Posets, graphs and algebras: a case study for the fine-grained complexity of CSP's

Part 3: More Evidence: Graphs and Posets

Benoit Larose [1] [2]

[1]Department of Mathematics and Statistics
Concordia University, Montréal

[2]Department of Mathematics
Champlain Regional College, St-Lambert

SSAOS, Slovakia, September 2009

Recap of Talk 2

- to each CSP we associate an idempotent algebra $\mathbb{A}$;
- we conjecture that the typeset of $\mathcal{V}(\mathbb{A})$ "controls" the (descriptive and algorithmic) complexity of $CSP(\mathbf{H})$;
- there is some good evidence supporting these conjectures.

Overview of Talk 3

- We investigate CSP's whose target structures are related to digraphs, graphs and posets:
- Feder-Vardi have shown that the Dichotomy Conjecture can settled by looking only at these special cases;
- a natural setting;
- a good testing ground for the conjectures;
- we can use tools from graph theory and topology to investigate some of these problems;

Overview of Talk 3, cont'd

- We present a complete classification in the cases of:
  - list homomorphisms of graphs;
  - series-parallel posets.
- more generally, we address the problem: what graphs, digraphs, posets admit (no) nice identities ?
- we give several open problems as we go along.

Preliminaries

Definition

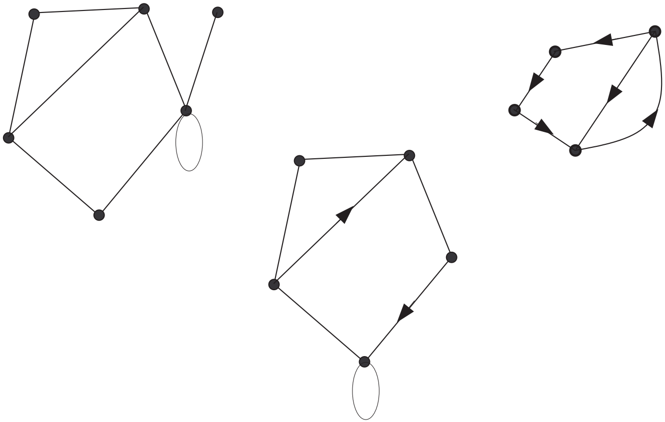
A *digraph* is a structure $\mathbf{H} = \langle H; \theta \rangle$ with a single binary relation θ. We say $\mathbf{H}$ is a

- *graph*, if θ is symmetric: $(a, b) \in \theta$ iff $(b, a) \in \theta$;
- a *poset*, if θ is
  - reflexive: $(x, x) \in \theta$ for all x;
  - antisymmetric: $(a, b), (b, a) \in \theta \Rightarrow a = b$;
  - transitive: $(a, b), (b, c) \in \theta \Rightarrow (a, c) \in \theta$.

Remark: Our graphs may have loops on certain vertices.

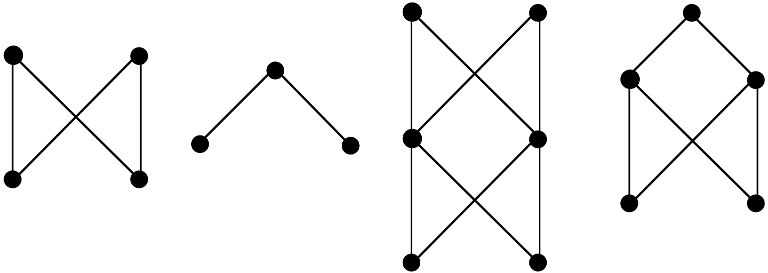
Pictures of digraphs

Some graphs and digraphs:



Pictures of posets

We depict posets by their Hasse diagrams:



List Homomorphism Problems

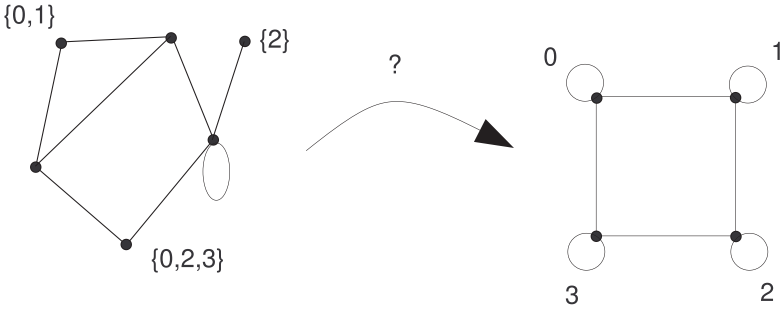
Given a structure $\mathbf{H}$, the *list homomorphism problem for* $\mathbf{H}$ is $CSP(\mathbf{H}')$ where $\mathbf{H}'$ is the structure obtained from $\mathbf{H}$ by adding ALL subsets of H as unary relations. Formally: If $\mathbf{H} = \langle A; \theta_1, \dots, \theta_r \rangle$, let

$$\mathbf{H}' = \langle A; \theta_1, \dots, \theta_r, B (B \subseteq A) \rangle.$$

Shorthand:

$$CSP(\mathbf{H}') = CSP(\mathbf{H} + \mathit{lists}).$$

List Homomorphism Problems, cont'd



Motivation for $CSP(\mathbf{H} + \mathit{lists})$

- natural, well-studied for graphs;
- algebraic dichotomy holds (Bulatov);
- easier to handle because of forbidden induced substructures;
- algebraically easier: 2-element divisors must appear as subalgebras.

Retraction Problems

Given a structure $\mathbf{H}$, the *retraction problem for* $\mathbf{H}$ is $CSP(\mathbf{H}')$ where $\mathbf{H}'$ is the structure obtained from $\mathbf{H}$ by adding all one-element subsets of H as unary relations. Formally: if $\mathbf{H} = \langle A; \theta_1, \ldots, \theta_r \rangle$, let

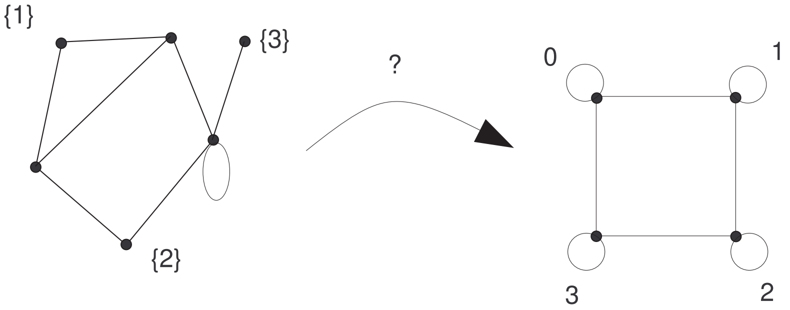
$$\mathbf{H}' = \langle A; \theta_1, \ldots, \theta_r, \{a\}(a \in A) \rangle.$$

Shorthand:

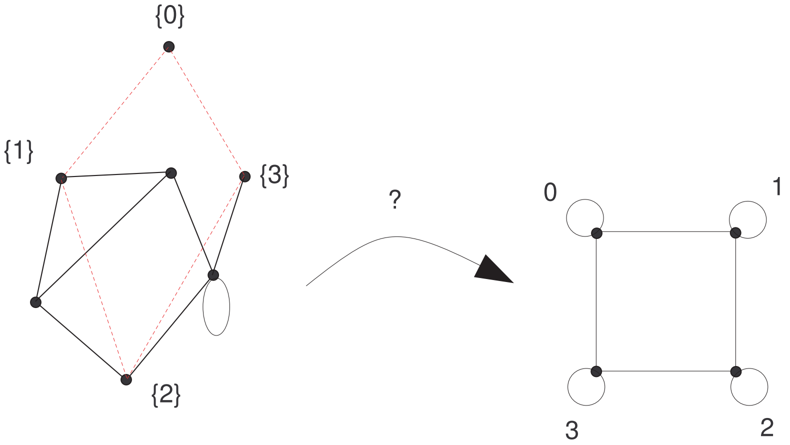
$$CSP(\mathbf{H} + csts)$$

Note: aka the *one-or-all list homomorphism problem*.

Retraction Problems, cont'd



Why "Retraction" ?



Motivation for $CSP(\mathbf{H} + csts)$

- natural problem;
- when target has a loop, CSP is trivial;
- target $\mathbf{H} + csts$ is automatically a core;
- algebraically: corresponds to finding idempotent polymorphisms of the structure $\mathbf{H}$;
- and see next result.

Note: Not as well-understood as the list case, as the next result shows.

Reductions

Theorem (FV 98; Feder, Hell 98)

Let $\mathbf{H}$ *be a structure. Then there exist a poset* $\mathbf{P}$, *a bipartite graph* $\mathbf{Q}$, *a reflexive graph* $\mathbf{R}$ *and a digraph* $\mathbf{S}$ *such that the following problems are poly-time equivalent:*

- $CSP(\mathbf{H})$;
- $CSP(\mathbf{P} + csts)$;
- $CSP(\mathbf{Q} + csts)$;
- $CSP(\mathbf{R} + csts)$;
- $CSP(\mathbf{S})$.

Reductions, cont'd

Some drawbacks of these reductions:

- not known to be logspace reductions
  (not fine enough to see what's in $\mathcal{L}$, $\mathcal{NL}$, etc.)
- do not behave so well with respect to the associated algebras.

Reductions, cont'd

However: for each structure $\mathbf{H}$ one may construct a structure $\mathbf{H}'$ with only unary and binary relations such that

- $CSP(\mathbf{H})$ and $CSP(\mathbf{H}')$ are equivalent under logspace reductions;
- the reduction respects expressibility in (linear, symmetric) Datalog;
- the binary relations are graphs of permutations and equivalence relations (McKenzie).

We shall not require this result in what follows.

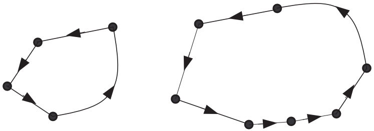
Results on digraphs: $CSP(\mathbf{H})$

- Let $\mathbf{H}$ be a digraph.
- By FV classifying the complexity of $CSP(\mathbf{H})$ is as hard as the general case. But some special cases have been determined:
- A vertex in a digraph is a *source (sink)* if it has no incoming (outgoing) edges.

Theorem (Barto, Kozik, Niven (2009))

Let $\mathbf{H}$ be a digraph with no sources and no sinks. Then $CSP(\mathbf{H})$ is in $\mathcal{P}$ if the core of $\mathbf{H}$ is a disjoint union of directed cycles, and it is $\mathcal{NP}$-complete otherwise.

Results on digraphs: $CSP(\mathbf{H})$, cont'd



- first conjectured by Bang-Jensen and Hell in 1990;
- proof uses algebraic methods: if $\mathbf{H}$ is invariant under a weak NU operation then its core is a disjoint union of cycles;
- if $\mathbf{H}$ is a disjoint union of cycles, then its binary relation is the graph of a permutation; consequently $\neg CSP(\mathbf{H})$ is in symmetric Datalog and $CSP(\mathbf{H})$ is $\mathcal{L}$-complete (ELT 07).

Results on digraphs: $CSP(\mathbf{H})$, cont'd

Definition

Let $n \geq 2$. An n-ary operation t is *totally symmetric (TSI)* if it is idempotent and $t(x_1, \ldots, x_n) = t(y_1, \ldots, y_n)$ whenever $\{x_1, \ldots, x_n\} = \{y_1, \ldots, y_n\}$.

Example

Let $\wedge$ be a semilattice operation (idempotent, commutative, associative.) For any $n \geq 2$, the operation

$$t(x_1, \ldots, x_n) = x_1 \wedge x_2 \wedge \cdots \wedge x_n$$

is a TSI operation.

Results on digraphs: $CSP(\mathbf{H})$, cont'd

- $\neg CSP(\mathbf{H})$ is in $(1, k)$-Datalog for some k (aka *tree duality*) iff $\mathbf{H}$ is invariant under TSI operations of all arities $n \geq 2$ (Dalmau, Pearson, 1999);
- Barto, Kozik, Maroti and Niven (2009) have proved dichotomy for "special triads"; the tractable cases either admit TSI's of all arities or a majority operation.
- Result extended by Bulín (2009) to "special polyads":
  - proof invokes the BW Theorem: if polyad admits a weak NU then it admits weak NU's for all but finitely many arities;
  - hence $\neg CSP(\mathbf{H})$ is in Datalog.
- refined complexity for triads is being investigated (A. Lemaître)

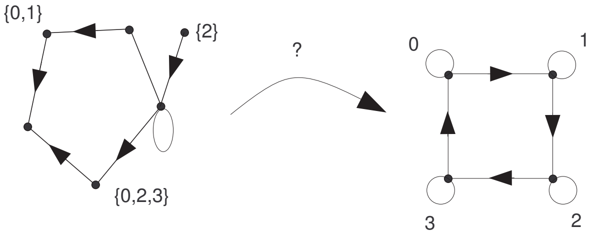
Results on digraphs: $CSP(\mathbf{H} + lists)$

- Let **H** be a digraph; consider the problem $CSP(\mathbf{H} + lists)$.
- We know that dichotomy holds in the list case;
- but can we find a "nice" (graph-theoretic ?) description of the tractable cases ? This should help to understand the refined complexity.
- The case of reflexive digraphs is nice:

List homomorphisms on reflexive digraphs

Theorem (Carvalho, Feder, Hell, Huang, Rafiey (TBA))

Let **H** *be a reflexive digraph. If* **H** *admits a weak NU, then it admits a semilattice polymorphism, and CSP*(**H**) *is in* $\mathcal{P}$*; otherwise it is* $\mathcal{NP}$*-complete.*



List homomorphisms on reflexive digraphs, cont'd

- Notice: if $\mathbf{H} + \textit{lists}$ admits a semilattice operation $\wedge$, it preserves every subset of $\mathbf{H}$;
- hence $a \wedge b \in \{a, b\}$ for all a, b;
- i.e. there exists some ordering of the vertices such that $a \wedge b = \min(a, b)$ for all $a, b, \in H$.

An aside on reflexive digraphs

- the category of reflexive digraphs is equipped with a nice homotopy theory (BL, Tardif, 2004);
- coincides with the usual homotopy for posets;
- the nature of the homotopy groups of $\mathbf{H}$ is closely related to the algebra $\mathbb{A}(\mathbf{H})$:

Theorem (BL, 2006)

Let $\mathbf{H}$ be a connected, reflexive digraph and let $\mathbb{A} = \mathbb{A}(\mathbf{H})$. If $\mathbb{A}$ admits a weak NU operation then every homotopy group of $\mathbf{H}$ is trivial.

- a useful tool to prove hardness results;
- some evidence that perhaps there is more to this story (see Posets);

Results on graphs: $CSP(\mathbf{H})$

Theorem (Hell, Nešetřil, 1990)

Let $\mathbf{H}$ *be a graph. If* $\mathbf{H}$ *has a loop or is bipartite, then* $CSP(\mathbf{H})$ *is in* $\mathcal{P}$*; otherwise it is* $\mathcal{NP}$*-complete.*

- Notice: this is a special case of the Barto et al. result on digraphs without sources and sinks;
- result has been refined independently by Bulatov (05), Kún & Szegedy (09), Siggers (09):

Theorem

If a graph $\mathbf{H}$ *is non-bipartite and has no loops then it admits no weak NU polymorphism.*

Results on graphs: $CSP(\mathbf{H} + lists)$

- Let $\mathbf{H}$ be a graph.
- there is a complete classification of the complexity of $CSP(\mathbf{H} + lists)$;
- our starting point is the following dichotomy result:

Theorem (Feder, Hell, Huang, 1999)

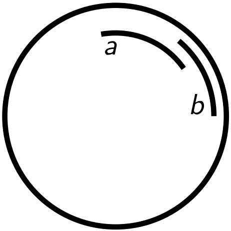
Let $\mathbf{H}$ be a graph. Then t.f.a.e.:

1. $\mathbf{H} + lists$ *admits a majority operation;*
2. $\mathbf{H}$ *is a bi-arc graph.*

If this condition is satisfied then $CSP(\mathbf{H} + lists)$ is in $\mathcal{P}$, otherwise it is $\mathcal{NP}$-complete.

Classification of $CSP(\mathbf{H} + lists)$

- (FHH) a graph $\mathbf{H}$ is *bi-arc* iff $\mathbf{H} \times \mathbf{K}_2$ is the complement of a *circular arc graph*:
- vertices are arcs; vertices are adjacent if the corresponding arcs intersect.



- odd cycles, 6-cycle are NOT bi-arc graphs.

Classification of $CSP(\mathbf{H} + \textit{lists})$, cont'd

First we confirm the algebraic dichotomy conjecture:

Lemma (Egri, Krokhin, BL, Tesson, 2009)

Let $\mathbf{H}$ *be a graph. If* $\mathbf{H} + \textit{lists}$ *admits a weak NU then it admits a majority operation.*

- it follows that $CSP(\mathbf{H} + \textit{lists})$ is either $\mathcal{NP}$-complete, else $\neg CSP(\mathbf{H} + \textit{lists})$ is in linear Datalog.
- it remains to determine for which graphs the problem is in symmetric Datalog (and which are FO).

Classification of $CSP(\mathbf{H} + \mathit{lists})$, cont'd

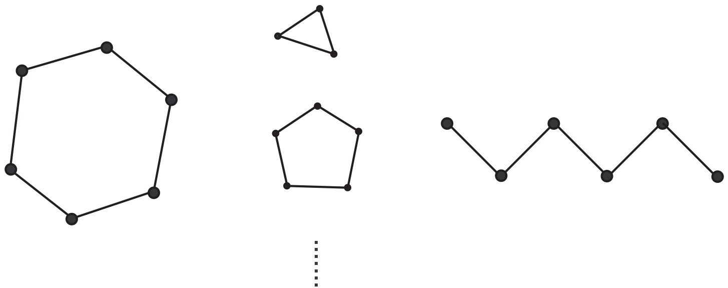
Let $\mathbf{H}$ be a graph, let $\mathbb{A}$ be the algebra associated to $\mathbf{H} + \mathit{lists}$.

- Strategy: to characterize graphs $\mathbf{H}$ such that $\mathcal{V}(\mathbb{A})$ omits types 1, 2, 4, 5 (i.e. pure type 3);
- we sieve to eliminate as much “bad guys” as possible;
- hopefully we can get a nice description of the remaining graphs to show the corresponding problem is in symmetric Datalog.

Classification of $CSP(\mathbf{H} + \mathit{lists})$, cont'd

To illustrate we consider the irreflexive case (graphs with no loops):

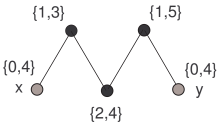
- the bad guys are: odd cycles, the 6-cycle, and the 5-path;

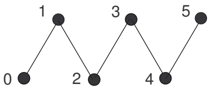


Classification of $CSP(\mathbf{H} + lists)$, cont'd

An illustration: Why the 5-path is bad:

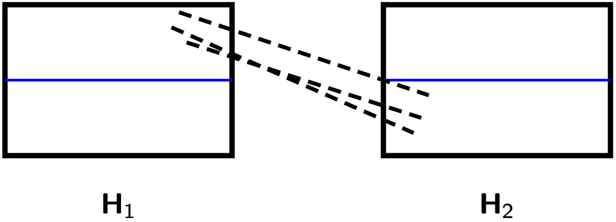
- the 5-path is a bi-arc graph, so admits a majority operation and hence $\mathcal{V}(\mathbb{A})$ omits types 1, 2 and 5;
- we produce (by pp-definability) a 2-element subalgebra with monotone terms;
- hence this divisor is of type 4.





Classification of $CSP(\mathbf{H} + lists)$, cont'd

- Let $Good_I$ be the family of irreflexive graphs $\mathbf{H}$ that have no induced 6-cycle, odd cycle or 5-path.
- We give an inductive definition of this family:
- define the *special sum* of two bipartite graphs $\mathbf{H}_1$ and $\mathbf{H}_2$ as follows: connect every vertex of one colour class of $\mathbf{H}_1$ to every vertex of one colour class of $\mathbf{H}_2$:



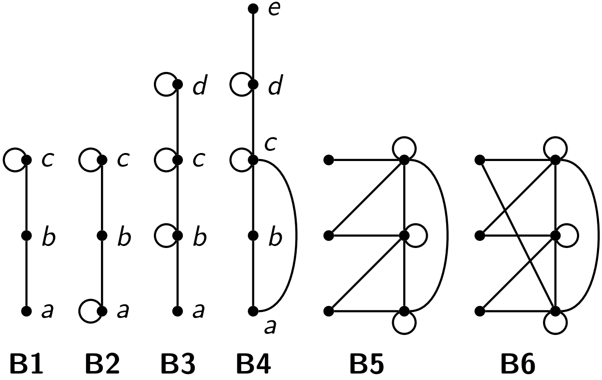
Classification of $CSP(\mathbf{H} + lists)$, cont'd

Lemma

$Good_I$ is the smallest class of irreflexive graphs containing the one-element graph and closed under disjoint union and special sum.

- The general case is handled in a similar way;
- the inductive definition is only slightly more involved;
- let *Good* denote the class of graphs that avoid the following forbidden subgraphs:
  - the irreflexive 6-cycle, odd cycles and 5-path;
  - the reflexive 4-cycle and 4-path;
  - and the following "mixed" graphs:

Classification of $CSP(\mathbf{H} + \mathit{lists})$, cont'd



Classification of $CSP(\mathbf{H} + lists)$, cont'd

Theorem (E,K,BL,T)

Let $\mathbf{H}$ *be a graph, and let* $\mathbb{A}$ *be the algebra associated to* $\mathbf{H} + lists$*. Then t.f.a.e.:*

1. $\mathbf{H} \in Good$;
2. $\mathcal{V}(\mathbb{A})$ *is pure type 3;*
3. $\mathcal{V}(\mathbb{A})$ *is 4-permutable;*
4. $\neg CSP(\mathbf{H} + lists)$ *is expressible in symmetric Datalog.*

If these conditions hold then $CSP(\mathbf{H} + lists)$ *is in* $\mathcal{L}$*; otherwise it is* $\mathcal{NL}$*-complete (and* $\neg CSP(\mathbf{H} + lists)$ *is expressible in linear Datalog) or it is* $\mathcal{NP}$*-complete.*

Results on Posets: $CSP(\mathbf{Q} + csts)$

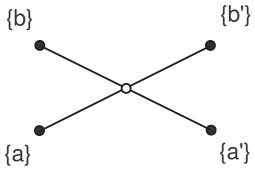
- Let $\mathbf{Q}$ be a poset.
- Since $\mathbf{Q}$ is reflexive, the problem $CSP(\mathbf{Q})$ is trivial, hence we consider the problem $CSP(\mathbf{Q} + csts)$;
- by FV this problem is as hard as the general case;
- several special cases are of interest (e.g. only family of maximal clones whose complexity is not classified);
- $CSP(\mathbf{Q} + lists)$ is a special case of the reflexive digraph problem (already under investigation !)

Results on Posets: $CSP(\mathbf{Q} + csts)$, cont'd

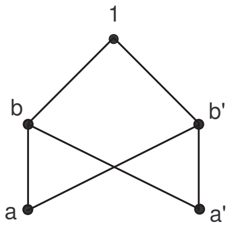
- Remarks on the preprimal algebra (maximal clone) 6th case:
- for any bounded poset $\mathbf{Q}$, the variety admits type 4, hence $CSP(\mathbf{Q} + csts)$ is $\mathcal{NL}$-hard (and not expressible in symmetric Datalog);
- one can construct various examples of bounded posets $\mathbf{Q}$ such that $CSP(\mathbf{Q} + csts)$ is in $\mathcal{P}$ but the variety admits type 2, or type 5, etc.
- hence even the special case of *bounded* posets appears to be quite complicated.
- Now back to general posets:

Results on Posets: $CSP(\mathbf{Q} + csts)$, cont'd

- Consider for a moment the special subproblem $\mathcal{S}$ of $CSP(\mathbf{Q} + csts)$, where the inputs are themselves posets;
- (Zádori) A $\mathbf{Q}$-*zigzag* is an input $\mathbf{P}$ to the problem $\mathcal{S}$ such that
  - there is no homomorphism from $\mathbf{P}$ to $\mathbf{Q}$;
  - every proper substructure of $\mathbf{P}$ (in $\mathcal{S}$) admits a homomorphism to $\mathbf{Q}$;



P



Q

Results on Posets: $CSP(\mathbf{Q} + csts)$, cont'd

Theorem (Zádori, 1993)

Let $\mathbf{Q}$ *be a connected poset. Then t.f.a.e.:*

1. $\mathbf{Q}$ *admits an NU operation;*
2. *there are only finitely many* $\mathbf{Q}$*-zigzags.*

It will follow from this result that in the case of posets, presence of an NU operation implies expressibility in linear Datalog:

Results on Posets: NU implies linear Datalog

Theorem

Let $\mathbf{Q}$ *be a connected poset. If* $\mathbf{Q}$ *admits an NU operation then* $\neg CSP(\mathbf{Q} + csts)$ *is expressible in linear Datalog.*

Sketch of proof:

- let $\mathbf{R}$ be an input structure; one may (easily) construct a poset $\mathbf{R}'$ from $\mathbf{R}$ using pp-definitions and transitive closure, such that $\mathbf{R}'$ admits a homomorphism to $\mathbf{Q}$ iff $\mathbf{R}$ does;
- hence $\mathbf{R}$ does not map to $\mathbf{Q}$ iff some $\mathbf{Q}$-zigzag $\mathbf{P}$ maps to $\mathbf{R}'$;
- the existence of the map from $\mathbf{P}$ to $\mathbf{R}'$ is easily encoded as a sentence in positive FO with transitive closure;
- since there are finitely many zigzags, $\neg CSP(\mathbf{Q} + csts)$ is in pos(FO + TC), and hence in linear Datalog (Dalmau, Krokhin, BL).

Results on Posets: linear Datalog, cont'd

Corollary

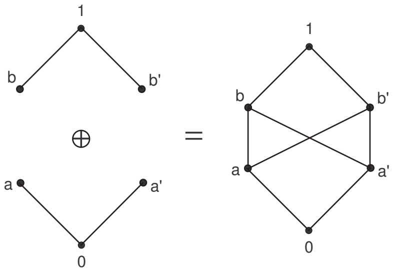
Let $\mathbf{Q}$ be a connected poset, and let $\mathbb{A} = \mathbb{A}(\mathbf{Q} + csts)$. If $\mathcal{V}(\mathbb{A})$ is congruence-modular then $\neg CSP(\mathbf{Q} + csts)$ is expressible in linear Datalog, and $CSP(\mathbf{Q} + csts)$ is in $\mathcal{NL}$. If $\mathbf{Q}$ is bounded, then $CSP(\mathbf{Q} + csts)$ is $\mathcal{NL}$-complete.

- it is known that congruence-modularity, congruence-distributivity and NU are equivalent conditions for posets (BL, Zádori, 1997);
- bounded case: follows from earlier remark;
- there are cases in linear Datalog that are not congruence-modular (see 5 element poset 3 slides ago).

Results on Posets: The Series-Parallel Case

Definition

Let $\mathbf{Q}_1$ and $\mathbf{Q}_2$ be two posets; the *(ordinal) sum* $\mathbf{Q}_1 \bigoplus \mathbf{Q}_2$ of $\mathbf{Q}_1$ and $\mathbf{Q}_2$ is the poset obtained from their disjoint union by making every element of $\mathbf{Q}_1$ smaller than every element of $\mathbf{Q}_2$.



Results on Posets: The Series-Parallel Case, cont'd

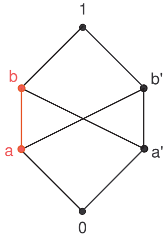
Definition

The class of *series-parallel* posets is the smallest containing the one-element poset and closed under disjoint union and ordinal sum.

Remark: these are also known as "N-free" posets: they are precisely the posets that do not contain an induced poset isomorphic to N.

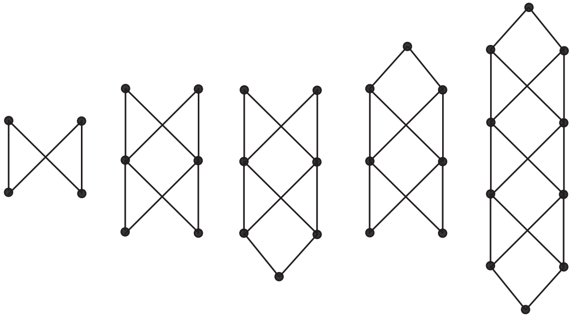
Results on Posets: The Series-Parallel Case, cont'd

- we say a (induced) subposet $\mathbf{P}$ of $\mathbf{Q}$ is a *subalgebra* of $\mathbf{Q}$ if its universe is a subuniverse of the algebra $\mathbb{A} = \mathbb{A}(\mathbf{Q} + csts)$.
- it is easy to see that every covering pair is a 2-element subalgebra of $\mathbf{Q}$; in particular $\mathcal{V}(\mathbb{A})$ admits type 1, 4 or 5;



Results on Posets: The Series-Parallel Case, cont'd

- we say that **Q** *retracts* onto **P** if there exist maps $R : \mathbf{Q} \longrightarrow \mathbf{P}$ and $e : \mathbf{P} \longrightarrow \mathbf{Q}$ such that $r \circ e = id_P$;
- the posets below turn out to characterise the "bad" series-parallel posets (via retractions):



Results on Posets: The Series-Parallel Case, cont'd

Theorem (Dalmau, Krokhin, BL, 2008)

Let $\mathbf{Q}$ be a connected series-parallel poset. Then t.f.a.e:

1. $\mathbf{Q}$ *admits a weak NU operation;*
2. $\mathbf{Q}$ *admits TSI operations of all arities;*
3. *every connected subalgebra of* $\mathbf{Q}$ *has a trivial fundamental group;*
4. $\mathbf{Q}$ *does not retract on any of the posets pictured above.*

If any of these conditions hold then CSP$(\mathbf{Q})$ *is in* $\mathcal{P}$*; otherwise it is* $\mathcal{NP}$*-complete.*

Results on Posets: The Series-Parallel Case, cont'd

- for series-parallel posets, we can say a bit more in the tractable case:
- it turns out one can express the condition that a poset **P** does NOT retract to **Q** in pos(FO+TC);
- we can conclude as before that $\neg CSP(\mathbf{Q} + csts)$ is in linear Datalog;
- since posets will always admit type 1, 4 or 5, this is the best we can hope for and the classification is complete.