \theta_1, \cdots, \theta_r \rangle$ where H is a non-empty, finite set, and for each $1 \leq i \leq r$, θ_i is a relation of arity ρ_i on H, i.e. $\theta_i \subseteq H^{\rho_i}$; the *signature* of $\mathbb{H}$ is the sequence $(\rho_1, \ldots, \rho_r)$. In the sequel, all structures will be assumed finite, and equipped with finitely many basic relations. Let $\mathbb{H}_i = \langle H_i; \mu_1^i, \cdots, \mu_r^i \rangle$, $i = 1, 2$ be two structures of the same signature $\rho_1, \ldots, \rho_r$. The *product* of $\mathbb{H}_1$ and $\mathbb{H}_2$, is the relational structure $\mathbb{H}_1 \times \mathbb{H}_2 = \langle H_1 \times H_2; \mu_1, \cdots, \mu_r \rangle$ of the same signature as the $\mathbb{H}_i$ where for every $j = 1, \ldots, r$, $((x_1, y_1), \ldots, (x_{\rho_j}, y_{\rho_j})) \in \mu_j$ if $(x_1, \ldots, x_{\rho_j}) \in \mu_j^1$ and $(y_1, \ldots, y_{\rho_j}) \in \mu_j^2$. This extends naturally to any number of factors, and we use the notation $\mathbb{H}^k$ to denote the product of k copies of the structure $\mathbb{H}$.

Let $\mathbb{G} = \langle G; \mu_1, \cdots, \mu_r \rangle$ be a structure with the same signature as $\mathbb{H}$. A function $f : G \to H$ is a *homomorphism* from $\mathbb{G}$ to $\mathbb{H}$, and we write $f : \mathbb{G} \to \mathbb{H}$, if for every $1 \leq i \leq r$, $(f(x_1), \ldots, f(x_{\rho_i})) \in \theta_i$ whenever $(x_1, \ldots, x_{\rho_i}) \in \mu_i$. If there exist homomorphisms from $\mathbb{G}$ to $\mathbb{H}$ and from $\mathbb{H}$ to $\mathbb{G}$, we say that $\mathbb{G}$ and $\mathbb{H}$ are *homomorphically equivalent.* A structure $\mathbb{H}$ is a *core* if every homomorphism $f : \mathbb{H} \to \mathbb{H}$ is a bijection. It is well-known and easy to verify that every finite relational structure is homomorphically equivalent to a core which is unique up to isomorphism; hence we may speak of *the* core of a structure.

▶ **Definition 1.** Let $\mathbb{H}$ be a relational structure. The set of structures that admit a homomorphism to $\mathbb{H}$ is denoted by $CSP(\mathbb{H})$.

Viewed as a decision problem, $CSP(\mathbb{H})$ consists in determining on input $\mathbb{G}$ whether there exists a homomorphism from $\mathbb{G}$ to $\mathbb{H}$. Obviously if $\mathbb{H}'$ and $\mathbb{H}$ are homomorphically equivalent then $CSP(\mathbb{H}') = CSP(\mathbb{H})$; in particular, for every structure $\mathbb{H}$, we have that $CSP(\mathbb{H}) = CSP(\mathbb{H}')$ where $\mathbb{H}'$ is the core of $\mathbb{H}$.

▶ **Definition 2.** Let $\mathbb{H} = \langle H; \theta_1, \ldots, \theta_r \rangle$ be a structure.
- Let $\mathbb{H}^{+c} = \langle H; \theta_1, \ldots, \theta_r, \{h\} \, (h \in H) \rangle$ be the structure obtained from $\mathbb{H}$ by adding every singleton unary relation $\{h\} \, (h \in H)$ to $\mathbb{H}$ as a basic relation.
- Let $\mathbb{H}^{+l} = \langle H; \theta_1, \ldots, \theta_r, S \, (\emptyset \subset S \subseteq H) \rangle$ be the structure obtained from $\mathbb{H}$ by adding every non-empty subset $S \subseteq H$ to $\mathbb{H}$ as a basic relation.

Viewed as a decision problem, $CSP(\mathbb{H}^{+c})$ takes as input a structure $\mathbb{G}$ of the same signature as $\mathbb{H}$ where certain elements of $\mathbb{G}$ have been "pre-coloured" by some value in H; one must decide if there exists a homomorphism from $\mathbb{G}$ to $\mathbb{H}$ that respects the pre-colouring. This problem is known in the literature as the *homomorphism extension problem* [78], or *one-or-all list homomorphism problem* [40], and can easily seen to be equivalent to the so-called *retraction problem* [40]. On the other hand, $CSP(\mathbb{H}^{+l})$ takes as input a structure $\mathbb{G}$ of the same signature as $\mathbb{H}$ where certain elements of $\mathbb{G}$ are assigned a list of prescribed values in H; one must decide if a homomorphism f exists from $\mathbb{G}$ to $\mathbb{H}$ such that $f(x)$ belongs to the list assigned to x. This is known as the *list homomorphism problem*, and such problems are also known as *conservative CSPs*. Notice that the structures $\mathbb{H}^{+c}$ and $\mathbb{H}^{+l}$ are cores.

2.2 Digraphs

A *digraph* is a relational structure $\mathbb{H} = \langle H, \theta \rangle$ with a single binary relation θ; the members of H are the *vertices* of $\mathbb{H}$ and the elements of θ are called *arcs*. If (h, h') is an arc we say that h is an *in-neighbour* of h' and h' is an *out-neighbour* of h; we also say that h and h' are neighbours. The digraph $\mathbb{G} = \langle G; \rho \rangle$ is an *induced subdigraph* of $\mathbb{H}$ if $G \subseteq H$ and $\rho = \theta \cap G^2$. A digraph $\mathbb{H}$ is *connected (strongly connected)* if for every distinct $h, h' \in H$ there exists a

sequence $h = x_0, \ldots, x_n = h'$ of vertices of $\mathbb{H}$ such that x_i and x_{i+1} are neighbours $((x_i, x_{i+1})$ is an arc, respectively) for all $0 \le i \le n - 1$. A *connected component (strong connected component)* of $\mathbb{H}$ is a connected (strongly connected respectively) induced subdigraph of $\mathbb{H}$ maximal with respect to inclusion. A digraph $\mathbb{H} = \langle H; \theta \rangle$ is *bipartite* if $H = A \cup B$ with A and B disjoint such that $\theta \subseteq A \times B \cup B \times A$.

An arc of the form (h, h) is a *loop*; the digraph $\mathbb{H}$ is *reflexive* if θ contains all loops, and is *symmetric* if $(h, h') \in \theta$ implies $(h', h) \in \theta$; symmetric digraphs are often called *graphs*, and a *simple graph* is a graph without loops. The *underlying graph* of a digraph $\mathbb{H}$ is the graph obtained from $\mathbb{H}$ by replacing every arc by a symmetric edge. A digraph is *antisymmetric* if $(h, h'), (h', h) \in \theta$ implies $h = h'$, and it is *transitive* if $(h, h'') \in \theta$ whenever $(h, h'), (h', h'') \in \theta$. A *poset* is a reflexive, antisymmetric, transitive digraph.

An *oriented path* is a digraph with vertex set $\{x_0, \ldots, x_n\}$ $(n \ge 1)$ such that, for every $i = 0, \ldots, n - 1$, precisely one of (x_i, x_{i+1}) or (x_{i+1}, x_i) is an arc, and there are no other arcs; an *oriented cycle* is a digraph with vertex set $\{x_0, \ldots, x_n\}$ $(n \ge 1)$ such that, for every $i = 0, \ldots, n - 1$, precisely one of (x_i, x_{i+1}) or (x_{i+1}, x_i) is an arc, precisely one of (x_0, x_n) or (x_n, x_0) is an arc and there are no other arcs. The *net length* (or *algebraic length)* of an oriented cycle is the number of forward arcs minus the number of backward arcs according to some fixed traversal of the cycle. An oriented cycle is *balanced* if it has net length 0, and *unbalanced* otherwise. An n-vertex oriented cycle of net length n we call a *directed cycle (or circle)*. An *oriented tree* is an antisymmetric digraph whose underlying graph is a tree, i.e. an acyclic connected graph.

2.3 Polymorphisms

Let $\mathbb{H}$ be a relational structure. A *polymorphism* of $\mathbb{H}$ of *arity k* is a homomorphism from $\mathbb{H}^k$ to $\mathbb{H}$. If f is a polymorphism of $\mathbb{H}$ we also say that $\mathbb{H}$ *admits f*, or that $\mathbb{H}$ is *invariant under f*. A polymorphism is *idempotent* if it satisfies $f(x, x, \ldots, x) = x$ for all $x \in H$, and is *conservative* if $f(x_1, \ldots, x_n) \in \{x_1, \ldots, x_n\}$ for all $x_i \in H$.

We use the convenient notation $f(x_1, \ldots, x_k) \approx g(y_1, \ldots, y_n)$ to indicate equality where all variables are universally quantified, and call such an expression a *linear identity*.

A *semilattice* operation is an associative, idempotent, commutative binary operation. Let $k \ge 2$; a k-ary operation f is *cyclic* if it obeys

$$f(x_1, \ldots, x_k) \approx f(x_k, x_1, \ldots, x_{k-1});$$

it is *symmetric* if, for every permutation σ of the set $\{1, \ldots, k\}$, it obeys the identity

$$f(x_1, \ldots, x_k) \approx f(x_{\sigma(1)}, \ldots, x_{\sigma(k)});$$

and finally call f *totally symmetric (TS)* if

$$f(x_1, \ldots, x_k) \approx f(y_1, \ldots, y_k)$$

whenever $\{x_1, \ldots, x_k\} = \{y_1, \ldots, y_k\}$.

For $k \ge 3$, the operation f is a *near-unanimity (NU) operation* if it obeys the identity

$$f(x, \ldots, x, y, x, \ldots, x) \approx x$$

for any position of the lone y. A 3-ary NU operation is called a *majority* operation. For $k \ge 2$, the idempotent operation f is a *weak near-unanimity (WNU) operation* if it obeys the identities

$$f(x, \ldots, x, y, x, \ldots, x) \approx f(x, \ldots, x, y, x, \ldots, x)$$

for any positions of the lone y's.

A 3-ary operation f is *Mal'tsev* if it obeys the identities

$$f(y, y, x) \approx f(x, y, y) \approx x.$$

A 4-ary, idempotent operation f is *Siggers* if it satisfies the identity

$$f(a, r, e, a) \approx f(r, a, r, e).$$

We now gather some well-known implications involving the special polymorphisms defined here; as some of these results are folklore, we give a general reference only [24] (see also [66].)

▶ **Proposition 3.** *If a structure admits a (conservative) semilattice polymorphism then it admits (conservative) idempotent k-ary TS polymorphisms for all $k \geq 2$. A structure admits a Siggers polymorphism if and only if it admits a WNU polymorphism. If a structure admits an idempotent polymorphism f which is cyclic, symmetric, TS, NU or Mal'tsev then it admits a WNU polymorphism; moreover, in each case, if f is conservative, so is the WNU polymorphism.*

2.4 Datalog

Many naturally occurring tractable CSPs fall within one of two families of CSPs, namely problems of *bounded width* and those with *few subpowers*. The first family consists of problems solvable by local consistency methods; the CSPs in the second family are those that are solvable by an algorithm that shares many characteristics with Gaussian elimination; both classes of problems are characterised by identities [13], [18, 58]. As far as we can tell very little is known about digraph problems with few subpowers which do not have bounded width.

It is convenient for us to describe problems of bounded width with the use of the logic programming language Datalog; for more details see for instance [75]. A *Datalog program* is a finite set of rules of the form

$$T_0 :- T_1, \ldots, T_n$$

where each T_i is an atomic formula $R(x_{i_1}, \ldots, x_{i_k})$ from some fixed signature. Then T_0 is called the *head* of the rule, and the sequence $T_1, \ldots, T_n$ the *body* of the rule. There are two kinds of relational predicates occurring in the program: predicates R that occur at least once in the head of a rule are called *intensional database predicates* (IDBs) and are not part of τ. The other predicates which occur only in the body of a rule are called *extensional database predicates* and must all lie in τ. One special IDB, which is 0-ary, is the *goal predicate* of the program. Each Datalog program is a recursive specification of the IDBs, with semantics obtained via least fixed-points of monotone operators. The goal predicate is initially set to `false`, and the Datalog program *accepts* a structure $\mathbb{G}$ if its goal predicate evaluates to `true` on $\mathbb{G}$.

A Datalog program is *linear* if each of its rules has at most one occurrence of an IDB in its body. Given a rule t of the form

$$I :- J, T_1, \ldots, T_n$$

of a linear Datalog program where I and J are IDB's, its *symmetric* complement t_s is the rule

$$J :- I, T_1, \ldots, T_n;$$

if t has no IDB in the body then we let $t_s = t$. A linear program is said to be *symmetric* if it contains the symmetric complement of each of its rules. Finally, a Datalog program is *non-recursive* if the body of every rule contains only EDB's.

▶ **Definition 4.** A class $\mathcal{C}$ of structures is said to be *definable in (linear, symmetric, non-recursive) Datalog* if there is a (linear, symmetric, non-recursive) Datalog program which accepts precisely the structures from $\mathcal{C}$.

By their nature, Datalog programs define homomorphism closed classes of structures, hence in the context of CSPs a Datalog program accepts precisely the structures that *do not* map to the target structure; for instance it is a nice exercise to write up a symmetric Datalog program that recognises precisely non-bipartite graphs. To simplify the presentation we shall just say that $CSP(\mathbb{H})$ is definable in (linear, symmetric, non-recursive) Datalog rather than introduce extra notation. CSPs definable in Datalog are said to be of *bounded width*; CSPs definable in non-recursive Datalog are precisely those that are first-order definable; this was first proved in [3]; a slightly more refined result is Theorem 2 of [23]. $CSP(\mathbb{H})$ has *width 1* if it is recognised by a Datalog program whose IDBs are at most unary; a structure $\mathbb{H}$ has this property precisely if it admits a *set polymorphism*, or equivalently, if it admits TS polymorphisms of all arities [38, 33].

If $CSP(\mathbb{H})$ is definable in Datalog, then it is tractable; if it is definable in linear (symmetric) Datalog then it is solvable in non-deterministic (deterministic) logspace (see [75]). Combining results from [9], [83] and [66], the following characterises CSPs of bounded width.

▶ **Theorem 5.** *Let $\mathbb{H}$ be a core structure. Then the following are equivalent:*
1. $CSP(\mathbb{H})$ *has bounded width;*
2. *there is some N such that $\mathbb{H}$ admits k-ary WNU polymorphisms for all $k \geq N$;*
3. $\mathbb{H}$ *admits idempotent polymorphisms v and w satisfying*

$$v(x,x,y) \approx v(x,y,x) \approx v(y,x,x)$$
$$w(x,x,x,y) \approx w(x,x,y,x) \approx w(x,y,x,x) \approx w(y,x,x,x)$$
$$v(y,x,x) \approx w(y,x,x,x).$$

3 General Results

3.1 Three Conjectures and Some Results

It follows from deep results in universal algebra [57] that the existence of certain well-behaved polymorphisms on a structure $\mathbb{H}$ is equivalent to the impossibility of obtaining from $\mathbb{H}$ certain "minimal" structures via so-called *pp-interpretations*. It turns out that the CSPs associated to these minimal structures are logspace reducible to the original CSP [20, 75]; and hence the non-existence of the polymorphisms gives rise to natural hardness results, which are presented in Theorem 9 below. Conversely, it is believed (at least by some ...) that the presence of these polymorphisms should give complexity upper bounds (Conjecture 8). For completeness' sake we now define the polymorphisms in question.

▶ **Definition 6.** Let H be a structure, and $n \geq 2$. We say that $\mathbb{H}$ is *n-permutable* if there exist 3-ary polymorphisms $\{f_1, \ldots, f_{n-1}\}$ of $\mathbb{H}$ that satisfy for $i \leq n - 2$ the identities

$$x \approx f_1(x,y,y)$$
$$f_i(x,x,y) \approx f_{i+1}(x,y,y)$$
$$f_{n-1}(x,x,y) \approx y.$$

In particular a structure is 2-permutable precisely when it admits a Mal'tsev polymorphism.

Let t be a k-ary operation on the set H and let A be a $k \times k$ matrix with entries in H. We write $t[A]$ to denote the $k \times 1$ matrix whose entry on the i-th row is the value of f applied to row i of A.

▶ **Definition 7** ([66, 47])**.** Let H be a structure. We say that $\mathbb{H}$ is *join semidistributive* if there exists a k-ary idempotent polymorphism of $\mathbb{H}$ satisfying $t[A] = t[B]$ where A and B are $k \times k$ matrices with entries in $\{x, y\}$ such that $a_{ii} = x$ for all i, $a_{ij} = b_{ij} = x$ for all $i > j$ and $b_{ii} = y$ for all i.

▶ **Conjecture 8.** *Let $\mathbb{H}$ be a core structure.*
1. [20] *If $\mathbb{H}$ admits a WNU polymorphism then $CSP(\mathbb{H})$ is tractable.*
2. [75] *If $\mathbb{H}$ is join-semidistributive then $CSP(\mathbb{H})$ is definable in linear Datalog (and hence is solvable in non-deterministic logspace).*
3. [75] *If $CSP(\mathbb{H})$ has bounded width and $\mathbb{H}$ is n-permutable for some $n \geq 2$ then $CSP(\mathbb{H})$ is definable in symmetric Datalog (and hence is solvable in logspace).*

The first of these conjectures is known as the *algebraic dichotomy conjecture*; the "converse" of all three conjectures holds:

▶ **Theorem 9.** *Let $\mathbb{H}$ be a core structure.*
1. [20] *If $\mathbb{H}$ admits no WNU polymorphism then $CSP(\mathbb{H})$ is NP-complete.*
2. [75] *If $\mathbb{H}$ is not join-semidistributive then $CSP(\mathbb{H})$ is not expressible in linear Datalog and is P-hard.*
3. [75] *If $\mathbb{H}$ is not n-permutable for any n then $CSP(\mathbb{H})$ is not expressible in symmetric Datalog and is NL-hard.*

We gather in the next theorem some special cases of the conjectures that are known to hold.

▶ **Theorem 10.** *Let $\mathbb{H}$ be a core structure.*
1. [16] *If $\mathbb{H}$ admits an NU polymorphism then $CSP(\mathbb{H})$ is definable in linear Datalog;*
2. [32] *If $CSP(\mathbb{H})$ has bounded width and $\mathbb{H}$ is 2-permutable then $CSP(\mathbb{H})$ is definable in symmetric Datalog;*
3. [65] *If $CSP(\mathbb{H})$ is definable in linear Datalog and $\mathbb{H}$ is n-permutable for some $n \geq 2$ then $CSP(\mathbb{H})$ is definable in symmetric Datalog.*

Since join semidistributive structures automatically satisfy the equivalent conditions of Theorem 5 (see [66]), statement (3) in the previous result reduces the symmetric Datalog conjecture to the linear Datalog conjecture.

First-order definable CSPs are in a sense the "easiest" of all non-trivial CSPs. It is known that CSPs that are not first-order definable are logspace-hard [75], and hence there are no fixed template CSPs with complexity strictly between AC^0 and L. Furthermore, first-order definable CSPs cannot be characterised in a purely algebraic way in the sense of the above conjectures: indeed, adding the equality relation to a structure's basic relations does not change its polymorphisms but will make the CSP trivially logspace-hard. On the other hand, there exists a fairly simple combinatorial description of first-order definable CSPs via a dismantling algorithm [71], which in many special cases allows an explicit description of the underlying structures, see section 6 for some examples. It is known that the core of a structure $\mathbb{H}$ with first-order definable CSP admits an NU polymorphism [71], and by Theorem 9 it must also be k-permutable for some $k \geq 2$; furthermore, it follows from [85] that the CSP has tree duality, and hence the core of $\mathbb{H}$ must also admit idempotent TS polymorphisms of all arities.

3.2   Reductions to Digraph Problems

Digraph CSPs are, in full generality, as difficult as CSPs on more general templates. In fact, this remains true even for restricted families of digraphs. We say that two problems $\mathcal{A}$ and $\mathcal{B}$ are *poly-time (logspace, first-order) equivalent* if there exists polynomial time (logspace, first-order) reductions both from $\mathcal{A}$ to $\mathcal{B}$ and from $\mathcal{B}$ to $\mathcal{A}$.

▶ **Theorem 11.** *Let $\mathbb{H}$ be a relational structure.*
1. [46] *There exists a digraph $D(\mathbb{H})$ such that $CSP(D(\mathbb{H}))$ and $CSP(\mathbb{H})$ are poly-time equivalent.*
2. [46] *There exists a bipartite graph $B(\mathbb{H})$ such that $CSP(B(\mathbb{H})^{+c})$ and $CSP(\mathbb{H})$ are poly-time equivalent.*
3. [46] *There exists a poset $P(\mathbb{H})$ such that $CSP(P(\mathbb{H})^{+c})$ and $CSP(\mathbb{H})$ are poly-time equivalent.*
4. [40] *There exists a reflexive graph $R(\mathbb{H})$ such that $CSP(R(\mathbb{H})^{+c})$ and $CSP(\mathbb{H})$ are poly-time equivalent.*

Feder and Vardi [46] actually refine results (1)-(3) by imposing various stringent conditions on the digraphs, bipartite graphs and posets. Unfortunately, the reductions given do not seem to behave well with respect to polymorphisms. The next result handles this situation, and guarantees that all interesting polymorphism identities will be preserved, with the notable exception of Malt'sev polymorphisms; indeed, Kazda [63] has shown that every digraph that admits a Mal'tsev polymorphism also admits a majority polymorphism, a property which does not hold for structures in general.

Let $\mathbb{Z} = \langle Z; \theta \rangle$ be the digraph with $Z = \{0, 1, 2, 3\}$ and $\theta = \{(0,1), (2,1), (2,3)\}$. A linear identity $f(x_1, \ldots, x_k) \approx g(y_1, \ldots, y_n)$ is *balanced* if the variables appearing on each side are the same, i.e. $\{x_1, \ldots, x_k\} = \{y_1, \ldots, y_n\}$. Call a set Γ of linear identities *idempotent* if for each operation symbol f appearing in some identity of Γ the identity $f(x, \ldots, x) \approx x$ is in Γ. We say that a structure $\mathbb{H}$ *obeys* or *satisfies* Γ if for each operation symbol f appearing in Γ it admits a polymorphism $f_{\mathbb{H}}$ such that the set $\{f_{\mathbb{H}}\}$ satisfies the identities in Γ.

▶ **Theorem 12** ([26]). *Let $\mathbb{H}$ be a relational structure. There exists a digraph $\mathcal{D}(\mathbb{H})$ such that the following hold:*
1. *The problems $CSP(\mathcal{D}(\mathbb{H}))$ and $CSP(\mathbb{H})$ are logspace equivalent;*
2. *$\mathbb{H}$ is a core if and only if $\mathcal{D}(\mathbb{H})$ is a core;*
3. *If Γ is an idempotent set of linear identities such that*
   (a) *$\mathbb{Z}$ satisfies Γ,*
   (b) *every identity in Γ is either balanced or contains at most two variables,*
   then $\mathbb{H}$ satisfies Γ if and only if $\mathcal{D}(\mathbb{H})$ satisfies Γ.

We remark that condition (a) is not very restrictive since $\mathbb{Z}$ satisfies all interesting identities in the present context, with the exception of 2-permutability (but it is 3-permutable) [26].

4 $CSP(\mathbb{H})$

Obviously if the digraph $\mathbb{H}$ has a loop the problem $CSP(\mathbb{H})$ is trivial as any digraph $\mathbb{G}$ then admits a constant homomorphism to $\mathbb{H}$; consequently in this section all digraphs are assumed to have no loops. We begin with a classic result of Hell and Nešetřil's, reformulated in its stronger form that shows the algebraic dichotomy conjecture holds.

▶ **Theorem 13** ([50, 21]). *Let $\mathbb{H}$ be a symmetric digraph. If $\mathbb{H}$ is bipartite then $CSP(\mathbb{H})$ is tractable; otherwise $\mathbb{H}$ admits no WNU polymorphism and hence $CSP(\mathbb{H})$ is NP-complete.*

If $\mathbb{H}$ is a non-trivial bipartite graph then its core is an edge, and hence the problem $CSP(\mathbb{H})$ is in fact logspace-complete [1]. Other algebraic proofs of Hell and Nešetřil's result can be found in [12] and [93].

Arguably the simplest digraphs are oriented paths and cycles; the classification of the complexity of their associated CSPs was completed by Feder [39]. The case of balanced cycles is settled by a reduction to so-called bipartite boolean constraint-satisfaction problems that are shown to be either tractable or NP-complete, but the polymorphism behaviour is not quite transparent in the proof.

▶ **Theorem 14** ([56, 48, 39]). *Let $\mathbb{H}$ be a digraph.*

1. *If $\mathbb{H}$ is an oriented path, then it admits (conservative) majority and semilattice polymorphisms;*

2. *If $\mathbb{H}$ is an unbalanced oriented cycle, then it admits a (conservative) majority polymorphism;*

3. *If $\mathbb{H}$ is a balanced oriented cycle, then $CSP(\mathbb{H})$ is either tractable or NP-complete.*

There are known oriented trees with NP-complete CSP [48], [49]; the smallest known example is a 33-vertex triad [14]: a *polyad* is an oriented tree whose underlying graph has a unique vertex of degree greater than 2; a *triad* is a polyad with a unique vertex of degree 3. The algebraic dichotomy conjecture has been verified for a restricted family of triads called *special triads* [14], generalised to *special polyads* [11], and then to *special oriented trees* [25]. It turns out that the tractable CSPs on special oriented trees all have bounded width, and Bulín conjectures this holds for all oriented trees [25].

A *semi-complete* digraph is a digraph (without loops) such that there is at least one arc between every two vertices; this family includes complete graphs and tournaments as special cases. A dichotomy was first proved for semi-complete digraphs in [5]; the polymorphism behaviour of these digraphs is completely described in [60], Theorem 8.1. A digraph is *locally semi-complete* if if for every vertex v of $\mathbb{H}$, both the sets of in- (out-) neighbours of v induce semicomplete digraphs. A dichotomy for CSPs on connected locally semi-complete digraphs is proved in [6], Theorem 6.1; it turns out that the dividing line between tractability and NP-completeness is exactly the same for the list homomorphism problem on these digraphs ([6], Theorem 6.2).

A digraph is *smooth* if it has no sinks or sources, i.e. if every vertex has both in- and out-degree at least 1. The next result proves the algebraic dichotomy conjecture for CSPs on smooth digraphs, as well as confirming a conjecture of Bang-Jensen and Hell [4]:

▶ **Theorem 15** ([15]). *Let $\mathbb{H}$ be a smooth digraph. If each connected component of the core of $\mathbb{H}$ is a circle, then $CSP(\mathbb{H})$ is tractable, otherwise $\mathbb{H}$ admits no WNU polymorphism (and hence $CSP(\mathbb{H})$ is NP-complete).*

5 $CSP(\mathbb{H}^{+c})$

If we add to $\mathbb{H}$ all unary singleton relations as possible constraints, we obtain the problem $CSP(\mathbb{H}^{+c})$ which is in general harder than $CSP(\mathbb{H})$, unless $\mathbb{H}$ is itself a core, in which case the problems are logspace equivalent [62]. Notice that the polymorphisms of $\mathbb{H}^{+c}$ are precisely the *idempotent* polymorphisms of $\mathbb{H}$. For instance, if $\mathbb{H}$ is the symmetric 6-cycle, then its core is the symmetric edge and hence $CSP(\mathbb{H})$ is logspace-complete; on the other hand, $\mathbb{H}$ admits no idempotent polymorphisms other than projections (exercise), and hence by Theorem 9 above the problem $CSP(\mathbb{H}^{+c})$ is NP-complete. In the other direction, any complexity upper bound on the list homomorphism problem for $\mathbb{H}$ also applies to $CSP(\mathbb{H}^{+c})$;

notice also that the polymorphisms of $\mathbb{H}^{+l}$ are precisely the *conservative* polymorphisms of $\mathbb{H}$.

One of the interesting aspects of the decision problem $CSP(\mathbb{H}^{+c})$ from a combinatorial point of view is that, since $\mathbb{H}^{+c}$ is a core for any digraph $\mathbb{H}$, we may consider digraphs with possible loops. We start by examining a few results on mixed digraphs, then consider results on mixed undirected graphs, and then finally move on to reflexive digraphs. We shall use the following terminology: A *reflexive oriented path (cycle, tree)* is an oriented path (cycle, tree) where all loops have been added.

5.1 $CSP(\mathbb{H}^{+c})$ for Mixed Digraphs

An antisymmetric semi-complete digraph $\mathbb{H}$ is called a *tournament*, i.e. for every pair of distinct vertices u and v exactly one of $(u,v),(v,u)$ is an arc of $\mathbb{H}$. A *tournament of mixed type* is obtained from a tournament by adding some (perhaps not all) loops; more generally, we use the same terminology and talk about *mixed (di)graphs*, etc.

▶ **Theorem 16** ([92]). *Let $\mathbb{H}$ be a tournament of mixed type. Then either $CSP(\mathbb{H}^{+c})$ has bounded width or $\mathbb{H}$ admits no idempotent WNU polymorphism, and hence $CSP(\mathbb{H}^{+c})$ is NP-complete.*

A *strongly bipartite digraph* is a digraph $\mathbb{H} = \langle H; \theta \rangle$ where H is the disjoint union of two non-empty sets A and B and $\theta \subseteq A \times B$; equivalently, a digraph is strongly bipartite if every vertex is a source or a sink. The retraction problems for these digraphs exhibit a sharp collapse: indeed, by Theorem 10 the presence of an NU polymorphism guarantees that $CSP(\mathbb{H}^{+c})$ is definable in linear Datalog; in the present case we actually get all the way down to non-recursive Datalog:

▶ **Theorem 17** ([45]). *Let $\mathbb{H}$ be a connected strongly bipartite digraph. Then the following are equivalent:*
1. $\mathbb{H}$ *admits an NU polymorphism;*
2. $CSP(\mathbb{H}^{+c})$ *is first-order definable.*

5.2 $CSP(\mathbb{H}^{+c})$ for Mixed Undirected Graphs

A *(mixed) pseudotree* is a connected undirected graph that contains at most one cycle (other than loops, which are permitted.) The complexity of $CSP(\mathbb{H}^{+c})$ for pseudotrees is characterised in the next result, although the polymorphism behaviour of the tractable case is not transparent in the proof.

▶ **Theorem 18** ([43]). *Let $\mathbb{H}$ be a mixed pseudotree. If the loops of $\mathbb{H}$ induce a disconnected graph, or $\mathbb{H}$ contains an induced cycle of length at least 5, or a reflexive 4-cycle or an irreflexive 3-cycle, then $CSP(\mathbb{H}^{+c})$ is NP-complete. Otherwise $CSP(\mathbb{H}^{+c})$ is tractable.*

The special case of mixed undirected cycles is worth delineating. We note that the result invokes [90] which classifies the complexity of $CSP(\mathbb{H}^{+c})$ for all mixed undirected graphs with at most 4 vertices.

▶ **Theorem 19** ([43]). *Let $\mathbb{H}$ be a mixed undirected cycle on $n \geq 3$ vertices. If $n = 3$ and $\mathbb{H}$ has at least one loop, or if $n = 4$, $\mathbb{H}$ has at least one non-loop and the loops of $\mathbb{H}$ induce a connected graph, then $CSP(\mathbb{H}^{+c})$ is tractable. Otherwise, $CSP(\mathbb{H}^{+c})$ is NP-complete.*

Analogous to Theorem 17 above, retraction problems on bipartite graphs also exhibit some collapse, although not quite as sharp as in the strongly bipartite case. Notice that a non-trivial bipartite graph without loops cannot admit an idempotent binary TS polymorphism and hence its retraction problem cannot be first-order definable.

▶ **Theorem 20** ([70]). *Let $\mathbb{H}$ be a connected, irreflexive bipartite graph. If $\mathbb{H}$ admits an NU polymorphism then $CSP(\mathbb{H}^{+c})$ is definable in symmetric Datalog, and hence solvable in logspace.*

Combined with Theorem 9, it follows that a bipartite graph with an NU polymorphism must be k-permutable for some $k \geq 2$. R. Willard has verified that the converse holds if $k \leq 5$, however there exists a 6-permutable bipartite graph that admits no NU polymorphism [91].

5.3 $CSP(\mathbb{H}^{+c})$ for Reflexive Digraphs

A digraph is *intransitive* if, whenever $(u,v),(v,w),(u,w)$ are arcs then $|\{u,v,w\}| \leq 2$. Notice that any digraph of girth at least 4 (i.e. whose underlying graph contains no induced cycle of size 3 or less) is intransitive, in particular oriented trees as well as oriented cycles on 4 or more vertices are intransitive. [1]

▶ **Theorem 21** ([69]). *Let $\mathbb{H}$ be an intransitive reflexive digraph. Then the following are equivalent:*
1. *$\mathbb{H}$ admits a WNU polymorphism;*
2. *$\mathbb{H}$ admits a majority polymorphism;*
3. *$\mathbb{H}$ is a disjoint union of oriented trees;*
if any of these conditions hold, $CSP(\mathbb{H}^{+c})$ is definable in linear Datalog; otherwise $CSP(\mathbb{H}^{+c})$ is NP-complete.

The fact that reflexive oriented trees admit a majority polymorphism is from [42] (the majority operation defined in the undirected case in Corollary 2.58 of [51] respects orientations.)

Reflexive trees also admit a semilattice polymorphism [88]; hence the problem $CSP(\mathbb{H}^{+c})$ has width 1. We note in passing that there exist reflexive graphs whose retraction problem has width 1 but admit no semilattice polymorphism [55]; in fact, M. Siggers has recently found examples of such reflexive graphs that even admit a majority polymorphism [89]. A stronger statement than the last theorem holds for reflexive oriented cycles with at least 4 vertices, even allowing symmetric edges, which are in fact *idempotent trivial*, i.e. their only idempotent polymorphisms are projections [69]. The theorem as well as this last result are proved using a natural topological structure underlying reflexive digraphs; topological methods have also been used to study polymorphisms on digraphs in [30], [34], [74], [79].

Gumm terms characterise the important property of congruence modularity in varieties of algebras; by a result of Barto [7], a digraph $\mathbb{H}$ admits Gumm polymorphisms exactly when it admits edge or cube terms, i.e. when $CSP(\mathbb{H}^{+c})$ has few subpowers (and hence is tractable.) In general the existence of Gumm polymorphisms does not imply the existence of NU polymorphisms (although the converse is true); the next result shows that for reflexive digraphs these conditions are actually equivalent. This result generalises [84] and [77] which previously proved it for bounded posets and general posets respectively. The fact that width 1 is implied by the presence of an NU polymorphism was first proved for posets in [78].

---

[1] For completeness' sake we note briefly the behaviour of the remaining reflexive cycles: the directed cycle $\mathbb{H} = \langle\{0,1,2\};\theta\rangle$ with $\theta = \{(0,0),(1,1),(2,2),(0,1),(1,2),(2,0)\}$ is idempotent trivial, and all other 3-cycles admit either a majority or a semilattice polymorphism.

▶ **Theorem 22** ([81]). *Let $\mathbb{H}$ be a connected reflexive digraph. Then the following are equivalent:*
1. $\mathbb{H}$ *admits Gumm polymorphisms;*
2. $\mathbb{H}$ *admits an NU polymorphism.*
If these conditions hold, then $\mathbb{H}$ admits idempotent TS polymorphisms of all arities, and hence $CSP(\mathbb{H}^{+c})$ has width 1.

Combined with Theorem 42 of [27], it follows from the last statement that reflexive digraphs admitting a majority polymorphism are precisely those for which $CSP(\mathbb{H}^{+c})$ has so-called *path duality*.

Reflexive digraphs whose retraction problem is first-order definable have a nice description:

▶ **Theorem 23** ([73]). *Let $\mathbb{H}$ be a connected reflexive digraph. If $\mathbb{H}$ is k-permutable for some $k \geq 2$ then it is strongly connected. Consequently, the following are equivalent:*
1. $CSP(\mathbb{H}^{+c})$ *is first-order definable;*
2. $\mathbb{H}$ *is strongly connected and admits an NU polymorphism.*

The *sum* $\mathbb{G} \oplus \mathbb{H}$ of two reflexive digraphs $\mathbb{G}$ and $\mathbb{H}$ is the digraph obtained from the disjoint union of the digraphs, adding all arcs of the form (g, h) with $g \in G$ and $h \in H$. A reflexive digraph is *series-parallel* if it can be obtained from copies of the one element digraph using disjoint unions and sums. It is easy to see that such a digraph is in fact a poset. Equivalently, a poset is series-parallel if it is *N-free*, i.e. it does not contain the digraph $\mathbb{Z}$ (defined in section 3.2) as an induced subdigraph. The following result describes a dichotomy for series-parallel posets; the tractable posets are also characterised by a finite list of forbidden retracts, as well as a simple topological condition.

▶ **Theorem 24** ([30]). *Let $\mathbb{H}$ be a connected, series-parallel poset. Then the following are equivalent:*
1. $\mathbb{H}$ *admits a WNU polymorphism;*
2. $\mathbb{H}$ *admits idempotent TS polymorphisms of all arities $k \geq 2$.*
If these conditions hold, then $CSP(\mathbb{H}^{+c})$ has width 1, otherwise it is NP-complete.

There is a similar characterisation of series-parallel posets admitting an NU polymorphism [76]. For other work on the study of polymorphisms on posets the reader may consult the references of [84], [79] and [77]. There also has been extensive work on the study of polymorphisms on reflexive graphs, but most results relevant to this survey can be obtained as special cases of the above results of reflexive digraphs; for instance the analogs of Theorems 22 and 23 were first proved for reflexive graphs in [72] and [31]. [80] contains various interesting examples, [44] describes explicit generators for the variety of reflexive graphs, [19] studies NU polymorphisms on reflexive graphs, and [55, 87] investigate semilattice and lattice polymorphisms on these same graphs. [17] studies the idempotent polymorphisms of digraphs with at most 5 vertices.

6 $CSP(\mathbb{H}^{+l})$

Recall that the polymorphisms of the structure $\mathbb{H}^{+l}$ are precisely the conservative polymorphisms of $\mathbb{H}$. The proof of the algebraic dichotomy conjecture for the conservative case is due to Bulatov (see [8] for an alternative proof):

▶ **Theorem 25** ([22]). *Let $\mathbb{H}$ be a structure. If $\mathbb{H}$ admits a conservative WNU polymorphism then $CSP(\mathbb{H}^{+l})$ is tractable, otherwise it is NP-complete.*

It turns out that a stronger result holds for structures whose basic relations are at most binary:

▶ **Theorem 26** ([64]). *Let $\mathbb{H}$ be a structure whose basic relations are at most binary. If $\mathbb{H}$ admits a conservative WNU polymorphism then $CSP(\mathbb{H}^{+l})$ has bounded width.*

Hell and Rafiey had obtained this result earlier in the case of digraph CSPs (i.e. for a single binary relation) [52], as a by-product of a graph-theoretic description of the tractable cases, in terms of *digraph asteroidal triples (DAT)*; because the definition of a DAT is rather involved and technical we do not give it here. In a very recent paper [54], Hell and Rafiey have characterised digraphs admitting a conservative semilattice polymorphism; the following result is implicit in their proof, and shows that there is quite a bit of collapse for digraphs in the conservative case. Note that the equivalence of the last two conditions does not hold for general structures, see example 99 in [67].

▶ **Theorem 27** ([54]). *Let $\mathbb{H}$ be a digraph. Then the following are equivalent:*
1. *$\mathbb{H}$ admits a conservative semilattice polymorphism;*
2. *$\mathbb{H}$ admits conservative cyclic polymorphisms of all arities;*
3. *$\mathbb{H}$ admits conservative symmetric polymorphisms of all arities;*
4. *$\mathbb{H}$ admits conservative TS polymorphisms of all arities, i.e. $CSP(\mathbb{H}^{+l})$ has width 1.*

The logspace conjecture (Conjecture 8 (3)) has been verified for at most binary structures [29]; here we state the graph-theoretic description of the digraphs with $CSP(\mathbb{H}^{+l})$ definable in symmetric Datalog which is from [36].

Let $\mathbb{H}$ be a digraph, and let $x, y \in H$. We say that (x, y) is an *edge* if either (x, y) or (y, x) is an arc of $\mathbb{H}$. A sequence of vertices $x_0, \dots, x_n$, $(n \geq 0)$ in $\mathbb{H}$ such that (x_i, x_{i+1}) is an edge for all $0 \leq i \leq n - 1$ is called a *walk* in $\mathbb{H}$ from x_0 to x_n; we call the pair (x_i, x_{i+1}) a *forward (backward) edge* if (x_i, x_{i+1}) $((x_{i+1}, x_i)$ respectively) is an arc. Two walks $P = x_0, \dots, x_n$ and $Q = y_0, \dots, y_n$ in $\mathbb{H}$ are *congruent*, if they follow the same pattern of forward and backward edges, i.e., when (x_i, x_{i+1}) is an arc if and only if (y_i, y_{i+1}) is an arc. Suppose P, Q and $R = z_0, \dots, z_n$ are pairwise congruent walks. We say that (x_i, y_{i+1}) is a *faithful edge from P to Q* if it is an edge of $\mathbb{H}$ in the same direction (forward or backward) as (x_i, x_{i+1}). We say that *P avoids Q* in $\mathbb{H}$ if there is no faithful edge from P to Q; *R protects Q from P* if the existence of faithful edges (x_i, z_{i+1}) and (z_j, y_{j+1}) implies that $j \leq i$. The digraph $\mathbb{H}$ *contains a circular N* if there exist vertices $x, y \in H$, congruent walks P from x to x, Q from y to y and R from y to x such that P avoids Q and R protects Q from P.

▶ **Theorem 28** ([29, 36]). *Let $\mathbb{H}$ be a digraph. Then the following are equivalent:*
1. *$\mathbb{H}$ does not contain a circular N;*
2. *$\mathbb{H}$ is k-permutable for some $k \geq 2$;*
3. *$CSP(\mathbb{H}^{+l})$ is definable in symmetric Datalog.*
If one of these conditions holds then $CSP(\mathbb{H}^{+l})$ is solvable in logspace, otherwise it is NL hard.

[35] contains related results on oriented trees; digraphs that admit a conservative semilattice polymorphism are characterised in [53]. The digraphs with first-order definable list homomorphism problem also admit a nice graph-theoretic description [59]: two arcs (x_1, y_1) and (x_2, y_2) of a digraph $\mathbb{H}$ are said to be *separated* if neither (x_1, y_2) nor (x_2, y_1) is an arc of $\mathbb{H}$. A *hindering bicycle in* $\mathbb{H}$ is a subset $\{x_0, \dots, x_n, y_0, \dots, y_n\}$ of vertices of $\mathbb{H}$ $(n \geq 0)$ such that (i) (x_i, x_{i+1}), (y_i, y_{i+1}) and (x_i, y_{i+1}) are arcs of $\mathbb{H}$ for all $i = 0, \dots, n$ (indices modulo $n + 1$) and (ii) (x_{i+1}, y_i) is not an arc of $\mathbb{H}$ for any $i = 0, \dots, n$ (indices modulo $n + 1$).

▶ **Theorem 29** ([59])**.** *Let $\mathbb{H}$ be a digraph. Then the following are equivalent:*
1. $\mathbb{H}$ *contains no separated arcs nor any hindering bicycle;*
2. $CSP(\mathbb{H}^{+l})$ *is first-order definable.*

The special case of graphs (with loops allowed) in interesting in its own right. The algebraic dichotomy conjecture for list homomorphism problems has a very neat dividing line in this context: the graphs such that $CSP(\mathbb{H}^{+l})$ is tractable are the so-called *bi-arc graphs* [41], which are precisely the graphs that admit a conservative majority polymorphism [19]. In [70] it is shown that among these, the graphs whose list homomorphism problem has width 1 are the bi-arc graphs that do not have a loopless edge; equivalently, these are the graphs that admit a *binary* conservative WNU polymorphism.

Since the presence of a majority polymorphism guarantees the CSP is definable in linear Datalog, Conjecture 8 (2) for the list homomorphism problem on graphs follows from the above; the proof of Conjecture 8 (3) in this special case can be found in [37]; an explicit description by finitely many forbidden subgraphs is given for the graphs $\mathbb{H}$ such that $CSP(\mathbb{H}^{+l})$ is definable in symmetric Datalog.

7   Open Problems and Further Discussion

We list, in no particular order, some open questions and problems, as well as further discussion of the results presented earlier.

1. If $\mathbb{H}$ is an oriented tree such that $CSP(\mathbb{H})$ is tractable, does it also have bounded width [25]?
2. There is very little known about digraphs admitting (conservative) cube or edge terms, i.e. such that the problems $CSP(\mathbb{H})$, $CSP(\mathbb{H}^{+c})$ and $CSP(\mathbb{H}^{+l})$ have few subpowers. Investigate.
3. Characterise those digraphs $\mathbb{H}$ whose list homomorphism problem is definable in linear Datalog and confirm Conjecture 8 (2) in this case.
4. Give a (simple?) graph-theoretic characterisation of digraphs that admit a conservative NU polymorphism.
5. Which posets admit a semilattice polymorphism? Does there exist a poset that admits TS polymorphisms of all arities, or even an NU polymorphism, but no semilattice polymorphism?
6. There exist posets whose retraction problem is tractable but does not have bounded width [68] [2], but the ones that are known are quite large. Find small examples of such posets. Same question for reflexive graphs.
7. There exists an acyclic digraph $\mathbb{H}$ such that $CSP(\mathbb{H})$ is tractable but does not have bounded width [3] but it is quite large; find some amenable examples.
8. M. Maróti [82] has analysed small reflexive digraphs by computer and obtained several 6-element examples whose retraction problem has bounded width but not width 1; there are no such examples for posets nor reflexive graphs of size at most 8. Investigate.
9. Topological methods would appear promising in the analysis of polymorphisms on reflexive digraphs, but there has been only preliminary work in this direction. For instance, is there a characterisation of reflexive digraphs admitting a WNU polymorphism via homotopy groups of idempotent subalgebras analogous to the case of posets? See the remarks after Corollary 4.5 in [77], but see also Proposition 1.3 of [72].

---

[2] L. Barto has verified that the example there is indeed tractable without the assumption that $P \neq NP$.

10. The complexity of deciding if a relational structure admits such and such "nice" polymorphism has been investigated in [28]. For many identities, the hardness results for general structures are still valid for structures with at most binary basic relations; however, for a single binary relation, i.e. a digraph, the problem often turns out to be better behaved. Investigate.

References

1 Eric Allender, Michael Bauland, Neil Immerman, Henning Schnoor, and Heribert Vollmer. The complexity of satisfiability problems: refining Schaefer's theorem. In *Mathematical foundations of computer science 2005*, volume 3618 of *Lecture Notes in Comput. Sci.*, pages 71–82. Springer, Berlin, 2005. `doi:10.1007/11549345_8`.

2 Sanjeev Arora and Boaz Barak. *Computational complexity.* Cambridge University Press, Cambridge, 2009. A modern approach. `doi:10.1017/CBO9780511804090`.

3 Albert Atserias. On digraph coloring problems and treewidth duality. *European J. Combin.*, 29(4):796–820, 2008. `doi:10.1016/j.ejc.2007.11.004`.

4 Jørgen Bang-Jensen and Pavol Hell. The effect of two cycles on the complexity of colourings by directed graphs. *Discrete Appl. Math.*, 26(1):1–23, 1990. `doi:10.1016/0166-218X(90)90017-7`.

5 Jørgen Bang-Jensen, Pavol Hell, and Gary MacGillivray. The complexity of colouring by semicomplete digraphs. *SIAM J. Discrete Math.*, 1(3):281–298, 1988. `doi:10.1137/0401029`.

6 Jørgen Bang-Jensen, Gary MacGillivray, and Jacobus Swarts. The complexity of colouring by locally semicomplete digraphs. *Discrete Math.*, 310(20):2675–2684, 2010. `doi:10.1016/j.disc.2010.03.033`.

7 L. Barto. Finitely related algebras in congruence modular varieties have few subpowers. to appear in JEMS.

8 Libor Barto. The dichotomy for conservative constraint satisfaction problems revisited. In *26th Annual IEEE Symposium on Logic in Computer Science – LICS 2011*, pages 301–310. IEEE Computer Soc., Los Alamitos, CA, 2011.

9 Libor Barto. The collapse of the bounded width hierarchy. *Journal of Logic and Computation*, 2014. `doi:10.1093/logcom/exu070`.

10 Libor Barto. The constraint satisfaction problem and universal algebra. *Bull. Symb. Log.*, 21(3):319–337, 2015. `doi:10.1017/bsl.2015.25`.

11 Libor Barto and Jakub Bulín. CSP dichotomy for special polyads. *Internat. J. Algebra Comput.*, 23(5):1151–1174, 2013. `doi:10.1142/S0218196713500215`.

12 Libor Barto and Marcin Kozik. Absorbing subalgebras, cyclic terms, and the constraint satisfaction problem. *Log. Methods Comput. Sci.*, 8(1):1:07, 27, 2012. `doi:10.2168/LMCS-8(1:7)2012`.

13 Libor Barto and Marcin Kozik. Constraint satisfaction problems solvable by local consistency methods. *J. ACM*, 61(1):3, 2014.

14 Libor Barto, Marcin Kozik, Miklós Maróti, and Todd Niven. CSP dichotomy for special triads. *Proc. Amer. Math. Soc.*, 137(9):2921–2934, 2009. `doi:10.1090/S0002-9939-09-09883-9`.

15 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, 2008/09. `doi:10.1137/070708093`.

16   Libor Barto, Marcin Kozik, and Ross Willard. Near unanimity constraints have bounded pathwidth duality. In *Proceedings of the 2012 27th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 125–134. IEEE Computer Soc., Los Alamitos, CA, 2012. `doi:10.1109/LICS.2012.24`.

17   Libor Barto and David Stanovský. Polymorphisms of small digraphs. *Novi Sad J. Math.*, 40(2):95–109, 2010.

18   J. Berman, P. Idziak, P. Markovic, R. McKenzie, M. Valeriote, and R. Willard. Varieties with few subalgebras of powers. *Transactions of the American Mathematical Society*, 362(3):1445–1473, 2010.

19   Richard C. Brewster, Tomas Feder, Pavol Hell, Jing Huang, and Gary MacGillivray. Near-unanimity functions and varieties of reflexive graphs. *SIAM J. Discrete Math.*, 22(3):938–960, 2008. `doi:10.1137/S0895480103436748`.

20   A. Bulatov, P. Jeavons, and A. Krokhin. Classifying the Complexity of Constraints using Finite Algebras. *SIAM Journal on Computing*, 34(3):720–742, 2005.

21   Andrei A. Bulatov. *H*-coloring dichotomy revisited. *Theoret. Comput. Sci.*, 349(1):31–39, 2005. `doi:10.1016/j.tcs.2005.09.028`.

22   Andrei A. Bulatov. Complexity of conservative constraint satisfaction problems. *ACM Trans. Comput. Log.*, 12(4):24, 2011.

23   Andrei A. Bulatov, Andrei A. Krokhin, and Benoit Larose. Dualities for constraint satisfaction problems. In *Complexity of Constraints – An Overview of Current Research Themes (Result of a Dagstuhl Seminar)*, volume 5250 of *Lecture Notes in Computer Science*, pages 93–124. Springer, 2008. `doi:10.1007/978-3-540-92800-3_5`.

24   Andrei A. Bulatov and Matthew Valeriote. Recent results on the algebraic approach to the CSP. In *Complexity of Constraints – An Overview of Current Research Themes (Result of a Dagstuhl Seminar)*, volume 5250 of *Lecture Notes in Computer Science*, pages 68–92. Springer, 2008. `doi:10.1007/978-3-540-92800-3_4`.

25   Jakub Bulín. On the complexity of h-coloring for special oriented trees. arXiv:1407.1779v2, 2014.

26   Jakub Bulín, Dejan Delić, Marcel Jackson, and Todd Niven. A finer reduction of constraint problems to digraphs. *Log. Methods Comput. Sci.*, 11(4):4:18, 33, 2015.

27   Catarina Carvalho, Víctor Dalmau, and Andrei Krokhin. Two new homomorphism dualities and lattice operations. *J. Logic Comput.*, 21(6):1065–1092, 2011. `doi:10.1093/logcom/exq030`.

28   Hubie Chen and Benoit Larose. Asking the metaquestions in constraint tractability. arXiv:1604.00932, 2016.

29   Víctor Dalmau, László Egri, Pavol Hell, Benoit Larose, and Arash Rafiey. Descriptive complexity of list h-coloring problems in logspace: A refined dichotomy. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2015, Kyoto, Japan, July 6-10, 2015*, pages 487–498, 2015. `doi:10.1109/LICS.2015.52`.

30   Víctor Dalmau, Andrei Krokhin, and Benoit Larose. Retractions onto series-parallel posets. *Discrete Math.*, 308(11):2104–2114, 2008. `doi:10.1016/j.disc.2006.08.010`.

31   Víctor Dalmau, Andrei A. Krokhin, and Benoit Larose. First-order definable retraction problems for posets and reflexive graph. In *19th IEEE Symposium on Logic in Computer Science (LICS 2004), 14-17 July 2004, Turku, Finland, Proceedings*, pages 232–241, 2004. `doi:10.1109/LICS.2004.1319617`.

32   Víctor Dalmau and Benoit Larose. Maltsev + datalog –> symmetric datalog. In *Proceedings of the Twenty-Third Annual IEEE Symposium on Logic in Computer Science, LICS 2008, 24-27 June 2008, Pittsburgh, PA, USA*, pages 297–306, 2008. `doi:10.1109/LICS.2008.14`.

33   Victor Dalmau and Justin Pearson. Closure Functions and Width 1 Problems. In *CP 1999*, pages 159–173, 1999.

34 C. Delhommé. Projection properties and reflexive binary relations. *Algebra Universalis*, 41(4):255–281, 1999. `doi:10.1007/s000120050115`.

35 Lászlo Egri. Space complexity of list h-coloring revisited: the case of oriented trees. arXiv:1510.07124, 2015.

36 László Egri, Pavol Hell, Benoit Larose, and Arash Rafiey. Space complexity of list *H*-colouring: a dichotomy. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, Oregon, USA, January 5-7, 2014*, pages 349–365, 2014. `doi:10.1137/1.9781611973402.26`.

37 László Egri, Andrei Krokhin, Benoit Larose, and Pascal Tesson. The complexity of the list homomorphism problem for graphs. *Theory Comput. Syst.*, 51(2):143–178, 2012. `doi:10.1007/s00224-011-9333-8`.

38 T. Feder and M. 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.

39 Tomás Feder. Classification of homomorphisms to oriented cycles and of *k*-partite satisfiability. *SIAM J. Discrete Math.*, 14(4):471–480 (electronic), 2001. `doi:10.1137/S0895480199383353`.

40 Tomas Feder and Pavol Hell. List homomorphisms to reflexive graphs. *J. Combin. Theory Ser. B*, 72(2):236–250, 1998. `doi:10.1006/jctb.1997.1812`.

41 Tomas Feder, Pavol Hell, and Jing Huang. Bi-arc graphs and the complexity of list homomorphisms. *J. Graph Theory*, 42(1):61–80, 2003. `doi:10.1002/jgt.10073`.

42 Tomas Feder, Pavol Hell, and Jing Huang. List homomorphisms and retractions to reflexive digraphs. Manuscript, 2007.

43 Tomás Feder, Pavol Hell, Peter Jonsson, Andrei Krokhin, and Gustav Nordh. Retractions to pseudoforests. *SIAM J. Discrete Math.*, 24(1):101–112, 2010. `doi:10.1137/080738866`.

44 Tomás Feder, Pavol Hell, Benoît Larose, Cynthia Loten, Mark Siggers, and Claude Tardif. Graphs admitting *k*-NU operations. Part 1: The reflexive case. *SIAM J. Discrete Math.*, 27(4):1940–1963, 2013. `doi:10.1137/120894312`.

45 Tomás Feder, Pavol Hell, Benoît Larose, Mark Siggers, and Claude Tardif. Graphs admitting *k*-NU operations. Part 2: The irreflexive case. *SIAM J. Discrete Math.*, 28(2):817–834, 2014. `doi:10.1137/130914784`.

46 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 (electronic), 1999. `doi:10.1137/S0097539794266766`.

47 Ralph Freese and Ralph McKenzie. Maltsev families of varieties closed under join or maltsev product. Preprint, 2015.

48 Wolfgang Gutjahr, Emo Welzl, and Gerhard Woeginger. Polynomial graph-colorings. *Discrete Appl. Math.*, 35(1):29–45, 1992. `doi:10.1016/0166-218X(92)90294-K`.

49 P. Hell, J. Nešetřil, and X. Zhu. Complexity of tree homomorphisms. *Discrete Appl. Math.*, 70(1):23–36, 1996. `doi:10.1016/0166-218X(96)00099-6`.

50 Pavol Hell and Jaroslav Nešetřil. On the complexity of *H*-coloring. *J. Combin. Theory Ser. B*, 48(1):92–110, 1990. `doi:10.1016/0095-8956(90)90132-J`.

51 Pavol Hell and Jaroslav Nešetřil. *Graphs and homomorphisms*, volume 28 of *Oxford Lecture Series in Mathematics and its Applications*. Oxford University Press, Oxford, 2004. `doi:10.1093/acprof:oso/9780198528173.001.0001`.

52 Pavol Hell and Arash Rafiey. The dichotomy of list homomorphisms for digraphs. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1703–1713. SIAM, Philadelphia, PA, 2011.

53 Pavol Hell and Arash Rafiey. Monotone proper interval digraphs and min-max orderings. *SIAM J. Discrete Math.*, 26(4):1576–1596, 2012. `doi:10.1137/100783844`.

54 Pavol Hell and Arash Rafiey. Bi-arc digraphs and conservative polymorphisms. arXiv:1608.03368, 2016.

55 Pavol Hell and Mark Siggers. Semilattice polymorphisms and chordal graphs. *European J. Combin.*, 36:694–706, 2014. `doi:10.1016/j.ejc.2013.10.007`.

56 Pavol Hell and Xu Ding Zhu. The existence of homomorphisms to oriented cycles. *SIAM J. Discrete Math.*, 8(2):208–222, 1995. `doi:10.1137/S0895480192239992`.

57 D. Hobby and R. McKenzie. *The Structure of Finite Algebras*, volume 76 of *Contemporary Mathematics*. American Mathematical Society, 1988.

58 P. Idziak, P. Markovic, R. McKenzie, M. Valeriote, and R. Willard. Tractability and learnability arising from algebras with few subpowers. *SIAM J. Comput.*, 39(7):3023–3037, 2010.

59 Adrien Lemaître. *Complexité des homomorphismes de graphes avec listes.* PhD thesis, Université de Montréal (Canada), 2012.

60 Marcel Jackson, Tomasz Kowalski, and Todd Niven. Complexity and polymorphisms for digraph constraint problems under some basic constructions. to appear in Internat. J. Algebra Comput. (arXiv:1304.4986), 2016.

61 P. Jeavons, D. Cohen, and M. Cooper. Constraints, consistency, and closure. *Artificial Intelligence*, 101(1-2):251–265, 1998.

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

63 Alexandr Kazda. Maltsev digraphs have a majority polymorphism. *European Journal of Combinatorics*, 32(3):390–397, 2011.

64 Alexandr Kazda. CSP for binary conservative relational structures. *Algebra Universalis*, 75(1):75–84, 2016. `doi:10.1007/s00012-015-0358-8`.

65 Alexandr Kazda. n-permutability and linear datalog implies symmetric datalog. arXiv:1508.05766v1, 2016.

66 Marcin Kozik, Andrei Krokhin, Matt Valeriote, and Ross Willard. Characterizations of several Maltsev conditions. *Algebra Universalis*, 73(3-4):205–224, 2015. `doi:10.1007/s00012-015-0327-2`.

67 Gábor Kun and Mario Szegedy. A new line of attack on the dichotomy conjecture. *European J. Combin.*, 52(part B):338–367, 2016. `doi:10.1016/j.ejc.2015.07.011`.

68 B. Larose and L. Zádori. Bounded width problems and algebras. *Algebra Universalis*, 56(3-4):439–466, 2007.

69 Benoit Larose. Taylor operations on finite reflexive structures. *Int. J. Math. Comput. Sci.*, 1(1):1–21, 2006.

70 Benoît Larose and Adrien Lemaître. List-homomorphism problems on graphs and arc consistency. *Discrete Math.*, 313(22):2525–2537, 2013. `doi:10.1016/j.disc.2013.07.018`.

71 Benoit Larose, Cynthia Loten, and Claude Tardif. A characterisation of first-order constraint satisfaction problems. *Log. Methods Comput. Sci.*, 3(4):4:6, 22, 2007. `doi:10.2168/LMCS-3(4:6)2007`.

72 Benoit Larose, Cynthia Loten, and László Zádori. A polynomial-time algorithm for near-unanimity graphs. *J. Algorithms*, 55(2):177–191, 2005. `doi:10.1016/j.jalgor.2004.04.011`.

73 Benoit Larose and Mark Siggers. Reflexive digraphs admitting nu polymorphisms. In preparation.

74 Benoit Larose and Claude Tardif. A discrete homotopy theory for binary reflexive structures. *Adv. Math.*, 189(2):268–300, 2004. `doi:10.1016/j.aim.2003.11.011`.

75 Benoît Larose and Pascal Tesson. Universal algebra and hardness results for constraint satisfaction problems. *Theoret. Comput. Sci.*, 410(18):1629–1647, 2009. `doi:10.1016/j.tcs.2008.12.048`.

76 Benoit Larose and Ross Willard. Nu series parallel posets. preprint, 2016.

77 Benoit Larose and László Zádori. Algebraic properties and dismantlability of finite posets. *Discrete Math.*, 163(1-3):89–99, 1997. `doi:10.1016/0012-365X(95)00312-K`.

78 Benoit Larose and László Zádori. The complexity of the extendibility problem for finite posets. *SIAM J. Discrete Math.*, 17(1):114–121 (electronic), 2003. `doi:10.1137/S0895480101389478`.

79 Benoit Larose and László Zádori. Finite posets and topological spaces in locally finite varieties. *Algebra Universalis*, 52(2-3):119–136, 2004. `doi:10.1007/s00012-004-1819-7`.

80 Cynthia Loten. *Retractions of chordal and related graphs.* PhD thesis, Simon Fraser University (Canada), 2004.

81 M. Maróti and L. Zádori. Reflexive digraphs with near unanimity polymorphisms. *Discrete Math.*, 312(15):2316–2328, 2012. `doi:10.1016/j.disc.2012.03.040`.

82 Miklós Maróti. Personal communication, 2016.

83 Miklós Maróti and Ralph McKenzie. Existence theorems for weakly symmetric operations. *Algebra Universalis*, 59(3-4):463–489, 2008. `doi:10.1007/s00012-008-2122-9`.

84 Ralph McKenzie. Monotone clones, residual smallness and congruence distributivity. *Bull. Austral. Math. Soc.*, 41(2):283–300, 1990. `doi:10.1017/S0004972700018104`.

85 Jaroslav Nešetřil and Claude Tardif. Duality theorems for finite structures (characterising gaps and good characterisations). *J. Combin. Theory Ser. B*, 80(1):80–97, 2000. `doi:10.1006/jctb.2000.1970`.

86 Christos H. Papadimitriou. *Computational complexity.* Addison-Wesley Publishing Company, Reading, MA, 1994.

87 Mark Siggers. Distributive lattice polymorphism on reflexive graphs. arXiv:1411.7879, 2014.

88 Mark Siggers, 2016. Personal communication.

89 Mark Siggers. Reflexive graphs with near-unanimity but no semilattice polymorphisms. preprint, 2016.

90 Narayan Vikas. Computational complexity classification of partition under compaction and retraction. In *Computing and combinatorics*, volume 3106 of *Lecture Notes in Comput. Sci.*, pages 380–391. Springer, Berlin, 2004. `doi:10.1007/978-3-540-27798-9_41`.

91 Ross Willard, 2015. Personal communication.

92 Alexander Wires. Dichotomy for finite tournaments of mixed-type. *Discrete Math.*, 338(12):2523–2538, 2015. `doi:10.1016/j.disc.2015.06.024`.

93 Alexander Wires. A quasi-Mal'cev condition with unexpected application. *Algebra Universalis*, 73(3-4):335–346, 2015. `doi:10.1007/s00012-015-0322-7`.