A maximally tractable fragment of temporal reasoning plus successor

Antoine Mottet
Laboratoire d'Informatique de l'École Polytechnique

August 21, 2014

1. Introduction

A constraint satisfaction problem (CSP) is a decision problem that consists in deciding for a given set of variables and constraints over these variables whether there exists a mapping of the variables to a fixed domain such that the image of the variables satisfies the given constraints. The example that illustrates the basic definitions of this introduction is the graph 3-colourability problem. Even without a formal definition, it is easily seen to be a constraint satisfaction problem: the variables are the vertices of the graph and have to be assigned to the domain $\{0, 1, 2\}$, and the constraints say that two adjacent vertices have to be assigned to different values.

Definitions

We can formalize the notion of constraints using signatures and structures:

Definition 1. A *relational signature* σ is a set of relation symbols where each symbol is given an arity $k \geq 1$.

A *relational σ-structure* $\mathbf{A}$ consists of a set A, called the domain of the structure, and for each k-ary relation symbol R of σ the structure provides an *interpretation* $R^{\mathbf{A}} \subseteq A^k$. We assume in the following that σ always has the binary relation symbol $=$ whose interpretation in any σ-structure $\mathbf{A}$ is the equality relation $\{(x, x) \mid x \in A\}$.

When the meaning is clear, we denote by R both a symbol in a signature and its interpretation in a structure. Furthermore if A is a set and $R_1, \ldots, R_n$ are relations on A, we denote by $\mathbf{A} = (A, R_1, \ldots, R_n)$ the structure with signature $\sigma = \{R_1, \ldots, R_n\}$ such that $R_i^{\mathbf{A}} = R_i$. If $\mathbf{A}$ is a σ-structure and R is a relation on A, we denote by $(\mathbf{A}, R)$ the $(\sigma \cup \{R\})$-structure obtained by letting $R^{\mathbf{A}} = R$.

Let us go back to our example, with the signature $\sigma = \{E\}$, E being a relation symbol of arity 2. Let $V = \{0, 1, 2\}$, and let E^{K_3} be $V^2 \setminus \{(0, 0), (1, 1), (2, 2)\}$. Then $K_3 = (V, E^{K_3})$ is a σ-structure, and the problem of assigning vertices of a graph to K_3 so that if v and w are adjacent then the image of (v, w) is in E^{K_3} is exactly the 3-colourability problem.

Structures can be seen as databases: they express facts about points of the domain, and they can be queried using various languages. To know if the 5-cycle is three colourable, we query K_3 with the formula

$$\Phi := \exists x_1, \ldots, x_5. \bigwedge_{i=1}^{4} E(x_i, x_{i+1}) \wedge E(x_5, x_1).$$

If K_3 satisfies this formula, which we denote by the relation $K_3 \models \Phi$, then the 5-cycle is 3-colourable, otherwise it is not.

Using this database point of view, we can already grasp what the heart of the problem is: given a query (a sentence) to be evaluated on a database (a structure) $\mathbf{A}$, what is the complexity of evaluating this query on $\mathbf{A}$? In the study of constraint satisfaction problems, we restrict our attention to evaluating queries that have a particular form. A formula in the signature σ is said to be *primitive positive* (pp) if it consists of existential quantifiers, conjunctions, variables and symbols from the signature σ. In the following, the symbols x, y, z, u, v stand for variables that are supposed to be different, unless otherwise stated. A pp-*sentence* is a primitive positive formula where all variables are bound by a quantifier. The constraint satisfaction problem of a structure (in a finite signature) can be defined as:

Definition 2. Let $\mathbf{A}$ be a relational σ-structure, where σ is finite. The problem $\mathrm{CSP}(\mathbf{A})$ is the following decision problem:
INPUT: A primitive positive sentence Φ over the signature σ
OUTPUT: Does $\mathbf{A} \models \Phi$?

As we have seen before, $\mathrm{CSP}(K_3)$ is the 3-colourability problem: a graph $G = (V, E)$ has a *canonical query* Φ_G that can be used as input to $\mathrm{CSP}(K_3)$. The variables of Φ_G are the vertices of G, and for each edge $(v, w) \in E$, one adds the conjunct $E(v, w)$ to Φ_G. It is clear that G is 3-colourable if and only if $K_3 \models \Phi_G$. Conversely, a pp-sentence Φ yields a graph G_Φ whose vertices are the variables of Φ and there is an edge (v, w) in G if $E(v, w)$ or $E(w, v)$ appears in Φ. Again, we have that G_Φ is 3-colourable if and only if $K_3 \models \Phi$, so that $\mathrm{CSP}(K_3)$ is indeed the 3-colourability problem.

Algorithmic results

We study here the complexity of the CSP over the set of the rational numbers where the relations in the signature are defined by:

1. $y = x + 1$,
2. $x = y \to u = v$, and
3. $x > y \vee x > z \vee x = y = z$.

The corresponding structure is denoted by $\mathbf{L}$, and $+_1$ is the relation symbol that we use to represent the first relation. It was shown in [**?**] that when the language consists of the last two relations, the corresponding satisfaction problem is solvable in polynomial time. However, it is not clear how to adapt the algorithm presented there so that it can deal with constraints of the form $y = x + 1$.

We first use the technique of *sampling* presented in [**?**] to establish that a particular class of CSPs are polynomial-time tractable — namely the class of CSPs that arise from structures over the rational numbers that are min-closed. Using this result, we provide a polynomial-time algorithm for $\mathrm{CSP}(\mathbf{L})$. The algorithmic results are presented in Sections 2, 4, and 5.

Reducts and maximality result

Given a σ-structure $\mathbf{A}$ (that we call the *ground structure*), one can define new relations on the domain of $\mathbf{A}$ using first-order formulas. For example, in the structure $(\mathbb{Q}, <)$ one can define the ternary relation $Betw = \{(a, b, c) \in \mathbb{Q}^3 \mid a < b < c \vee c < b < a\} \subset \mathbb{Q}^3$. More rigorously:

Definition 3. Let $\mathbf{A}$ be a structure with domain A, and let $R \subseteq A^n$ be an n-ary relation. We say that R is *first-order (resp. pp) definable* in $\mathbf{A}$ if there exists a first-order (resp. primitive positive) formula $\Phi(x_1, \ldots, x_n)$ such that for all $(a_1, \ldots, a_n) \in A^n$,

$$(a_1, \ldots, a_n) \in R \Leftrightarrow \mathbf{A} \models \Phi(a_1, \ldots, a_n).$$

Let $\mathbf{A}, \mathbf{B}$ be structures. If all the relations of $\mathbf{B}$ have a first-order definition in $\mathbf{A}$ we say that $\mathbf{B}$ is a *first-order reduct* of $\mathbf{A}$ (or simply *reduct*). For example, $(\mathbb{Q}, Betw)$ and $(\mathbb{Q}, =)$ are both first-order reducts of $(\mathbb{Q}, <)$. If the structure $\mathbf{B}$ is a reduct of $\mathbf{A}$ and is so that all the relations of $\mathbf{A}$ are also in $\mathbf{B}$, we say that $\mathbf{B}$ is an *expansion* of $\mathbf{A}$.

All these reducts give rise to new CSPs, and the *classification project* for the ground structure $\mathbf{A}$ is the task of determining for each finite-signature first-order reduct $\mathbf{B}$ of $\mathbf{A}$ the complexity of CSP($\mathbf{B}$).

A very basic, yet important, fact about primitive positive definitions is the following:

Lemma 1. *Let $\mathbf{A}, \mathbf{B}$ be finite-signature structures over the same domain such that all the relations of $\mathbf{A}$ are pp-definable in $\mathbf{B}$. Then* CSP($\mathbf{A}$) $\leq_p$ CSP($\mathbf{B}$)*.*

Proof. Let Φ be an input to CSP($\mathbf{A}$). Since all relations of $\mathbf{A}$ are pp-definable in $\mathbf{B}$, we can replace each atom $R_i(\mathbf{y})$ (where $\mathbf{y}$ is a tuple of variables extracted from $\{x_1, \ldots, x_n\}$) in Φ by a pp-definition Ψ_i of R_i in $\mathbf{B}$. Let us write Ψ_i as $\exists \mathbf{z}. \bigwedge T_j(\mathbf{y}, \mathbf{z})$, where each T_j is a symbol from the signature of $\mathbf{B}$. Putting the resulting formula in prenex form (with all the quantifiers at the front of the formula), we obtain an input Ψ to CSP($\mathbf{B}$). We now have

$$\begin{aligned}
\mathbf{A} \models \Phi &\Leftrightarrow \mathbf{A} \models \exists x_1, \ldots, x_n. \bigwedge R_i(\mathbf{y}) \\
&\Leftrightarrow \mathbf{B} \models \exists x_1, \ldots, x_n. \bigwedge \Psi_i(\mathbf{y}, \mathbf{z}) \\
&\Leftrightarrow \mathbf{B} \models \exists x_1, \ldots, x_n, z_1, \ldots, z_l. \bigwedge T_i(\mathbf{y}, \mathbf{z}) \\
&\Leftrightarrow \mathbf{B} \models \Psi.
\end{aligned}$$

This reduction is easily seen to be polynomial-time. □

This lemma implies that if all relations of $\mathbf{A}$ are pp-definable in $\mathbf{B}$ and vice versa, CSP($\mathbf{A}$) and CSP($\mathbf{B}$) are polynomial-time equivalent. Thus, it makes sense from a complexity point of view to study first-order reducts of some ground structure up to primitive positive inter-definability. If we order the class of first-order reducts of some given ground structure so that $\mathbf{A} \leq \mathbf{B}$ if all the relations of $\mathbf{A}$ are pp-definable in $\mathbf{B}$, we obtain an infinite lattice where we can draw a border between reducts:

- if all finite-signature structures $\mathbf{B}$ such that $\mathbf{B} \leq \mathbf{A}$ are so that CSP($\mathbf{B}$) is in P, we say that $\mathbf{A}$ is tractable,
- otherwise we say that $\mathbf{A}$ is hard.

Establishing the complexity of reducts of the ground structure is then the task of investigating this lattice and finding those structures $\mathbf{A}$ that are tractable, but so that for every $\mathbf{B}$ such that $\mathbf{A} < \mathbf{B}$, $\mathbf{B}$ is hard. In this case, $\mathbf{A}$ is said to be maximally tractable among the class of reducts of the ground structure. We prove in Section 6 that the structure **L** and the structure that consists of all min-closed relations are indeed maximal in the class of reducts of $\mathbf{Q} = (\mathbb{Q}, <, +_1)$.

This result is a first step in the project of classifying the complexity of the CSP of the first-order reducts of **Q**. Since **L** and min-closed relations are maximal, any attempt to this classification project has to deal with this case.

2. Syntactic description

In this section, we give a description of the relations that have a primitive positive definition over the initial set of constraints. Let us first recall the structure that we are working on:

Definition 4. Let **L** be the structure over the domain $\mathbb{Q}$ whose relations are:

1. $\{(a, b) \mid b = a + 1\}$,
2. $\{(a, b, c, d) \mid a = b \to c = d\}$, and
3. $\{(a, b, c) \mid a > b \lor a > c \lor a = b = c\}$.

As stated in Lemma 1, a structure that is obtained by adding finitely many relations that have a pp-definition over **L** to the ones of **L** has a polynomial-time equivalent CSP. Furthermore, having a description of these relations allows us to apprehend the closure properties of **L**, as shown in Section 3.

First, note that the relations $\{(a, b) \mid b = a + p\}$ and $\{(a, b) \mid b > a + p\}$ are definable over **Q** for all $p \in \mathbb{Z}$, so that in the following we will use abbreviations like $z = y + p$ and $z > y + p$, where those abbreviations are to be understood as formulas over the language of **Q**. Since these relations are even pp-definable over **Q**, using the abbreviations in pp-formulas gives pp-formulas.

Definition 5. Let Φ be a formula over the language of **Q**. We say that Φ is ll-Horn if it is a conjunction of clauses of the form

$$(x_1 = y_1 + p_1 \land \cdots \land x_k = y_k + p_k) \to (z > u_1 + q_1 \lor \cdots \lor z > u_l + q_l)$$

or

$$\begin{aligned}(x_1 = y_1 + p_1 \land \cdots \land x_k = y_k + p_k) \to (&z > u_1 + q_1 \lor \cdots \lor z > u_l + q_l \\ &\lor z = u_1 + q_1 = \cdots = u_l + q_l)\end{aligned}$$

for $k, l \geq 0$, $p_i, q_i \in \mathbb{Z}$ and where x_i, y_i, z, u_i are not necessarily distinct variables. A relation $R \subseteq \mathbb{Q}^n$ is said to be ll-Horn if there exists an ll-Horn formula that defines R over **Q**.

Note that the set of ll-Horn relations contains the relations defined by $x \geq y$, $x > y$, and $x \neq y$. We first prove that the set of ll-Horn relations is a subset of the set of relations that are pp-definable over **L**:

Lemma 2. *Let $R \subseteq \mathbb{Q}^n$ be an ll-Horn relation. Then R has a primitive positive definition over* **L**.

Proof. It is enough to prove that every formula of the above form is pp-definable in $\mathbf{L}$ if $p_i = 0$ and $q_i = 0$ for all i. Through pp-definition and with the relation $+_1$, it is then easy to express the formulas where p_i and q_i are not necessarily 0.

We first show how to pp-define over $\mathbf{L}$ the relation defined by $(x_1 = y_1 \wedge \ldots \wedge x_k = y_k) \rightarrow u = v$ for all $k \geq 1$. For $k = 1$, the relation is in the language of $\mathbf{L}$, so there is nothing to prove. For larger k, the following formula is a pp-definition of the relation over $\mathbf{L}$, assuming that the relation with $k-1$ is already defined:

$$\exists d. \big[((x_1 = y_1 \wedge \cdots \wedge x_{k-1} = y_{k-1}) \rightarrow u = d) \wedge \\ (x_k = y_k \rightarrow v = d) \big].$$

It is straightforward to show that this indeed defines the desired relation.

Secondly, the relation defined by $(z > u_1 \vee \cdots \vee z > u_l \vee z = x_1 = \cdots = x_l)$ has a pp-definition over $\mathbf{L}$ for all $l \geq 1$. For $l \leq 2$ this is obvious, and for larger l, we have the following definition over $\mathbf{L}$:

$$\exists d. \big[(z > u_1 \vee \cdots \vee z > u_{l-2} \vee z > d \vee (z = u_1 = \cdots = d)) \wedge \\ (d > x_{l-1} \vee d > x_l \vee d = x_{l-1} = x_l) \big].$$

Finally, the relation defined by $(z > u_1 \vee \cdots \vee z > u_l)$ has a pp-definition over $\mathbf{L}$ for all $l \geq 1$. For $l = 2$, we first note that the relation defined by $(x = y \rightarrow u \neq v)$ is pp-definable over $\mathbf{L}$ by $\exists d.(x = y \rightarrow u = d) \wedge (v = d \rightarrow d = d + 1)$. Then $(z > u_1 \vee z > u_2)$ is equivalent to $(z > u_1 \vee z > u_2 \vee z = u_1 = u_2) \wedge (z = u_1 \rightarrow z \neq u_2)$. For larger l, the following pp-defines the relation over $\mathbf{L}$:

$$\exists d. \left[(z > u_1 \vee \cdots \vee z > u_{l-2} \vee z > d) \wedge (d > u_{l-1} \vee d > u_l) \right].$$

Then the formula of the first type in the statement of the theorem is equivalent to $\exists d.((x_1 = y_1 \wedge \cdots \wedge x_k = y_k) \rightarrow z = d) \wedge (d > u_1 \vee \cdots \vee d > u_l)$ while the formula of the second type is equivalent to $\exists d.((x_1 = y_1 \wedge \cdots \wedge x_k = y_k) \rightarrow z = d) \wedge (d > u_1 \vee \cdots \vee d > u_l \vee (d = u_1 = \cdots = u_l))$, which shows that these formulas are pp-definable over the relations of $\mathbf{L}$. □

Given a relation $R \subseteq \mathbb{Q}^n$ that is defined by $\Phi(x_1, \ldots, x_n)$ over $\mathbf{Q}$, it may be hard to decide whether R is ll-Horn. Consider the following example:

Example 1. The 4-ary relation R defined by $(x > y \vee u > v) \wedge u \geq y \wedge x \geq v$ is ll-Horn. Indeed, R is also defined by the formula $(x = y \rightarrow u > v) \wedge (u = v \rightarrow x > y) \wedge u \geq y \wedge x \geq v$, which is ll-Horn.

We will see in Section 3 how to characterize ll-Horn relations as the set of relations that are closed under a certain operation, which will allow us to prove that the set of ll-Horn relations is precisely the set of relations that are pp-definable over $\mathbf{L}$ (thus proving the converse of Lemma 2).

3. Closure properties

We investigate here some properties of ll-Horn relations by means of polymorphisms.

Definition 6. Let $f \colon \mathbb{Q}^k \rightarrow \mathbb{Q}$ be a k-ary function, and let $R \subseteq \mathbb{Q}^n$ be an n-ary relation. We say that f *preserves* R, that f is a *polymorphism* of R, or that R is f-closed, if for all n-tuples $\mathbf{a}_1, \ldots, \mathbf{a}_k$ in R, we have that the tuple $f(\mathbf{a}_1, \ldots, \mathbf{a}_k)$ is in R, where f is applied component-wise.

If $\Phi(x_1, \ldots, x_n)$ is a formula that defines R over $\mathbf{Q}$, and if $s_1, \ldots, s_k$ are functions from $\{x_1, \ldots, x_n\}$ to $\mathbb{Q}$ that satisfy Φ (i.e. such that $\mathbf{Q} \models \Phi(s_i(x_1), \ldots, s_i(x_n)))$, we also write $f(s_1, \ldots, s_k)$ for the function $(x_1, \ldots, x_n) \mapsto (f(s_1(x_1), \ldots, s_k(x_1)), \ldots, f(s_1(x_n), \ldots, s_k(x_n)))$. In the following, we shall make no difference between a tuple $\mathbf{a} \in R$ and an assignment s that satisfies some formula Φ defining R over $\mathbf{Q}$.

When all the relations of a structure $\mathbf{A}$ are preserved by some operation f, we say that f is a polymorphism of $\mathbf{A}$, and we denote by $\mathrm{Pol}(\mathbf{A})$ the set of polymorphisms of $\mathbf{A}$. Conversely, for a set of operations $\mathcal{F}$, we denote by $\mathrm{Inv}(\mathcal{F})$ the set of relations $R \subseteq \mathbb{Q}^n$ that are f-closed for all $f \in \mathcal{F}$. Polymorphisms behave nicely with primitive positive definitions:

Lemma 3. *Let $\mathbf{A}$ be a structure, and let R be a relation with a pp-definition in $\mathbf{A}$. If $f \colon \mathbb{Q}^k \to \mathbb{Q}$ is a polymorphism of $\mathbf{A}$, then f preserves R. In symbols,*

$$R \in \mathrm{Inv}(\mathrm{Pol}(\mathbf{A})).$$

Remark 1. When $\mathbf{A}$ is finite, or when $\mathbf{A}$ is ω-categorical (see [?]), the converse is true: a relation has a pp-definition over $\mathbf{A}$ if, and only if, it is preserved by all polymorphisms of $\mathbf{A}$. Even though the structure $\mathbf{L}$ we study here is not ω-categorical, we will see that R is pp-definable in $\mathbf{L}$ iff $R \in \mathrm{Inv}(\mathrm{Pol}(\mathbf{L}))$.

The first closure property we present in this section is the following:

Lemma 4. *Let $R \subseteq \mathbb{Q}^n$ be ll-Horn, and suppose that there exists a definition $\Phi(x_1, \ldots, x_n)$ of R over $\mathbf{Q}$ such that in the clauses of Φ, the premises are empty (i.e. k is 0 in Definition 5). Then R is* min*-closed.*

Proof. It follows immediately from the definition that min-closed relations are closed under intersection. Thus, it suffices to prove the lemma only for relations whose ll-Horn definition has only one clause. Hence let $\Phi(x_1, \ldots, x_n)$ be a definition of R, of the form $(z > u_1 + q_1 \vee \cdots \vee z > u_l + q_l)$, and let s, t be tuples in R that we consider as functions from $\{x_1, \ldots, x_n\}$ to $\mathbb{Q}$. Let $1 \leq i, j \leq l$ be such that $s(z) > s(u_i) + q_i$ and $t(z) > t(u_j) + q_j$. Suppose that $s(z) \geq t(z)$ (the other case being similar). Then $\min(s(z), t(z)) = t(z) > t(u_j) + q_j \geq \min(s(u_j), t(u_j)) + q_j$, i.e. $\min(s, t)$ is in R.

Suppose now that R is defined by $(z > u_1 + q_1 \vee \cdots \vee z > u_l + q_l \vee z = u_1 + q_1 = \cdots = u_l + q_l)$. If s and t satisfy literals in the first part, then we prove as in the paragraph before that $\min(s, t)$ is still in R. If both s and t satisfy the chain of equalities, it is immediate that $\min(s, t)$ also satisfies it. Hence let s and t be such that s satisfies $s(z) > s(u_i) + q_i$ for some i and t satisfies the chain of equalities.

Suppose first that $s(z) \geq t(z)$, and that there is an $i \leq l$ such that $s(u_i) < t(u_i)$. Then for this i, $\min(s(z), t(z)) = t(z) = t(u_i) + q_i > \min(s(u_i), t(u_i)) + q_i$ and $\min(s, t)$ is in R. If for all $i \leq l$ we have $s(u_i) \geq t(u_i)$ then $\min(s(z), t(z)) = t(z) = t(u_i) + q_i = \min(s(u_i), t(u_i)) + q_i$ is true for all i, and $\min(s, t)$ is in R.

If $s(z) < t(z)$, then $\min(s(z), t(z)) = s(z) > s(u_i) + q_i = \min(s(u_i), t(u_i)) + q_i$, as required. □

The fact that min preserves some relations of $\mathbf{L}$ will be used in the algorithm together with the sampling method presented in the next section.

We now prove that a particular binary injective operation is in $\mathrm{Pol}(\mathbf{L})$ through the next lemmas. For this, we first have to introduce some new terminology.

Definition 7. Let $\mathbf{A}, \mathbf{B}$ be σ-structures. Let $\mathbf{A} \times \mathbf{B}$ be the σ-structure whose domain is $A \times B$ and for each symbol $R \in \sigma$ of arity n, the interpretation of R in $\mathbf{A} \times \mathbf{B}$ satisfies

$$((a_1, b_1), \ldots, (a_n, b_n)) \in R^{\mathbf{A} \times \mathbf{B}} \Leftrightarrow \mathbf{a} \in R^{\mathbf{A}} \text{ and } \mathbf{b} \in R^{\mathbf{B}}.$$

For $k \geq 1$, we write $\mathbf{A}^k$ for the k-fold product $\mathbf{A} \times \cdots \times \mathbf{A}$.

Definition 8. Let $\mathbf{A}, \mathbf{B}$ be two σ-structures. A *homomorphism* from $\mathbf{A}$ to $\mathbf{B}$ is a function $f \colon A \to B$ such that for each symbol $R \in \sigma$ and every tuple $\mathbf{a} \in R^{\mathbf{A}}$, we have $f(\mathbf{a}) \in R^{\mathbf{B}}$. An *automorphism* of $\mathbf{A}$ is a bijective function $f \colon A \to A$ such that both f and f^{-1} are homomorphisms from $\mathbf{A}$ to $\mathbf{A}$.

With these definitions, we see that polymorphisms $f \colon A^k \to A$ of $\mathbf{A}$ are simply homomorphisms from $\mathbf{A}^k$ to $\mathbf{A}$. In the next lemma, we focus on particular homomorphisms from $\mathbf{Q}^2$ to $\mathbf{Q}$. Note that the interpretation of $+_1$ in $\mathbf{Q}^2$ is $\{((a, b), (a+1, b+1)) \mid a, b \in \mathbb{Q}\}$ and that in $\mathbf{Q}^2$ we have $(a, b) < (c, d) \Leftrightarrow a < c \wedge b < d$, so that the order is not linear.

Lemma 5. *Let $<_l$ be a linear order on $\mathbb{Q}^2$ such that:*

1. *for all $a, b \in \mathbb{Q}^2$, if $a < b$ then $a <_l b$, and*
2. *for all $a, b \in \mathbb{Q}^2$ and $p \in \mathbb{Z}$, if $a <_l b$ then $a + p <_l b + p$.*

Then there exists a homomorphism $f \colon \mathbf{Q}^2 \to \mathbf{Q}$ that is so $f(a) < f(b)$ whenever $a <_l b$.

Proof. Note that the property 1 implies the following property:

3. for all $a, b \in \mathbb{Q}^2$, if $a \neq b + p$ for all $p \in \mathbb{Z}$, there exists $p \in \mathbb{Z}$ such that $b + p <_l a <_l b + p + 1$.

Enumerate $\mathbb{Q}^2$ as $(a_n)_{n \in \mathbb{N}}$. For all $n \in \mathbb{N}$, define inductively an increasing chain $(f_n)_{n \in \mathbb{N}}$ of injective functions where the domain of f_n contains $\{a_0, \ldots, a_n\}$, the range of f_n is included in $\mathbb{Q}^2$ and for all $n \in \mathbb{N}$, f_n satisfies:

a) $\forall a, b \in \{a_0, \ldots, a_n\}, b - a \in \mathbb{Z} \Rightarrow f_n(b) = f_n(a) + b - a$, and

b) $\forall a, b \in \{a_0, \ldots, a_n\}, p \in \mathbb{Z}, a <_l b + p \Rightarrow f_n(a) < f_n(b) + p$.

Let f_0 be a function that maps a_0 to any point b_0 in $\mathbb{Q}$: this clearly satisfies the properties a) and b).

Assume that f_n is defined. If $a_{n+1} = a_i + p$ for some $i \leq n$ and $p \in \mathbb{Z}$, extend f_n by $b_{n+1} = f_{n+1}(a_{n+1}) := b_i + p$. If we have $a_j = a_{n+1} + p'$, then we also have $a_j = a_i + (p + p')$ and since f_n satisfies a), $b_j = b_i + (p + p')$, i.e. $b_j = b_{n+1} + p'$. Similarly, if $a_j \neq a_{n+1} + p'$, we obtain $b_j \neq b_{n+1} + p'$, so that f_{n+1} satisfies a). If $a_{n+1} >_l a_j + p'$, then $a_i >_l a_j + p' - p$ by property 2 of $<_l$ and $b_i > b_j + p' - p$ since f_n satisfies b), which means $b_{n+1} > b_i + p'$. If $a_j >_l a_{n+1} + p'$, then $a_j >_l a_i + p + p'$ and $b_j > b_i + p + p'$, i.e. $b_j > b_{n+1} + p'$. Hence f_{n+1} satisfies b).

Otherwise $a_{n+1} \neq a_i + p$ for all $i \leq n$ and $p \in \mathbb{Z}$, and by the property 3 on $<_l$ there are $p_i \in \mathbb{Z}$ such that $a_i + p_i <_l a_{n+1} <_l a_i + p_i + 1$. Let $L = \max_i a_i + p_i$ and $U = \min_j a_j + p_j + 1$, and in the following let i and j denote respectively the index of the maximum and of the minimum. We then have $a_i + p_i <_l a_j + p_j + 1$, and $b_i + p_i < b_j + p_j + 1$ by hypothesis on f_n and property 2 of $<_l$. Since $<$ is dense in $\mathbb{Q}$, there exists $b_{n+1} \in (b_i + p_i, b_j + p_j + 1)$. Extend f by setting $f_{n+1}(a_{n+1}) = b_{n+1}$. Since $a_{n+1} - a_k \notin \mathbb{Z}$ and $b_{n+1} - b_k \notin \mathbb{Z}$ for all $k \leq n$, f_{n+1} satisfies a). If $a_{n+1} >_l a_k + p$, we have

$p \leq p_k$, hence since $b_{n+1} > b_k + p_k$, $b_{n+1} > b_k + p$. If $a_k + p >_l a_{n+1}$, we have $p \geq p_k + 1$, hence $b_k + p \geq b_k + p_k + 1 > b_{n+1}$, so that f_{n+1} satisfies b).

Put $f = \bigcup f_n$. By properties a) and b), f is a homomorphism from $(\mathbb{Q}^2, +_1, <_l)$ to $(\mathbb{Q}, +_1, <)$, and since $<_l$ is a completion of the partial order $<$, f is a homomorphism from $(\mathbb{Q}, +_1, <)^2$ to $(\mathbb{Q}, +_1, <)$, i.e. it is a polymorphism of $(\mathbb{Q}, +_1, <)$. $\square$

Let $<_{lex}$ be the lexicographic ordering on $\mathbb{Q}^2$: it is a linear expansion of the order $<$ and it satisfies the property 2 in the statement of Lemma 5. Thus, there exists a binary operation $lex\colon \mathbb{Q}^2 \to \mathbb{Q}$ such that $lex(a+1, b+1) = lex(a, b) + 1$ and $a <_{lex} b \Rightarrow lex(a) < lex(b)$. In fact, lex preserves all the relations of $\mathbf{L}$:

Lemma 6. $lex \in \mathrm{Pol}(\mathbf{L})$.

Proof. That the relations $+_1$ is preserved by lex comes from the statement of Lemma 5. That the relation defined by $(x = y \to u = v)$ is preserved is also clear, since lex is injective: if s, t are assignments that satisfy the formula $(x = y \to u = v)$, either one of s or t satisfies $x \neq y$ and by injectivity $lex(s(x), t(x)) \neq lex(s(y), t(y))$, either s and t both satisfy $u = v$ and $lex(s(u), t(u)) = lex(s(v), t(v))$.

Let s, t be assignments of $\{x, y, z\}$ that satisfy $x > y \vee x > z \vee x = y = z$. Suppose that s satisfies $x > y$ (if s satisfies $x > z$ the proof is similar). We then have that $(s(y), t(y)) <_{lex} (s(x), t(x))$ so that $lex(s, t)$ satisfies the formula $x > y$. If s satisfies $x = y = z$ and t satisfies $x > y$, we are also done as previously, noting that $(s(y), t(y)) <_{lex} (s(x), t(x))$, and similarly if t satisfies $x > z$. If s and t both satisfy $x = y = z$, then so does $lex(s, t)$.

Hence all the relations of $\mathbf{L}$ are preserved by lex. $\square$

Unfortunately, $\mathrm{Inv}(lex)$ properly contains the set of pp-definable relations over $\mathbf{L}$. Indeed, the relation $Betw = \{(a, b, c) \in \mathbb{Q}^3 \mid a < b < c \vee c < b < a\}$ is preserved by lex, but this relation is not ll-Horn, hence it is not pp-definable over $\mathbf{L}$ by Theorem 1 below. It is also worth noting that $\mathrm{CSP}(\mathbb{Q}, Betw)$ is NP-complete, hence lex does not provide information about the tractability of a CSP. The rest of this section is devoted to the description of another binary operation ll that is such that $\mathrm{Inv}(ll)$ is precisely the set of ll-Horn relations.

We first focus on the characterization of relations that are ll-Horn and definable over $(\mathbb{Q}, <)$ (i.e. definable without using $+_1$). Let $<_f$ be the linear order on $\mathbb{Q}^2$ defined by the following rules:

- if $a, c \leq 0$, then $(a, b) <_f (c, d)$ iff $(a, b) <_{lex} (c, d)$,
- if $a, c > 0$ then $(a, b) <_f (c, d)$ iff $(b, a) <_{lex} (d, c)$, and
- if $a \leq 0$ and $c > 0$ then $(a, b) <_f (c, d)$ for all $b, d \in \mathbb{Q}$.

This linear order on $\mathbb{Q}^2$ is depicted in Figure 1. Using a similar proof than that of Lemma 5, we show that there exists a binary polymorphism f of $(\mathbb{Q}, <)$ that is so that $a <_f b$ implies $f(a) < f(b)$.

Lemma 7. *Let $R \subseteq \mathbb{Q}^n$ be first-order definable over $(\mathbb{Q}, <)$. If R is preserved by f, then R is ll-Horn.*

Proof. It is known that the structure $(\mathbb{Q}, <)$ admits quantifier elimination [**?**], that is, every first-order formula is equivalent in $(\mathbb{Q}, <)$ to a formula without quantifiers. Thus, we may consider a quantifier-free definition Φ of R that is in conjunctive normal form, and let V be the set of variables of Φ. We first describe three rewriting rules that yield a formula Ψ that also defines R, and such that R is preserved by f iff Ψ is ll-Horn. If Φ, Ψ are sets of formulas, let us write $\Phi \models \Psi$ if for every assignment s of the variables from Φ and Ψ, if s satisfies all the formulas in Φ then s satisfies one of the formulas in Ψ.

a. If C is a clause of Φ of the type $x > y \vee u > v \vee C'$ and such that both

$$\Phi \setminus C \wedge \neg C' \wedge x > y \models u \geq v \vee u \geq y$$

and

$$\Phi \setminus C \wedge \neg C' \wedge u > v \models x \geq y \vee x \geq v$$

are true, replace C by

$$\begin{aligned} &(x \geq y \vee x \geq v \vee C') \wedge (x = y \rightarrow C_{x,y}) \\ \wedge &(u \geq v \vee u \geq y \vee C') \wedge (u = v \rightarrow C_{u,v}) \end{aligned}$$

where $C_{x,y}$ (resp. $C_{u,v}$) is C where the literal $x > y$ (resp. $u > v$) is removed. Let Ψ be the formula that we obtain, and denote by D_i ($1 \leq i \leq 4$) the four clauses that we added to replace C, in the order they are written above. We claim that Ψ is equivalent to Φ. Suppose that s is a valid assignment to Φ. If s satisfies C', s satisfies D_1 and D_3. Since all the literals of C' are in $C_{x,y}$ and $C_{u,v}$, s also satisfies D_2 and D_4 so that s is a satisfying assignment to Ψ. If s doesn't satisfy C', s must satisfy $x > y$ or $u > v$. Suppose that we are in the first case. The clause D_1 is true, and since $\Phi \setminus C \wedge \neg C' \wedge x > y$ entails $u \geq v \vee u \geq y$, we have that D_3 is also true. Moreover $x > y$ is a literal in $C_{u,v}$ so D_4 is true, and s satisfies $x \neq y$ so that the premise in D_2 is false, and D_2 is true. The second case is treated in exactly the same way. Conversely, if s is a valid assignment for Ψ: if s satisfies C' then C is true. If s satisfies $C_{x,y}$ or $C_{u,v}$ then C is true. Otherwise since D_2 and D_4 are true we have that $x \neq y$ and $u \neq v$. Since C' is false and $x \leq y$ and $u \leq v$, we have $x \geq v$ and $u \geq y$. This is a contradiction, since this yields $x < y \leq u < v \leq x$. Hence we have that if s satisfies Ψ, s satisfies C.

b. If C is a clause of Φ of the type $x > y \vee u > v \vee C'$ and such that $\Phi \setminus C \wedge \neg C' \wedge x > y \models u \geq v$, replace C by $(u \geq v \vee C') \wedge (u = v \rightarrow C_{u,v})$. The formula we obtain is of course equivalent to Φ.

c. Finally, suppose that C is a clause of Φ of the form $x_1 \neq y_1 \vee \cdots \vee x_k \neq y_k \vee z_0 > u_1 \vee \cdots \vee z_0 > u_l \vee u = v$. Let $C_{u,v}$ be C where $u = v$ is removed. If $\Phi \setminus C \wedge \neg C_{u,v} \wedge u = v \models z_0 = u_1 = \ldots = u_l$, replace C by $(C_{u,v} \vee (z_0 = u_1 = \cdots = u_l)) \wedge (x_1 = y_1 \wedge \cdots \wedge x_k = y_k \wedge z_0 = u_1 = \cdots = u_l \rightarrow u = v)$ and call Ψ the formula that we obtain. This new formula Ψ is equivalent to Φ: if s is a valid assignment of Φ and s satisfies a literal of $C_{u,v}$, the corresponding literal exists in Ψ and the premise of the second clause that we added is false, thus s is an assignment of Ψ. If s satisfies $u = v$ and none of the literals of $C_{u,v}$, we have by hypothesis that s satisfies $z_0 = u_1 = \cdots = u_l$, thus the two new clauses of Ψ are true. Conversely, if s is an assignment of Ψ, either s satisfies $C_{u,v}$ and Φ is true, or s satisfies all the literals in the premise of the second clause, hence s satisfies $u = v$ and is a satisfying assignment of Φ.

Let Ψ be the formula obtained after performing every possible replacement. Suppose moreover that Ψ is *reduced* (that is, any formula obtained by removing a literal or a clause is not equivalent to Ψ).

Suppose that Ψ has a bad clause C of the form $x > y \vee u > v \vee C'$ with x and u distinct variables. Since the rewriting rule a. is not applicable, we have

$$\Phi \setminus C \wedge \neg C' \wedge x > y \not\models u \geq v \vee u \geq y$$

or

$$\Phi \setminus C \wedge \neg C' \wedge u > v \not\models x \geq y \vee x \geq v$$

and suppose that we are in the first case. There exists a satisfying assignment t_1 of Φ such that t_1 satisfies $x > y$, t_1 doesn't satisfy any other literal in C and t_1 satisfies $u < y$. Furthermore since the rewriting rule b. is not applicable, we have that $\Phi \setminus C \wedge \neg C' \wedge u > v + q_2 \not\models x \geq y + q_1$, so that there exists t_2 that satisfies Φ, $u > v + q_2$ and $x < y + q_1$ and that doesn't satisfy any other literal in C. Finally, there exists an automorphism γ of $(\mathbb{Q}, <)$ such that $f(\gamma t_1, t_2)$ doesn't satisfy any literal in C. This automorphism sends $t_1(u)$ to 0 and $t_1(x), t_1(y), t_1(v)$ to positive values. Hence, the relation is not preserved by f, a contradiction. The second case is treated similarly.

If Ψ has a clause which contains $x = y \vee u = v$, we immediately see that Ψ isn't preserved by f, using the fact that Ψ is reduced and that f is injective.

We have established that if a clause of Ψ contains $x > y$ and $u > v$, we have $x = u$, and that a clause contains at most one literal $x = y$. Suppose now that Ψ has a clause C which contains $u = v$. This clause is of the form $x_1 \neq y_1 \vee \cdots \vee x_k \neq y_k \vee z_0 > u_1 \vee \cdots \vee z_0 > u_l \vee u = v$ by what we have proved so far. Since the rewriting rule c. is not applicable, we have $\Psi \setminus C \wedge u = v \wedge \neg C_{u,v} \not\models z_0 = u_1 = \cdots = u_l$, hence there exists s a valid assignment of Ψ such that $s(u) = s(v)$, $s(z_0) < s(u_j)$ for some $1 \leq j \leq l$ and $s(x_i) = s(y_i)$ for all $1 \leq i \leq k$. Since Ψ is reduced, there are $(t_i)_{i \leq l}$ that are valid assignments of Ψ and such that $t_i(z_0) > t_i(u_i)$ for $i \leq l$, and such that all the other literals of C are false. But then there exists an automorphism α of $(\mathbb{Q}, <)$ such that $f(\alpha t_j, s)$ doesn't valid any literal in C, i.e. R is not f-closed, a contradiction. $\square$

The link between ll-Horn relations that are definable over $\mathbf{Q}$ and those that are definable over $(\mathbb{Q}, <)$ is detailed in the next lemma:

Lemma 8. *Let $R \subseteq \mathbb{Q}^n$ be a relation that is first-order definable over $\mathbf{Q}$. There exists $T \subseteq \mathbb{Q}^n$ that is first-order definable over $(\mathbb{Q}, <)$, pp-definable over $(\mathbb{Q}, R, +_1)$, and that is not ll-Horn.*

Proof. Let $\Phi(\mathbf{x})$ be a definition of R over $\mathbf{Q}$. There exist two formulas $\Phi_<(\mathbf{x}, \mathbf{y})$ and $\Phi_{+_1}(\mathbf{x}, \mathbf{y})$ in respectively the signature $\{<\}$ and the signature $\{+_1\}$ such that

$$\mathbf{Q} \models \forall \mathbf{x} (\Phi(\mathbf{x}) \Leftrightarrow \exists \mathbf{y}. \Phi_<(\mathbf{x}, \mathbf{y}) \wedge \Phi_{+_1}(\mathbf{x}, \mathbf{y})).$$

These formulas are built in the following manner: for each $x_j = x_i + p_i$ appearing in Φ, replace $x_i + p_i$ by a new variable $y_i^{p_i}$ and put in Φ_{+_1} the atom $y_i^{p_i} = x_i + p_i$. The equivalence displayed above is easily seen to hold, and Φ_{+_1} is a primitive positive formula in the signature $\{+_1\}$.

Note that the formula $(\exists \mathbf{y}. \Phi_<(\mathbf{x}, \mathbf{y}) \wedge \Phi_{+_1}(\mathbf{x}, \mathbf{y}))) \wedge \Phi_{+_1}(\mathbf{x}, \mathbf{z})$ is equivalent (in $\mathbf{Q}$) to $\Phi_<(\mathbf{x}, \mathbf{z})$, so that the relation T defined by $\Phi_<(\mathbf{x}, \mathbf{z})$ over $(\mathbb{Q}, <)$ is pp-definable over $(\mathbb{Q}, R, +_1)$.

It remains to show that T is not ll-Horn if R is not. Suppose that T has a ll-Horn definition $\Psi(\mathbf{x}, \mathbf{z})$. Since $\Phi_{+_1}(\mathbf{x}, \mathbf{z})$ is ll-Horn, we have that $\Phi(\mathbf{x})$ is equivalent to a ll-Horn formula, namely the formula $\exists \mathbf{y}. \Psi(\mathbf{x}, \mathbf{y}) \wedge \Phi_{+_1}(\mathbf{x}, \mathbf{y})$. As a consequence, R is ll-Horn, a contradiction. $\square$

Let S be a dense codense subset of $\mathbb{Q}$, that is closed under integer translations: if $a \in S$, then $a + p \in S$ for all $p \in \mathbb{Z}$. Let $<_{ll}$ be the linear extension of $<$ on $\mathbb{Q}^2$ defined as follows:

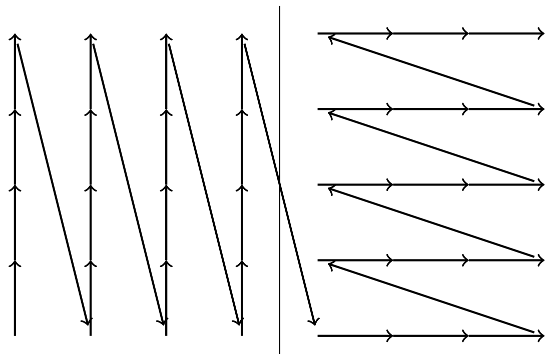

Figure 1: The order $<_f$

- if $a \in S$, then $(a,b) <_{ll} (c,a)$ for all $b,c > a$,
- if $a \notin S$, then $(b,a) <_{ll} (a,c)$ for all $b,c > a$, and
- $(a,a) <_{ll} (a,b)$ and $(a,a) <_{ll} (b,a)$ for all $b > a$.

By Lemma 5, there exists a binary polymorphism ll of $\mathbf{Q}$ that is so that if $a <_{ll} b$, then $ll(a) < ll(b)$.

Lemma 9. *Let $\mathbf{A}$ be a structure. Let $f \colon A^k \to A$, and let $R \subseteq A^n$ be first-order definable over $\mathbf{A}$. Suppose that for any finite subset A' of A^k, $f|_{A'}$ can be expressed as a composition of polymorphisms of $(\mathbf{A}, R)$. Then f is a polymorphism of $(\mathbf{A}, R)$.*

Proof. Let $\mathbf{a}_1, \ldots, \mathbf{a}_k$ be tuples from the relation R. Consider the finite subset A' of A^k that consists of the elements $(a_1^1, \ldots, a_k^1), \ldots, (a_1^n, \ldots, a_k^n)$. By assumption, on A' the function f can be expressed as a composition of polymorphisms of $(\mathbf{A}, R)$. Since these polymorphisms preserve R, f also preserves R. □

Lemma 10. *Let $R \subseteq \mathbb{Q}^n$ be first-order definable over $(\mathbb{Q}, <)$. If R is preserved by ll, R is preserved by f.*

Proof. It suffices to prove that on every finite subset A of $\mathbb{Q}^2$, f can be expressed as the composition of ll and polymorphisms of $(\mathbb{Q}, <)$, as per Lemma 9.

Let $A = \{(a_1, b_1), \ldots, (a_k, b_k)\}$ be such a subset. Suppose that $a_i \leq 0$ for $i \leq l$ and $a_i > 0$ otherwise. Since S is dense in $\mathbb{Q}$, there are points c_i $(i \leq l)$ arbitrarily close to each a_i so that $c_i \in S$. Similarly, since $\mathbb{Q} \setminus S$ is dense in $\mathbb{Q}$, there are points c_i $(l < i \leq k)$ arbitrarily close to each a_i so that $c_i \notin S$. Choosing the points c_i sufficiently close to the a_i, the partial function that maps a_i to c_i can be extended to an automorphism β of $(\mathbb{Q}, <)$. It suffices now to note that we have $ll(\beta(a_i), b_i) < ll(\beta(a_j), b_j)$ if, and only if, $(a_i, b_i) <_f (a_j, b_j)$, so that there exists an automorphism α such that $(\alpha \circ ll)(\beta(a_i), b_i) = f(a_i, b_i)$, which concludes the proof. □

The following theorem is the conclusion of Sections 2 and 3.

Theorem 1. *Let $R \subseteq \mathbb{Q}^n$ be a relation that is first-order definable over $\mathbf{Q}$. The following are equivalent:*

1. *R is ll-Horn,*

2. *R is pp-definable over* $\mathbf{L}$*,*

3. *R is preserved by* ll*.*

Proof. (1) $\Rightarrow$ (2). This is proved in Lemma 2.
(2) $\Rightarrow$ (3). It suffices to show that ll preserves the three relations of $\mathbf{L}$, by Lemma 3. The relation $+_1$ is preserved by construction of ll, as stated in Lemma 5. The relation defined by $(x = y \to u = v)$ is preserved since ll is injective. Finally, let s, t be valid assignments of the formula $(x > y \vee x > z \vee x = y = z)$. If both s and t satisfy the same literal, we are done since ll preserves $<$. Otherwise, one of $(s(y), t(y))$ and $(s(z), t(z))$ is less than $(s(x), t(x))$ with respect to the $<_{ll}$ order, so that $ll(s,t)$ indeed satisfies $(x > y \vee x > z \vee x = y = z)$.
(3) $\Rightarrow$ (1). By contraposition, suppose that R is not ll-Horn. By Lemma 8, there exists a non-ll-Horn relation R' that is pp-definable over $(\mathbb{Q}, R, +_1)$ and first-order definable over $(\mathbb{Q}, <)$. By Lemma 7, R' is not preserved by f. By Lemma 10, we have that R' is not preserved by ll, and finally by Lemma 3, R is not preserved by ll, which concludes the proof. $\square$

4. Sampling

Definition 9. Let $\mathbf{A}$ be a structure. We say that an algorithm *samples* from $\mathbf{A}$, if given $n \in \mathbb{N}$ it returns a substructure $\mathbf{B}$ of $\mathbf{A}$ so that for every pp-sentence Φ with at most n quantified variable, we have $\mathbf{A} \models \Phi \Leftrightarrow \mathbf{B} \models \Phi$. In other words, CSP($\mathbf{A}$) and CSP($\mathbf{B}$) are the same problems when restricted to inputs with at most n variables.

In this section, we prove the following:

Theorem 2. *Let* $\mathbf{A}$ *be a first-order reduct of* $\mathbf{Q}$ *with a finite signature. There is a polynomial-time sampling algorithm for* $\mathbf{A}$*.*

Let $+_n$ denote the relation $\{(a, a+n) : a \in \mathbb{Q}\} \subset \mathbb{Q}^2$ and $>_n$ denote the relation $\{(a,b) : a > b+n\} \subset \mathbb{Q}^2$. We begin by giving an algorithm that samples from the structure

$$\mathbf{Q}_m = (\mathbb{Q}, +_0, \ldots, +_m, >_{-m}, \ldots, >_m)$$

and then we prove that the structure $\mathbf{Q}_\infty = \bigcup_{m \geq 0} \mathbf{Q}_m$ admits *positive quantifier-elimination*, i.e. that for every first-order formula $\Phi(\overline{x})$, there exists a positive quantifier-free $\Psi(\overline{x})$ that is equivalent to it. Using this we finally provide a sampling algorithm for any first-order reduct $\mathbf{A}$ of $\mathbf{Q}$.

The algorithm Sample-$\mathbf{Q}_m$ takes as input $n \in \mathbb{N}$ and outputs the substructure $\mathbf{B}$ of $\mathbf{Q}_m$ induced by $B = \{\frac{i}{n} : 0 \leq i \leq n^2(m+1)\}$, that is, the structure with domain B and where there is a relation between two points of B if and only if this relation exists in $\mathbf{Q}_m$.

Lemma 11. *Sample-*$\mathbf{Q}_m$ *is an efficient sampling algorithm for* $\mathbf{Q}_m$*.*

Proof. It is clear that the algorithm runs in time a polynomial in n. Moreover, since $\mathbf{B}$ is induced in $\mathbf{Q}_m$, it is clear that if $\mathbf{B} \models \Phi$ then $\mathbf{Q}_m \models \Phi$, since an assignment of the variables of Φ to B can be seen as an assignment of the variables to $\mathbb{Q}$.

Suppose that $\mathbf{Q}_m \models \Phi$, and let $\Phi = \exists x_1, \ldots, x_n.\varphi(\mathbf{x})$. By assumption, there is a valid assignment $s\colon \{x_1, \ldots, x_n\} \to \mathbb{Q}$ that satisfies φ. Let $m = \min s(x)$. If $m \neq 0$, we may replace s by the assignment that maps x to $s(x) - m$ so that the minimum of the new assignment is 0. We order the variables $x_1, \dots, x_n$ as follows: if $s(x) - \lfloor s(x) \rfloor < s(y) - \lfloor s(y) \rfloor$, we pose $x < y$. This order defines an equivalence relation $x \equiv y$ that is such that $x \equiv y \Leftrightarrow s(x) - s(y) \in \mathbb{Z}$. Let us denote by $\overline{x}$ the $\equiv$-equivalence class of x. Finally, let $y_1, \dots, y_r$ enumerate the representatives of each equivalence class in such a way that $y_i < y_j$ when $i < j$ and each y_i is so that $s(y_i) \leq s(z)$ for $z \in \overline{y_i}$. Let $t \colon \{x_1, \dots, x_n\} \to \mathbb{Q}$ be defined for $z \in \overline{y_i}$ by $t(z) = \lfloor s(z) \rfloor + \frac{i}{n}$. We claim that t is a valid assignment of φ.

- if x, y are so that $\overline{x} = \overline{y}$, then $s(x) - s(y) = t(x) - t(y)$, so that the relations $+_n$ and $<_n$ are preserved;
- if $x \in \overline{y_i}$ and $y \in \overline{y_j}$ for $i \neq j$, we have that $s(x) - s(y) \notin \mathbb{Z}$, and $t(x) - t(y) \notin \mathbb{Z}$, so that the relation $+_n$ is preserved for all $n \leq m$;
- finally, suppose that $x \in \overline{y_i}$ and $y \in \overline{y_j}$ for $i < j$. If the constraint $y >_k x$ is in φ for $k \in \mathbb{Z}$, we have $t(y) = \lfloor s(y) \rfloor + \frac{j}{n} > \lfloor s(x) + k \rfloor + \frac{j}{n} > \lfloor s(x) \rfloor + k + \frac{i}{n} = t(x) + k$, so that the constraint $y >_k x$ is still satisfied by t. If the constraint $x >_k y$ is in φ, we have $s(x) \geq \lfloor s(y) \rfloor + k + 1$: indeed, if $\lfloor s(y) \rfloor + k < s(x) < \lfloor s(y) \rfloor + k + 1$, $\lfloor s(x) \rfloor = \lfloor s(y) \rfloor + k$ and $s(x) - \lfloor s(x) \rfloor > s(y) + k - \lfloor s(y) \rfloor - k = s(y) - \lfloor s(y) \rfloor$, a contradiction with $i < j$. Now it follows that $t(x) = \lfloor s(x) \rfloor + \dfrac{i}{n} \geq \lfloor s(y) \rfloor + k + 1 + \dfrac{i}{n} > \lfloor s(y) \rfloor + \dfrac{j}{n} + k = t(y) + k$, so that $x >_k y$ is still satisfied by t.

The image of t is included in the subset $\{\frac{i}{n} : i \in \mathbb{N}\} \subset \mathbb{Q}$, but it may go beyond $n(m+1)$. Suppose it does: then there exist x, y so that $t(y) - t(x) > m + 1$ and such that for all the variables z, either $t(z) \leq t(x)$ or $t(z) \geq t(y)$ (i.e. x and y are "consecutive" in the order induced by t). Let $X = \{z : t(z) \leq t(x)\}$ and $Y = \{z : t(z) \geq t(y)\}$, and define a new t' as follows. If $z \in X$, set $t'(z) = t(z)$, and if $z \in Y$, set $t'(z) = t(z) - t(y) + t(x) + m + 1$. This t' is still a valid assignment to φ: since the distance between variables of X or between variables of Y have not changed, the clauses in φ that involve these pairs of variables are still satisfied. Moreover, φ cannot have a clause $z_1 = z_2 + n$ (with $n \leq m$) where $z_1 \in X$ and $z_2 \in Y$, since we have $s(z_2) - s(z_1) > s(y) - s(x) > m + 1 > n$, so that s could not satisfy such a clause. Hence, all the clauses that can relate z_1 and z_2 in φ are of the form $z_1 > z_2 + n$ and $z_2 > z_1 + n$. In the former case, we have $t'(z_1) = t(z_1) > t(z_2) + n > t'(z_2) + n$, and in the latter case, we have $t'(z_2) = t(z_2) - t(y) + t(x) + m + 1 \geq t(y) - t(y) + t(x) + m + 1 = t(x) + m + 1 \geq t'(z_1) + m + 1 > t'(z_1) + n$ since $n \leq m$.

Finally, note that t' has one less pair (x, y) of consecutive variables (as defined above) that are so that $t'(y) - t'(x) > m + 1$. Repeating this process, we find an assignment such that all consecutive variables are at distance $\leq m + 1$, so that the image of this assignment is bounded above by $n(m+1)$. □

Lemma 12. $\mathbf{Q}_\infty$ *has positive quantifier elimination, that is, for every first-order formula* $\Phi(\mathbf{x})$ *over the language of* $\mathbf{Q}_\infty$*, there exists a quantifier-free formula* $\Psi(\mathbf{x})$*, in which negation does not appear and so that*

$$\mathbf{Q}_\infty \models \forall \mathbf{x} (\Phi(\mathbf{x}) \Leftrightarrow \Psi(\mathbf{x})).$$

Proof. We prove that for any two ω-saturated models $\mathbf{A}, \mathbf{B}$ of $T = \mathrm{Th}(\mathbf{Q}_\infty)$ that have a common substructure $\mathbf{C}$, the set $\mathcal{I}$ of all partial isomorphisms with finite domain between $\mathbf{A}$ and $\mathbf{B}$ have the back-and-forth property. The fact that this implies quantifier elimination is a corollary of Theorem 3.2.5 in [**?**]:

Theorem 3. *For a theory T the following are equivalent:*

1. *T has quantifier elimination,*

2. *For all models $\mathbf{A}, \mathbf{B}$ of T with a common substructure $\mathbf{C}$ we have $\mathbf{A}_C \equiv \mathbf{B}_C$.*

If $\mathbf{A}, \mathbf{B}$ are as in the theorem, we let $\mathbf{A}'$ (resp. $\mathbf{B}'$) be an ω-saturated elementary extension of $\mathbf{A}$ (resp. $\mathbf{B}$). If the set $\mathcal{I}$ of all partial isomorphisms with finite domain between $\mathbf{A}'$ and $\mathbf{B}'$ has the back-and-forth property, partial isomorphisms are elementary. In particular, we'd have that $\mathrm{id}_{\mathbf{C}}$ is elementary between $\mathbf{A}'$ and $\mathbf{B}'$, hence it is also elementary between $\mathbf{A}$ and $\mathbf{B}$, so that $\mathbf{A}_C \equiv \mathbf{B}_C$.

That $\mathcal{I}$ is non-empty is clear, since $\mathrm{id}_{\mathbf{C}}$ is a partial isomorphism.

Let $f\colon \{a_1, \ldots, a_n\} \to \{b_1, \ldots, b_n\} \in \mathcal{I}$ and let $a_{n+1} \in \mathbf{A}$. If there exists $i \leq n$ such that $a_{n+1} = a_i + k$ for $k \in \mathbb{Z}$, take $b_{n+1} = b_i + k$ and extend f by $f(a_{n+1}) = b_{n+1}$. This is still a partial isomorphism: it preserves $+_n$ since if $\mathbf{A} \models a_j = a_{n+1} + k'$, we have $\mathbf{A} \models a_j = a_i + k + k'$ hence $\mathbf{B} \models b_j = b_i + k + k'$ and $\mathbf{B} \models b_j = b_{n+1} + k'$. If $a_{n+1} > a_j + k'$, $\mathbf{A} \models a_i > a_j + k' - k$, hence $\mathbf{B} \models b_i > b_j + k' - k$ and $b_{n+1} > b_j + k'$. The case $a_{n+1} < a_j + k'$ is treated similarly.

Otherwise, let τ be $\bigcup_{\mathbf{A} \models a_{n+1} > a_i + k} \{x > b_i + k\} \cup \bigcup_{\mathbf{A} \models a_{n+1} < a_i - k} \{x < b_i - k\}$. Every finite subset of τ is realized in $\mathbf{B}$: if ρ is a finite subset of τ, for each $i \leq n$ there are l_i, u_i maximal such that the formulas $x > b_i + l_i$ and $x < b_i - u_i$ are in ρ. Letting $L = \max b_i + l$ and $U = \min b_i - u$, we know that the interval (L, U) in $\mathbf{B}$ is not empty, for if it were, we would have $b_i + l_i \geq b_j - u_j$, hence $a_i + l_i \geq a_j - u_j$, which is a contradiction. Hence there is an element in (L, U) and ρ is satisfiable. As a consequence, τ is a partial type over a finite number of parameters, hence it is realized in $\mathbf{B}$ by saturation. Let b_{n+1} in $\mathbf{B}$ be an element that realizes τ, and we can extend f by setting $f(a_{n+1}) = b_{n+1}$.

Checking that $\mathcal{I}$ has the "back" property is done in a similar way.

We conclude that $\mathbf{Q}_m$ has quantifier elimination. To prove that the quantifier elimination process indeed yields positive formulas, simply note that the negation of an atom $y = x + k$ is equivalent modulo T to $y > x + k \vee x > y - k$, which is in the language and that the negation of $y > x + k$ is equivalent modulo T to $x > y - k \vee y = x + k$. Every quantifier-free formula is then equivalent to a positive quantifier-free formula, which concludes the proof. $\square$

We finally provide the desired sampling algorithm for reducts $\mathbf{A}$ of $(\mathbb{Q}, +_1, <)$ that have a finite signature. Since $\mathbf{A}$ is first-order definable over $(\mathbb{Q}, +_1, <)$, it is positively quantifier-free definable over $\mathbf{Q}_\infty$ by Lemma 12, and since $\mathbf{A}$ has finite signature there exists $m \in \mathbb{N}$ such that $\mathbf{A}$ is positively quantifier-free definable over $\mathbf{Q}_m$. The sampling algorithm for $\mathbf{A}$ simply samples $\mathbf{B}'$ from $\mathbf{Q}_m$, and returns $\mathbf{B}$, the structure with the same signature as $\mathbf{A}$ where relations are built according to their positive quantifier-free definition on $\mathbf{B}'$. That this is efficient is clear, since the running time of the sampling algorithm of $\mathbf{Q}_m$ is polynomial in n, and the time taken to build $\mathbf{B}$ is $\Theta(|B'|^k)$, where k is the maximal arity of relations of $\mathbf{A}$.

Lemma 13. *Let Φ be an instance of* $\mathrm{CSP}(\mathbf{A})$ *with n variables. Then $\mathbf{A} \models \Phi \Leftrightarrow \mathbf{B} \models \Phi$.*

Proof. Let Ψ be the formula obtained by replacing relation symbols in Φ by their positive quantifier-free definition over $\mathbf{Q}_m$. Ψ is equivalent to a disjunction of primitive positive formula over the language of $\mathbf{Q}_m$, and it has n variables. Hence, $\mathbf{Q}_m \models \Psi \Leftrightarrow \mathbf{B}' \models \Psi$ by Lemma 11. Now by construction, we have $\mathbf{A} \models \Phi \Leftrightarrow \mathbf{Q}_m \models \Psi$ and $\mathbf{B} \models \Phi \Leftrightarrow \mathbf{B}' \models \Psi$, which proves the lemma. $\square$

Remark 2. Using the same techniques, one can also sample from finite-signature reducts of $(\mathbb{Z}, \text{succ}, <)$. It is even easier (from a complexity point of view) to do so: the sample here is the substructure of $(\mathbb{Z}, \text{succ}, <)$ induced by the set $\{0, \ldots, n(m+1)\}$, which is n times smaller than the sample for $(\mathbb{Q}, \text{succ}, <)$.

5. Algorithm

Let $\mathbf{A}$ be the structure over $\mathbb{Q}$ with the three relations defined by $y = x + 1$, $x > y$, and $x > y \vee x > z \vee (x = y = z)$. Our algorithm for $\text{CSP}(\mathbf{L})$ relies on two points: $\mathbf{L}$ has an injective binary polymorphism, and there is a polynomial-time algorithm for $\text{CSP}(\mathbf{A})$. While the first point is proved in Lemma 6, the second item is a consequence of a theorem of [**?**] and of Lemmas 4 and 13:

Theorem 4 (Theorem 2.4 in [**?**])**.** *Let* $\mathbf{A}$ *be a structure over a finite relational signature with a semi-lattice polymorphism. If there exists an efficient sampling algorithm for* $\mathbf{A}$*, then* $\text{CSP}(\mathbf{A})$ *is in* P.

A semi-lattice operation is a binary operation f that is:

- idempotent, i.e. $\forall x, f(x, x) = x$,
- associative, i.e. $\forall x, y, z, f(x, f(y, z)) = f(f(x, y), z)$, and
- commutative, i.e. $\forall x, y, f(x, y) = f(y, x)$.

It is straightfoward to verify that the binary function $\min\colon \mathbb{Q}^2 \to \mathbb{Q}$ is indeed a semi-lattice operation. As per Lemma 4, min is a polymorphism of the structure $\mathbf{A}$ described above, since $\mathbf{A}$ consists of ll-Horn relations that are definable using clauses that have no premises, and Lemma 13 asserts that there is an efficient sampling algorithm for $\mathbf{A}$. As a consequence, there exists a polynomial-time algorithm for $\text{CSP}(\mathbf{A})$, which we denote by $\mathsf{Solve}\,\text{CSP}(\mathbf{A})$.

For a finite set of formulas ψ, let us denote by $\bigwedge \psi$ the formula obtained by taking the conjunction of all the formulas in ψ. Our algorithm for $\text{CSP}(\mathbf{L})$ is presented below.

It is obvious that the algorithm terminates: at each execution of the *repeat* loop except the last one, the size of Φ strictly decreases. In the worst case, the algorithm complexity is $O(|\Phi|^2 T(|\Phi|))$, where $T(n)$ is the worst-case complexity of the algorithm $\mathsf{Solve}\,\text{CSP}(\mathbf{A})$. We now prove that $\mathsf{Solve}\,\text{CSP}(\mathbf{L})$ is sound and complete for $\text{CSP}(\mathbf{L})$.

Lemma 14. *If* $\mathsf{Solve}(\Phi)$ *rejects,* $\mathbf{L} \not\models \Phi$.

Proof. Let m be the number of iterations of the *repeat* loop, and let us denote by Ψ_i the set Ψ at the beginning of the ith iteration. We prove by induction that for all $1 \leq i \leq m$, $\mathbf{L} \models \Phi \Rightarrow \exists \mathbf{x}. \bigwedge \Psi_i$.

Since Ψ_1 is a subset of the initial set Φ, we obviously have $\mathbf{L} \models \Phi \Rightarrow \exists \mathbf{x}. \bigwedge \Psi_1$. For each $i < m$, the set Ψ_{i+1} is obtained from Ψ_i by renaming variables. Let C be a clause of Ψ_{i+1}. If this clause is already in Ψ_i, there is nothing to prove. Otherwise there is in Ψ_i a clause C', where C and C' differ by a replacement of variables, say that v has been replaced by u and that this is the only replacement that occurred, for the sake of conciseness —the proof generalizes when several replacements occurred. Since v has been renamed to u, there is in Φ_i a clause of the form $x = y \to u = v$, such that both $\mathbf{A} \not\models \exists \mathbf{x}. \bigwedge \Psi_i \wedge x > y$ and $\mathbf{A} \not\models \exists \mathbf{x}. \bigwedge \Psi_i \wedge y > x$, but since the algorithm hasn't rejected at this point, we also have $\mathbf{A} \models \exists \mathbf{x}. \bigwedge \Psi_i$. Those three statements

Algorithm: $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{L})$

Data: a pp-sentence $\Phi = \exists \mathbf{x}.\varphi$ in the language of $\mathbf{L}$

Result: *accepts* if $\mathbf{L} \models \Phi$, *rejects* otherwise

$\Psi := \emptyset$;

repeat

  $\Psi := \{C \in \Phi \mid C \text{ is in the language of } \mathbf{A}\}$;

  if $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\exists \mathbf{x}. \bigwedge \Psi)$ *rejects* **then**

    reject;

  end

  forall *clauses of* Φ *of the type* $x = y \to u = v$ **do**

    if $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\exists \mathbf{x}. \bigwedge \Psi \wedge y > x)$ *rejects and* $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\exists \mathbf{x}. \bigwedge \Psi \wedge x > y)$ *rejects* **then**

      Delete the clause $x = y \to u = v$ from Φ;

      Replace all occurrences of v in Φ by u;

    end

  end

until Φ *doesn't change*;

accept;

together imply that $\Psi_i \models x = y$, and since $\mathbf{L} \models \Phi \Rightarrow \exists \mathbf{x}. \bigwedge \Psi_i$ by the induction hypothesis and Φ contains $x = y \to x = u$, we have that $\mathbf{L} \models \Phi \Rightarrow \exists \mathbf{x}. \bigwedge \Psi_i \wedge u = v$. It remains to note that $\bigwedge \Psi_i \wedge u = v$ is equivalent to $\bigwedge(\Psi_i \setminus \{C'\} \cup \{C\}) \wedge u = v$, so that Φ indeed implies C and $\mathbf{L} \models \Phi \to \exists \mathbf{x}. \bigwedge \Psi_{i+1}$.

Since $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\Psi_i)$ fails at some point, it must be that $\mathbf{A} \not\models \exists \mathbf{x}. \bigwedge \Psi_i$ so $\mathbf{L} \not\models \exists \mathbf{x}. \bigwedge \Psi_i$ and $\mathbf{L} \not\models \Phi$, as required. □

Lemma 15. *If* $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{L})(\Phi)$ *accepts,* $\mathbf{L} \models \Phi$.

Proof. Let Φ' be the resulting formula after the *repeat* loop, Ψ' be the set of clauses of Φ' that are in the language of $\mathbf{A}$, $V(\Phi')$ be the set of variables of Φ', and $V(\Psi')$ be the set of variables of Ψ'. Note that we have $\Phi' \Rightarrow \Phi$ (the proof is the same as in the previous lemma, noting that $\mathbf{L} \models \Phi_i \Rightarrow \Phi$ for all the executions i of the *repeat* loop). We have that $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\Psi')$ accepts, so that $\mathbf{A} \models \exists \mathbf{x}. \bigwedge \Psi'$. Let $x = y \to u = v$ be a clause not in Ψ'. Since this clause is still present in Φ', it must be that either $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\exists \mathbf{x}. \bigwedge \Psi' \wedge y > x)$ or $\mathsf{Solve}\,\mathrm{CSP}(\mathbf{A})(\exists \mathbf{x}. \bigwedge \Psi' \wedge x > y)$ accepts, so that there exists a valid assignment s to the variables of $\bigwedge \Psi'$ where $s(x) \neq s(y)$.

Let us enumerate all the clauses of the form $x_i = y_i \to u_i = v_i$ $(0 \leq i \leq r)$ and for each of them, let s_i be an assignment such that $s_i(x_i) \neq s_i(y_i)$, which exists by the argument above. By Lemma 6, $\mathbf{L}$ has *lex* as a polymorphism, so that the assignment $t = lex(s_1, lex(s_2, \ldots, lex(s_{r-1}, s_r)) \ldots)$ is a valid assignment of Ψ'. Moreover, since *lex* is injective t simultaneously breaks all the equalities in the premise of the constraints that don't belong to Ψ', hence those constraints are satisfied. If some variables of Φ are not in the domain of t (that is, if $V(\Psi') \subsetneq V(\Phi')$), we may assign them to values that will break the equalities in the premises of the clauses, since the domain is infinite. The assignment that we obtained is a solution to Φ', hence to Φ, so that $\mathbf{L} \models \Phi$. □

6. Maximality

We prove in this section that CSP($\mathbf{L}$) is maximally tractable. In order to do this, we again use Lemma 8 together with the maximality results from [**?**]:

Theorem 5. *Let $\mathbf{L}^-$ be the structure over the rational numbers whose relations are defined by $x \neq y$, $x > y \vee x > z \vee (x = y = z)$, and $(x = y \rightarrow u = v)$. Then $\mathbf{L}^-$ is maximally tractable within the class of reducts of $(\mathbb{Q}, <)$. Moreover, the reduct $\mathbf{M}^-$ that consists of* min*-closed relations fo-definable over $(\mathbb{Q}, <)$ is maximally tractable within the same class.*

The proofs of maximality in our case is straightforward:

Corollary 1. *The structure $\mathbf{L}$ is maximal within the class of reducts of $(\mathbb{Q}, <, +_1)$.*

Proof. Let $\mathbf{A}$ be such that $\mathbf{L} < \mathbf{A}$, i.e. every relation of $\mathbf{L}$ is pp-definable in $\mathbf{A}$ and $\mathbf{A}$ contains a relation R that is not pp-definable over $\mathbf{L}$. By Lemma 1, we may assume that $\mathbf{A}$ already contains the relations of $\mathbf{L}$, since this doesn't change the complexity of its CSP. Since R is not pp-definable over $\mathbf{L}$, it is by Lemma 2 not ll-Horn, so that by Lemma 8 there exists a non-ll-Horn relation R' that is pp-definable over $\mathbf{A}$ and first-order definable over $(\mathbb{Q}, <)$. Let us consider the structure $\mathbf{B} = (\mathbf{L}^-, R')$. Since R' is not ll-Horn, we have $\mathbf{L}^- < \mathbf{B}$, and $\mathbf{B}$ is a first-order reduct of $(\mathbb{Q}, <)$. By Theorem 5, CSP($\mathbf{B}$) is NP-hard. Moreover, since all the relations of $\mathbf{B}$ are pp-definable over $\mathbf{A}$, we have $\mathbf{B} \leq \mathbf{A}$, thus CSP($\mathbf{A}$) is NP-hard, which concludes the proof. □

Corollary 2. *The structure $\mathbf{M}$ that consists of* min*-closed relations fo-definable over $(\mathbb{Q}, <, +_1)$ is maximally tractable within the class of reducts of $(\mathbb{Q}, <, +_1)$.*

Proof. Let $\mathbf{A}$ be a reduct of $(\mathbb{Q}, <, +_1)$ such that $\mathbf{M} < \mathbf{A}$, and let R be a relation of $\mathbf{A}$ that is not min-closed. Looking at the proof of Lemma 8, we see that the relation R' that we build is so that $(\mathbb{Q}, R, +_1)$ and $(\mathbb{Q}, R', +_1)$ are pp-interdefinable, i.e., that $(\mathbb{Q}, R, +_1)$ is pp-definable over $(\mathbb{Q}, R', +_1)$ and vice-versa. As a consequence, any operation that preserves $+_1$ and R' preserves R, and since R is not min-closed and min preserves $+_1$, we have that R' is not min-closed either. Let $\mathbf{B}$ be the structure $(\mathbf{M}^-, R')$. We have that $\mathbf{B} < \mathbf{A}$. By Theorem 5, $\mathbf{B}$ is not tractable, in the sense that there exists a $\mathbf{B}'$, obtained from $\mathbf{B}$ by keeping finitely many relations, so that CSP($\mathbf{B}'$) is NP-hard. Since we also have $\mathbf{B}' < \mathbf{A}$, we have that $\mathbf{A}$ is not tractable and that $\mathbf{M}$ is maximally tractable. □

7. Conclusion

A way to rephrase the results presented here is as follows:

Corollary 3. *Let $\mathbf{A}$ be a first-order reduct of $(\mathbb{Q}, +_1, <)$ with a finite signature. If* min *or ll is a polymorphism of $\mathbf{A}$, then* CSP($\mathbf{A}$) *is in* P. *Moreover, if $\mathbf{L} < \mathbf{A}$ or $\mathbf{M} < \mathbf{A}$ then* CSP($\mathbf{A}$) *is* NP*-hard.*

The statement of Corollary 3 is visually represented in Figure 2. For all the first-order formula Φ defining a relation R over $(\mathbb{Q}, <, +_1)$, define the dual of R as the relation defined by Φ, where all the symbols $<$ are replaced by $>$. For an operation $f \colon \mathbb{Q}^2 \to \mathbb{Q}$, define the dual of f by $\overline{f} \colon (x, y) \mapsto -f(-x, -y)$. It is not hard to prove that an operation f preserves a relation R if, and only if, $\overline{f}$ preserves the dual relation of R. Using this fact, we obtain two more maximally tractable

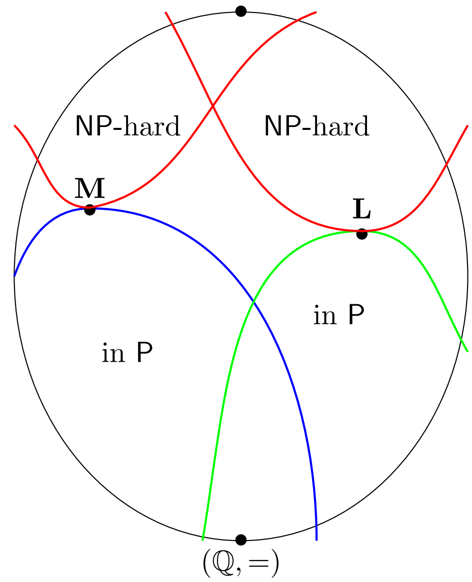

Figure 2: Visual summary of the results.

reducts, namely the structure that consists of max-closed relations and the structures that consists of $\overline{ll}$-closed relations.

The statement of Corollary 3 naturally leads to the following question: $\mathbf{A}$ being a reduct of $(\mathbb{Q}, +_1, <)$, for which operations $g\colon \mathbb{Q}^n \to \mathbb{Q}$ do we have that CSP($\mathbf{A}$) is polynomial-time tractable whenever g is a polymorphism of $\mathbf{A}$? This question has been answered for the class of reducts of $(\mathbb{Q}, <)$ and of $(\mathbb{V}, E)$, the Rado graph. Moreover, in the two previous cases one may even come up with a finite list of polymorphisms. Thus, the second question one may ask is: does there exist a finite family $\mathcal{F}$ of operations such that CSP($\mathbf{A}$) is tractable whenever $\mathbf{A}$ admits one of the operations of $\mathcal{F}$ as a polymorphism?

Some concrete examples of structures for which the complexity of the corresponding CSP is unknown are

- $(\mathbb{Q}, \{(a,b) \mid |a-b| = 1\}, \{(a,b,c) \mid a > b \vee a > c\})$, and
- $(\mathbb{Q}, \{(a,b,c,d) \mid a = b \rightarrow c = d\}, \{(a, a+1) \mid a \in \mathbb{Q}\}, \{(a,b,c) \mid a > b \vee a \geq c\})$.

For both these structures, removing the first relation yields a structure that admits min as a polymorphism, so that by the Corollary above we know that the corresponding CSP is polynomial-time tractable.

Moreover, as we said in the preamble, the classification project for $(\mathbb{Q}, +_1, <)$ is a part of a bigger classification project, that of $(\mathbb{Q}, +, <, 1)$. In [**?**], Bodirsky et al. described a dichotomy result for the class of first-order expansions of $(\mathbb{Q}, +)$, and for the class of first-order expansions of $(\mathbb{Q}, \{(a,b,c,d) \mid a+b = c+d\})$. Thus, the classification project for $(\mathbb{Q}, +, <, 1)$ has been approached from two angles. Using our partial knowledge of the complexity of reducts of $(\mathbb{Q}, <, +_1)$ and $(\mathbb{Q}, +)$, one might be able to describe maximally tractable reducts of $(\mathbb{Q}, +, <)$.

Finally, another classification project of interest is that of $(\mathbb{Z}, +, <)$. This classification project has been started in [1], and the full complexity classification for reducts of $(\mathbb{Z}, +_1)$ is done in a forthcoming article [2]. Interestingly, this work uses connections between the constraint satisfaction problems of structures over the domain $\mathbb{Z}$ and corresponding structures over the domain $\mathbb{Q}$. It may be fruitful to investigate this connection for reducts of $(\mathbb{Z}, +_1, <)$ and $(\mathbb{Q}, +_1, <)$.

References

[1] Bodirsky, M., Dalmau, V., Martin, B., and Pinsker, M. Distance constraint satisfaction problems. In *Proceedings of Mathematical Foundations of Computer Science* (August 2010), P. Hlinený and A. Kucera, Eds., Lecture Notes in Computer Science, Springer Verlag, pp. 162–173.

[2] Bodirsky, M., Martin, B., and Mottet, A. Constraint satisfaction problems over the integers with successor. In *Proceedings of ICALP* (2015), pp. 256–267. ArXiv:1503.08572.