

Fixed-Template Promise Model Checking Problems

Kristina Asimi ✉

Department of Algebra, Faculty of Mathematics and Physics,
Charles University, Prague, Czechia

Libor Barto ✉ 🏠 

Department of Algebra, Faculty of Mathematics and Physics,
Charles University, Prague, Czechia

Silvia Butti ✉ 🏠 

Department of Information and Communication Technologies,
Universitat Pompeu Fabra, Barcelona, Spain

Abstract

The fixed-template constraint satisfaction problem (CSP) can be seen as the problem of deciding whether a given primitive positive first-order sentence is true in a fixed structure (also called model). We study a class of problems that generalizes the CSP simultaneously in two directions: we fix a set \mathcal{L} of quantifiers and Boolean connectives, and we specify two versions of each constraint, one strong and one weak. Given a sentence which only uses symbols from \mathcal{L} , the task is to distinguish whether the sentence is true in the strong sense, or it is false even in the weak sense.

We classify the computational complexity of these problems for the existential positive equality-free fragment of first-order logic, i.e., $\mathcal{L} = \{\exists, \wedge, \vee\}$, and we prove some upper and lower bounds for the positive equality-free fragment, $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$. The partial results are sufficient, e.g., for all extensions of the latter fragment.

2012 ACM Subject Classification Theory of computation → Complexity theory and logic

Keywords and phrases Model Checking Problem, First-Order Logic, Promise Constraint Satisfaction Problem, Multi-Homomorphism

Digital Object Identifier 10.4230/LIPIcs.CP.2022.2

Funding Kristina Asimi and Libor Barto have received funding from the European Research Council (ERC) under the European Unions Horizon 2020 research and innovation programme (Grant Agreement No. 771005, CoCoSym). Silvia Butti was supported by a MICCIN grant PID2019-109137GB-C22 and by a fellowship from “la Caixa” Foundation (ID 100010434). The fellowship code is LCF/BQ/DI18/11660056. This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 713673.

1 Introduction

The fixed-template finite-domain constraint satisfaction problem (CSP) is a framework for expressing many computational problems such as various versions of logical satisfiability, graph coloring, and systems of equations, see the survey [3]. A convenient formalization, that we adopt in this paper, is as follows: a *template* is a relational structure \mathbb{A} , and the *CSP over \mathbb{A}* is the problem of deciding whether a given $\{\exists, \wedge\}$ -sentence is true in \mathbb{A} . Here, an $\{\exists, \wedge\}$ -sentence is a sentence of first-order logic that uses only the relation symbols of \mathbb{A} , the logical connective \wedge , and the quantifier \exists . To see that this formalization indeed expresses *constraint* satisfaction problems, consider, e.g., the sentence $\exists x \exists y \exists z R(x, y) \wedge S(y, z)$: this sentence is true in a structure \mathbb{A} if the variables x, y, z can be evaluated so that both atomic formulas (constraints) are satisfied in \mathbb{A} .



© Kristina Asimi, Libor Barto, and Silvia Butti;
licensed under Creative Commons License CC-BY 4.0

28th International Conference on Principles and Practice of Constraint Programming (CP 2022).

Editor: Christine Solnon; Article No. 2; pp. 2:1–2:17



Leibniz International Proceedings in Informatics

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

Motivated by recent developments in the area, we study an extension of this framework in two simultaneous directions. One direction, discussed in Subsection 1.1, is to enable other choices of permitted quantifiers and connectives. Another direction, discussed in Subsection 1.2, is to consider two versions of each relation, strong and weak (a so-called promise problem). Our contributions are then described in Subsection 1.3.

1.1 Model checking problem parametrized by the model

The model checking problem [13] takes as input a structure \mathbb{A} (often called a model) and a sentence ϕ in a specified logic and asks whether $\mathbb{A} \models \phi$, i.e., whether \mathbb{A} satisfies ϕ . We study the situation where \mathbb{A} is a fixed finite relational structure, so the input is simply ϕ , and the logic is a fragment of the first-order logic obtained by restricting the allowed quantifiers to a subset \mathcal{L} of $\{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$. Thus, for each \mathbb{A} and each of the 2^7 choices for \mathcal{L} , we obtain a computational problem, which we call the *\mathcal{L} -Model Checking Problem over \mathbb{A}* and denote $\mathcal{L}\text{-MC}(\mathbb{A})$.

The computational complexity classification of $\{\exists, \wedge\}\text{-MC}(\mathbb{A})$, i.e., CSP over \mathbb{A} , has been a very active research program in the last 20 years, which culminated in the celebrated dichotomy theorem obtained independently in [6] and [18]: each CSP over \mathbb{A} is in P (solvable in polynomial time) or is NP-complete. For the case $\mathcal{L} = \{\exists, \forall, \wedge\}$, $\mathcal{L}\text{-MC}(\mathbb{A})$ is the so called *quantified CSP*, another well-studied class of problems, see the survey [16]. It was widely believed that this class exhibits a P/NP-complete/PSPACE-complete trichotomy [8]. A recent breakthrough [19] shows that at least three more complexity classes appear within quantified CSPs, and ongoing work suggests that even 6 is not the final number. In any case, the full complexity classification of $\{\exists, \forall, \wedge\}\text{-MC}(\mathbb{A})$ is a challenging open problem.

The remaining $2^7 - 2$ choices for \mathcal{L} do not need to be considered separately. For instance, $\{\exists, \wedge, =\}\text{-MC}(\mathbb{A})$ is no harder than $\{\exists, \wedge\}\text{-MC}(\mathbb{A})$ because equalities can be propagated out in this case, and $\{\forall, \vee\}\text{-MC}(\mathbb{A})$ is dual to $\{\exists, \wedge\}\text{-MC}(\mathbb{A})$ so we get a P/coNP-complete dichotomy for free, etc. Moreover, some choices of \mathcal{L} , such as $\mathcal{L} = \{\exists, \vee\}$, lead to very simple problems. It turns out [14] (see Subsection 3.3) that, in addition to $\mathcal{L} = \{\exists, \wedge\}$ and $\mathcal{L} = \{\exists, \forall, \wedge\}$, only two more fragments need to be considered in order to fully understand the complexity of $\mathcal{L}\text{-MC}(\mathbb{A})$, namely $\mathcal{L} = \{\exists, \wedge, \vee\}$ and $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$.

The former fragment was addressed in [14]: except for a simple case solvable in polynomial time (in fact, L, the logarithmic space), all the remaining problems are NP-complete. The latter fragment turned out to be more challenging but, after a series of partial results [14, 11, 17] (see also [15, 7]), the full complexity classification was given in [12, 13]: each problem in this class is in P (even L), or is NP-complete, coNP-complete, or PSPACE-complete. These results are summarized in Table 1.

■ **Table 1** Known complexity results for $\mathcal{L}\text{-MC}(\mathbb{A})$.

$\mathcal{L}\text{-MC}(\mathbb{A})$	Complexity
$\{\exists, \wedge\}\text{-MC}(\mathbb{A})$ (CSP)	dichotomy: P or NP-complete
$\{\exists, \forall, \wedge\}\text{-MC}(\mathbb{A})$ (QCSP)	≥ 6 classes
$\{\exists, \wedge, \vee\}\text{-MC}(\mathbb{A})$	dichotomy: L or NP-complete
$\{\forall, \exists, \wedge, \vee\}\text{-MC}(\mathbb{A})$	tetrachotomy: L, NP-complete, coNP-complete, PSPACE-complete

1.2 Promise model checking problem

The Promise CSP is a recently introduced extension of the CSP framework motivated by open problems in (in)approximability of satisfiability and coloring problems [1, 5, 2]. The template consists of two structures \mathbb{A} and \mathbb{B} of the same signature, where \mathbb{A} specifies a strong form of each relation and \mathbb{B} its weak form. The Promise CSP over (\mathbb{A}, \mathbb{B}) is then the problem of distinguishing $\{\exists, \wedge\}$ -sentences that are true in \mathbb{A} from those that are not true in \mathbb{B} .

For example, by choosing an appropriate template, we obtain the problem of distinguishing k -colorable graphs from those that are not even l -colorable (where $k \leq l$ are fixed), a problem whose complexity is notoriously open.

The generalization of Promise CSP over (\mathbb{A}, \mathbb{B}) to an arbitrary choice $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$ is referred to as the \mathcal{L} -Promise Model Checking Problem over (\mathbb{A}, \mathbb{B}) and is denoted \mathcal{L} -PMC (\mathbb{A}, \mathbb{B}) . Similarly as in the special case $\mathbb{A} = \mathbb{B}$, which is exactly \mathcal{L} -MC (\mathbb{A}) , it is sufficient to consider only four fragments. A full complexity classification for $\{\exists, \wedge\}$ -PMC (i.e., Promise CSP) is much desired but widely open, and $\{\exists, \forall, \wedge\}$ -PMC is likely even harder. This work concentrates on the remaining two classes of problems, $\{\exists, \wedge, \vee\}$ -PMC and $\{\exists, \forall, \wedge, \vee\}$ -PMC.

Our motivation was that these cases might be substantially simpler, as indicated by the non-promise special case, and at the same time, the investigation could uncover interesting intermediate problems towards the grand endeavor of understanding the sources of tractability and hardness in computation. We believe that our findings confirm this hope.

► **Example 1.** Consider structures \mathbb{A} and \mathbb{B} with a single relation symbol $=$ interpreted as the equality on a three-element domain in \mathbb{A} and as the equality on a two-element domain in \mathbb{B} . For $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$, both \mathcal{L} -MC (\mathbb{A}) and \mathcal{L} -MC (\mathbb{B}) are PSPACE-complete problems, see [14].

It is not hard to see that every \mathcal{L} -sentence true in \mathbb{A} is also true in \mathbb{B} . In this sense, the relation in \mathbb{A} is stronger than the relation in \mathbb{B} . On the other hand, there are \mathcal{L} -sentences true in \mathbb{B} that are not true in \mathbb{A} , e.g., $\phi = \forall x \exists y \forall z (z = x) \vee (z = y)$. Therefore, \mathcal{L} -PMC (\mathbb{A}, \mathbb{B}) could potentially be easier than the above non-promise problems – instances such as ϕ need not be considered (there is no requirement on the algorithm for such inputs). Nevertheless, the problem remains PSPACE-complete, as shown in Proposition 18.

1.3 Contributions

Theorem 7 and Theorem 13 provide basics for an algebraic approach to $\{\exists, \wedge, \vee\}$ -PMC and $\{\exists, \forall, \wedge, \vee\}$ -PMC by characterizing definability in terms of compatible functions: multi-homomorphisms for the $\{\exists, \wedge, \vee\}$ fragment and surjective multi-homomorphisms (*smuhoms*) for $\{\exists, \forall, \wedge, \vee\}$. The proofs can be obtained as relatively straightforward generalizations of the proofs for MC in [13]; however, we believe that our approach is somewhat more transparent. In particular, it allows us to easily characterize meaningful templates for these problems (Propositions 6 and 12).

For $\{\exists, \wedge, \vee\}$ -PMC, we obtain an L/NP-complete dichotomy in Theorem 9. It turns out that, apart from some simple cases, the problem is NP-complete. Interestingly, there is a “single reason” for hardness: the NP-hardness of coloring a rainbow colorable hypergraph from [9].

For $\{\exists, \forall, \wedge, \vee\}$ -PMC, our complexity results are only partial, leaving two gaps for further investigation. The results are sufficient for full complexity classification of \mathcal{L} -PMC (\mathbb{A}, \mathbb{B}) in the case that $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ and one of the structures \mathbb{A}, \mathbb{B} has a two-element domain, and also in the case that $\mathcal{L} \supseteq \{\exists, \forall, \wedge, \vee\}$. We also give some examples where our efforts

2:4 Fixed-Template Promise Model Checking Problems

have failed so far. One such example is a particularly interesting $\{\exists, \forall, \wedge, \vee\}$ -PMC over 3-element domains: given a $\{\exists, \forall, \wedge, \vee\}$ -sentence ϕ whose atomic formulas are all of the form $R^i(x)$, $i \in \{1, 2, 3\}$, distinguish between the case where ϕ is true when $R^i(x)$ is interpreted as “ $x = i$ ”, and the case where ϕ is false when $R^i(x)$ is interpreted as “ $x \neq i$ ”.

Our complexity results are summarized in Table 2, the conditions for $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ are stated in terms of special surjective multi-homomorphisms of the template, introduced in Subsection 5.3.

■ **Table 2** Complexity results for \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}).

\mathcal{L} -PMC(\mathbb{A}, \mathbb{B})	Condition	Complexity
$\{\exists, \forall, \wedge\}$ -PMC(\mathbb{A}, \mathbb{B})		L/NP-complete
$\{\exists, \forall, \wedge, \vee\}$ -PMC(\mathbb{A}, \mathbb{B})	A \bar{E} -smuhom, or A-smuhom and E-smuhom and \mathbb{A}, \mathbb{B} digraphs	L
	A-smuhom and \bar{E} -smuhom	NP \cap coNP
	A-smuhom, no \bar{E} -smuhom	NP-complete
	\bar{E} -smuhom, no A-smuhom	coNP-complete
	no A-smuhom and no \bar{E} -smuhom	NP-hard and coNP-hard
$\{\exists, \forall, \wedge, \vee, =\}$ -PMC(\mathbb{A}, \mathbb{B}), $\{\exists, \forall, \wedge, \vee, \neq\}$ -PMC(\mathbb{A}, \mathbb{B}), $\{\exists, \forall, \wedge, \vee, \neg\}$ -PMC(\mathbb{A}, \mathbb{B})		L/PSPACE-complete

2 Preliminaries

Structures. We use a standard model-theoretic terminology, but restrict the generality of some concepts for the purposes of this paper. A *relation* of arity $n \geq 1$ on a set A is a set of n -tuples of elements of A , i.e., a subset of A^n . The *complement* of a relation S is denoted $\bar{S} := A^n \setminus S$. The equality relation on A is denoted $=_A$ and the disequality relation \neq_A . Components of a tuple \mathbf{a} are referred to as a_1, a_2, \dots , i.e., $\mathbf{a} = (a_1, \dots, a_n)$.

A *signature* is a nonempty collection of relation symbols each with an associated arity, denoted $\text{ar}(R)$ for a relation symbol R . A *relational structure* (also called a *model*) \mathbb{A} in the signature σ , or simply a *structure*, consists of a finite set A of size at least two, called the *universe* of \mathbb{A} , and a nonempty proper relation $\emptyset \subsetneq R^{\mathbb{A}} \subsetneq A^{\text{ar}(R)}$ for each symbol R in σ , called the *interpretation* of R in \mathbb{A} . Two structures are called *similar* if they are in the same signature. The *complement* of a relational structure \mathbb{A} is obtained by taking complements of all relations in the structure and is denoted $\bar{\mathbb{A}}$. A structure over a signature containing a single binary relation symbol is called a *digraph*.

We emphasize that the universe of a structure is denoted by the same letter as the structure, that the universe of every structure in this paper is assumed to be finite and at least two-element, and that each relation in a structure is assumed to be at least unary, nonempty and proper. These nonstandard requirements are placed for technical convenience and do not significantly decrease the generality of our results.

Given two similar structures \mathbb{A} and \mathbb{B} , a function f from A to B is called a *homomorphism* from \mathbb{A} to \mathbb{B} if $f(\mathbf{a}) \in R^{\mathbb{B}}$ for any $\mathbf{a} \in R^{\mathbb{A}}$, where $f(\mathbf{a})$ is computed component-wise. We only work with total functions, that is, $f(a)$ is defined for every $a \in A$.

Multi-homomorphisms. A *multi-valued function* f from A to B is a mapping from A to $\mathcal{P}_{\neq \emptyset} B$, the set of all nonempty subsets of B . It is called *surjective* if for every $b \in B$, there exists $a \in A$ such that $b \in f(a)$. The *inverse* of a surjective multi-valued function f from A to B is the multi-valued function from B to A defined by $f^{-1}(b) = \{a : b \in f(a)\}$. For

a tuple $\mathbf{a} \in A^n$ we write $f(\mathbf{a})$ for $f(a_1) \times \cdots \times f(a_n)$. The value $\max\{|f(a)| : a \in A\}$ is referred to as the *multiplicity* of f ; in particular, multi-valued functions of multiplicity one are essentially functions. For two multi-valued functions f and f' from A to B , we say that f' is *contained in* f if $f'(a) \subseteq f(a)$ for each $a \in A$.

Given two similar structures \mathbb{A} and \mathbb{B} , a multi-valued function f from A to B is called a *multi-homomorphism*¹ from \mathbb{A} to \mathbb{B} if for any R in the signature and any $\mathbf{a} \in R^{\mathbb{A}}$, we have $f(\mathbf{a}) \subseteq R^{\mathbb{B}}$, i.e., $\mathbf{b} \in R^{\mathbb{B}}$ whenever $b_i \in f(a_i)$ for each $i \in [\text{ar}(R)] = \{1, 2, \dots, \text{ar}(R)\}$. Notice that if f is a multi-homomorphism from \mathbb{A} to \mathbb{B} , then so is any multi-valued function contained in f . In particular, if f is a multi-homomorphism from \mathbb{A} to \mathbb{B} , then any function $g : A \rightarrow B$ with $g(a) \in f(a)$ for each $a \in A$ is a homomorphism from \mathbb{A} to \mathbb{B} . The converse does not hold in general, as witnessed by structures $\mathbb{A} = \mathbb{B}$ with a single binary equality relation and any multi-valued function of multiplicity greater than one.

The set of all multi-homomorphisms from \mathbb{A} to \mathbb{B} is denoted by $\text{MuHom}(\mathbb{A}, \mathbb{B})$ and the set of all surjective multi-homomorphisms by $\text{SMuHom}(\mathbb{A}, \mathbb{B})$.

Fragments of first-order logic. Let $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$ and fix some signature. By an \mathcal{L} -sentence (resp., \mathcal{L} -formula) we mean a sentence (resp., formula) of first-order logic that only uses variables (denoted x_i, y_i, z_i), relation symbols in the signature, and connectives and quantifiers in \mathcal{L} . We refer to this fragment of first-order logic as the \mathcal{L} -logic.

The *prenex normal form* of an \mathcal{L} -formula is an equivalent formula that begins with quantified variables followed by a quantifier-free formula. The prenex normal form can be computed in logarithmic space and it is an \mathcal{L} -formula whenever \mathcal{L} does not contain the negation.

For a structure \mathbb{A} in the signature and an \mathcal{L} -sentence ϕ , we write $\mathbb{A} \models \phi$ if ϕ is satisfied in \mathbb{A} . More generally, given an \mathcal{L} -formula ψ , a tuple of distinct variables (v_1, \dots, v_n) which contains every free variable of ψ and a tuple $(a_1, \dots, a_n) \in A^n$, we write $\mathbb{A} \models \psi(a_1, \dots, a_n)$ if ψ is satisfied when v_1, \dots, v_n are evaluated as $\varepsilon_A(v_1) = a_1, \dots, \varepsilon_A(v_n) = a_n$, respectively. Notice that variables v_1, \dots, v_n indeed need to be pairwise distinct, otherwise this notation would not make sense. The tuple (v_1, \dots, v_n) is often specified by writing $\psi = \psi(v_1, \dots, v_n)$.

We say that a relation $S \subseteq A^n$ is \mathcal{L} -definable from \mathbb{A} if there exists an \mathcal{L} -formula $\psi(v_1, \dots, v_n)$ such that, for all $(a_1, \dots, a_n) \in A^n$, we have $(a_1, \dots, a_n) \in S$ if and only if $\mathbb{A} \models \psi(a_1, \dots, a_n)$. In this case, we also say that $\psi(v_1, \dots, v_n)$ defines S in \mathbb{A} .

3 Promise model checking

In this section we define the promise model checking problem restricted to $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$. We start by briefly discussing the non-promise setting.

3.1 Model checking problem

Let $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$ and \mathbb{A} be a structure in a signature σ . Recall that the \mathcal{L} -Model Checking Problem over \mathbb{A} , denoted $\mathcal{L}\text{-MC}(\mathbb{A})$, is the problem of deciding whether a given \mathcal{L} -sentence ϕ (in the same signature as \mathbb{A}) is true in \mathbb{A} .

A simple but important observation sometimes allows us to compare the complexity of the \mathcal{L} -MC problems over two templates \mathbb{A} and \mathbb{C} with the same universe $A = C$ but possibly different signatures: If every relation in \mathbb{C} is \mathcal{L} -definable from \mathbb{A} , then $\mathcal{L}\text{-MC}(\mathbb{C})$ can be reduced in polynomial-time (even logarithmic space) to $\mathcal{L}\text{-MC}(\mathbb{A})$. Indeed, the reduction amounts to replacing atomic formulas of the form $R(\mathbf{v})$ by their definitions.

¹ We deviate here from the terminology of [12, 11] because it would not work well in the promise setting.

The starting point of the algebraic approach to \mathcal{L} -MC is to find a characterization of definability in terms of certain “compatible functions” or “symmetries” (so called polymorphisms for $\mathcal{L} = \{\exists, \wedge, =\}$ [3], surjective polymorphisms for $\mathcal{L} = \{\exists, \forall, \wedge, =\}$ [16], multi-endomorphisms for $\mathcal{L} = \{\exists, \wedge, \vee\}$, surjective multi-endomorphisms for $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ [13]; see also [4]). Because such characterizations are central in this paper as well, we now explain the basic idea for a simple case.

For $\mathcal{L} = \{\exists, \wedge, \vee, =\}$, the appropriate type of compatible function is endomorphism: a nonempty relation $S \subseteq A^n$ is \mathcal{L} -definable from \mathbb{A} if and only if it is invariant under every endomorphism of \mathbb{A} (i.e., a homomorphism from \mathbb{A} to itself). The forward direction is well-known and easy to verify. For the backward direction, assume $A = [k] := \{1, \dots, k\}$ and consider the following formula.

$$\phi(x_1, \dots, x_k) := \bigwedge_{R \in \sigma} \bigwedge_{\mathbf{r} \in R^A} R(x_{r_1}, \dots, x_{r_{\text{ar}(R)}}) \quad (1)$$

It follows immediately from definitions that, for any structure \mathbb{E} in the signature of \mathbb{A} , $\mathbb{E} \models \phi(e_1, \dots, e_k)$ if and only if the mapping defined by $i \mapsto e_i$ for each $i \in [k]$ is a homomorphism from \mathbb{A} to \mathbb{E} . This in particular holds for $\mathbb{E} = \mathbb{A}$. By existential quantification we can then obtain an \mathcal{L} -formula defining the closure of any tuple $\mathbf{a} \in A^n$ with distinct entries under endomorphisms of \mathbb{A} ; e.g., $\psi(x_1, x_3, x_2) := (\exists x_4)(\exists x_5) \dots (\exists x_k)\phi$ defines the closure of $(1, 3, 2)$ under endomorphisms. Using $=$ we can also define closures of the remaining tuples with repeated entries. Finally, S is the union of closures of its members (since it is closed under endomorphisms of \mathbb{A}), so S can be defined by a disjunction of formulas that we have already found (after appropriately renaming variables).

Notice that this construction would not work without the equality in \mathcal{L} because of tuples with repeated entries. This is the reason why we need to work with multi-valued functions for the equality-free logics that we deal with in this paper.

3.2 Promise model checking problem

Let $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, =, \neq, \neg\}$. The \mathcal{L} -Promise Model Checking Problem over a pair of similar structures (\mathbb{A}, \mathbb{B}) is the problem of distinguishing \mathcal{L} -sentences ϕ that are true in \mathbb{A} from those that are not true in \mathbb{B} . This problem makes sense only if every \mathcal{L} -sentence that is true in \mathbb{A} is also true in \mathbb{B} ; we call such pairs \mathcal{L} -PMC templates.

► **Definition 2.** *A pair of similar structures (\mathbb{A}, \mathbb{B}) is called an \mathcal{L} -PMC template if $\mathbb{A} \models \phi$ implies $\mathbb{B} \models \phi$ for every \mathcal{L} -sentence ϕ in the signature of \mathbb{A} and \mathbb{B} .*

Given an \mathcal{L} -PMC template (\mathbb{A}, \mathbb{B}) , the \mathcal{L} -Promise Model Checking Problem over (\mathbb{A}, \mathbb{B}) , denoted $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$, is the following problem.

Input: an \mathcal{L} -sentence ϕ in the signature of \mathbb{A} and \mathbb{B} ;

Output: Yes if $\mathbb{A} \models \phi$; No if $\mathbb{B} \not\models \phi$.

The definition of a template guarantees that the sets of **Yes**-instances and **No**-instances are disjoint. However, their union need not be the whole set of \mathcal{L} -sentences; an algorithm for \mathcal{L} -PMC is only required to produce correct outputs for **Yes**-instances and **No**-instances. Alternatively, we are *promised* that the input sentence is a **Yes**-instance or a **No**-instance. The complexity-theoretic notions (such as membership in NP, NP-completeness, reductions) can be adjusted naturally for the promise setting. We write $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ if the former problem can be reduced to the latter problem by a logarithmic space reduction, that is, a logarithmic space transformation that maps each **Yes**-instance ϕ of $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D})$ to a **Yes**-instance ψ of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ (equivalently, $\mathbb{C} \models \phi$ must imply $\mathbb{A} \models \psi$) and **No**-instances to **No**-instances (equivalently, $\mathbb{B} \models \psi$ must imply $\mathbb{D} \models \phi$).

An appropriate adjustment of definability for the promise setting is as follows. Note that we do not allow the negation in \mathcal{L} , otherwise the concept would need to be defined differently because of the inclusions in the definition.

► **Definition 3.** Assume $\neg \notin \mathcal{L}$ and let (\mathbb{A}, \mathbb{B}) be a pair of similar structures. We say that a pair of relations (S, T) , where $S \subseteq A^n$ and $T \subseteq B^n$, is *promise- \mathcal{L} -definable* (or *p- \mathcal{L} -definable*) from (\mathbb{A}, \mathbb{B}) if there exist relations S' and T' and an \mathcal{L} -formula $\psi(v_1, \dots, v_n)$ such that $S \subseteq S'$, $T' \subseteq T$, $\psi(v_1, \dots, v_n)$ defines S' in \mathbb{A} , and $\psi(v_1, \dots, v_n)$ defines T' in \mathbb{B} .

We say that an \mathcal{L} -PMC template (\mathbb{C}, \mathbb{D}) is *p- \mathcal{L} -definable* from (\mathbb{A}, \mathbb{B}) (the signatures can differ) if $(Q^{\mathbb{C}}, Q^{\mathbb{D}})$ is p- \mathcal{L} -definable from (\mathbb{A}, \mathbb{B}) for each relation symbol Q in the signature of \mathbb{C} and \mathbb{D} .

► **Theorem 4.** Assume $\neg \notin \mathcal{L}$. If (\mathbb{A}, \mathbb{B}) and (\mathbb{C}, \mathbb{D}) are \mathcal{L} -PMC templates such that (\mathbb{C}, \mathbb{D}) is p- \mathcal{L} -definable from (\mathbb{A}, \mathbb{B}) , then $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$.

Proof. The reduction is to replace each atomic $Q(\mathbf{v})$ by the corresponding formula ψ from Definition 3. For correctness of this reduction, observe that an \mathcal{L} -sentence which is true in a structure \mathbb{E} remains true when we add tuples to the relations of \mathbb{E} (since \mathcal{L} does not contain \neg). ◀

3.3 Interesting fragments

We now explain why only four fragments of first-order logic need to be considered in order to fully understand the problems $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$. Observe first that if \mathcal{L} does not contain any connective (\wedge, \vee) , or \mathcal{L} does not contain any quantifier (\exists, \forall) , or $\mathcal{L} \subseteq \{\exists, \forall\}$, then each \mathcal{L} -PMC is in L, the logarithmic space. (In some of these cases we do not even have any valid inputs in our definition of structures.)

Secondly, notice that $(\mathcal{L} \cup \{=\})\text{-PMC}(\mathbb{A}, \mathbb{B})$ is essentially the same as $\mathcal{L}\text{-PMC}(\mathbb{A}', \mathbb{B}')$, where \mathbb{A}' and \mathbb{B}' are obtained from the original structures by adding a fresh binary symbol Q to the signature and setting $Q^{\mathbb{A}'}$ to $=_A$ and $Q^{\mathbb{B}'}$ to $=_B$. The disequality is dealt with analogously, thus we can and shall restrict to fragments with $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee, \neg\}$.

Next, we deal with the negation. If \neg is in \mathcal{L} , and \mathcal{L} contains a quantifier and a connective, then it is enough to consider the case $\mathcal{L} = \{\exists, \forall, \wedge, \vee, \neg\}$ since the remaining quantifier and connective can be expressed using negation. Moreover, the complements of relations can also be expressed, so we may assume that each template (\mathbb{A}, \mathbb{B}) is *closed under complementation*, meaning that for every symbol R in the signature, we have a symbol \bar{R} interpreted as $\bar{R}^{\mathbb{A}} = \overline{R^{\mathbb{A}}}$, $\bar{R}^{\mathbb{B}} = \overline{R^{\mathbb{B}}}$. But then \neg is no longer necessary since we can propagate the negations inwards in an input sentence. We are down to $\mathcal{L} \subseteq \{\exists, \forall, \wedge, \vee\}$.

Finally, note that $\mathbb{E} \models \neg\phi$, where ϕ is an \mathcal{L} -sentence, is equivalent to $\bar{\mathbb{E}} \models \phi'$ where ϕ' is an \mathcal{L}' -sentence and \mathcal{L}' is obtained from \mathcal{L} by swapping $\forall \leftrightarrow \exists$ and $\vee \leftrightarrow \wedge$ (ϕ' can be, again, computed from $\neg\phi$ by inward propagation). It follows that $\phi \mapsto \phi'$ transforms every **Yes**-instance (resp., **No**-instance) of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ to a **No**-instance (resp., **Yes**-instance) of $\mathcal{L}'\text{-PMC}(\bar{\mathbb{B}}, \bar{\mathbb{A}})$, and a similar “dual” reduction works in the opposite direction. Therefore, the latter PMC has the “dual” complexity to the former PMC, e.g., if the former is NP-complete, then the latter is coNP-complete; and if the former is PSPACE-complete, then the latter is PSPACE-complete as well. We will refer to this reasoning as the *duality argument*.

Eliminating one of the logic fragments from each of the “dual” pairs, we are left with only four fragments: $\mathcal{L} = \{\exists, \wedge\}$ (whose \mathcal{L} -PMC is Promise CSP), $\mathcal{L} = \{\exists, \forall, \wedge\}$ (Promise Quantified CSP), $\mathcal{L} = \{\exists, \wedge, \vee\}$, and $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$. We investigate the last two separately in the next two sections.

4 Existential positive fragment

This section concerns the existential positive equality-free logic, that is, the \mathcal{L} -logic with $\mathcal{L} = \{\exists, \wedge, \vee\}$. We fix this \mathcal{L} for the entire section.

4.1 Characterization of templates and p- \mathcal{L} -definability

We start by characterizing \mathcal{L} -PMC templates. One direction of the characterization follows from the discussion below (1), the other one from the following observation.

► **Lemma 5.** *Let f be a multi-homomorphism from \mathbb{A} to \mathbb{B} , let $\phi(x_1, \dots, x_n)$ be a quantifier-free \mathcal{L} -formula in the same signature, and let $\mathbf{a} \in A^n$, $\mathbf{b} \in B^n$. If $\mathbb{A} \models \phi(\mathbf{a})$ and $\mathbf{b} \in f(\mathbf{a})$, then $\mathbb{B} \models \phi(\mathbf{b})$.*

Proof. The claim holds for atomic formulas by definition of multi-homomorphisms. The proof is then finished by induction on the complexity of ϕ ; both \vee and \wedge are dealt with in a straightforward way. ◀

► **Proposition 6.** *A pair (\mathbb{A}, \mathbb{B}) of similar structures is an \mathcal{L} -PMC template if and only if there exists a homomorphism from \mathbb{A} to \mathbb{B} .*

Proof. Suppose that there exists a homomorphism from \mathbb{A} to \mathbb{B} and $\mathbb{A} \models \phi$, where $\phi = \exists x_1 \exists x_2 \dots \exists x_n \phi'(x_1, \dots, x_n)$ is in prenex normal form. Then we have $\mathbb{A} \models \phi'(\mathbf{a})$ for some $\mathbf{a} \in A^n$, therefore $\mathbb{B} \models \phi'(f(\mathbf{a}))$ by Lemma 5, and it follows that $\mathbb{B} \models \phi$.

For the forward implication, observe that the sentence obtained from the formula (1) by existentially quantifying all the variables is true in \mathbb{A} (as there exists a homomorphism from \mathbb{A} to \mathbb{A} – the identity), so it must be true in \mathbb{B} , giving us a homomorphism from \mathbb{A} to \mathbb{B} . ◀

Note that this characterization would remain the same if we add $=$ to \mathcal{L} (and/or remove \vee). For the following characterization of promise definability, the absence of the equality relation does make a difference, which is why we need to use multi-homomorphisms instead of homomorphisms.

► **Theorem 7.** *Let (\mathbb{A}, \mathbb{B}) and (\mathbb{C}, \mathbb{D}) be \mathcal{L} -PMC templates such that $A = C$ and $B = D$. Then (\mathbb{C}, \mathbb{D}) is p- \mathcal{L} -definable from (\mathbb{A}, \mathbb{B}) if and only if $\text{MuHom}(\mathbb{A}, \mathbb{B}) \subseteq \text{MuHom}(\mathbb{C}, \mathbb{D})$. Moreover, in such a case, $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$.*

Proof. It is enough to verify the equivalence, since then the second claim follows from Theorem 4. To prove the forward implication, assume that (\mathbb{C}, \mathbb{D}) is p- \mathcal{L} -definable from (\mathbb{A}, \mathbb{B}) , let $f \in \text{MuHom}(\mathbb{A}, \mathbb{B})$, and let Q be a symbol in the signature of \mathbb{C} and \mathbb{D} . To show that $f(\mathbf{a}) \subseteq Q^{\mathbb{D}}$ for any $\mathbf{a} \in Q^{\mathbb{C}}$ we apply Lemma 5 as follows. We have $\mathbb{A} \models \psi(\mathbf{a})$, where $\psi(\mathbf{x}) = \exists y_1 \exists y_2 \dots \exists y_m \psi'(\mathbf{x}, \mathbf{y})$ is a formula from Definition 3, turned into prenex normal form. Then $\mathbb{A} \models \psi'(\mathbf{a}, \mathbf{a}')$ for some $\mathbf{a}' \in A^m$, thus $\mathbb{B} \models \psi'(\mathbf{b}, \mathbf{b}')$ for any $\mathbf{b} \in f(\mathbf{a})$ and $\mathbf{b}' \in f(\mathbf{a}')$ by Lemma 5. Therefore, $\mathbb{B} \models \psi(\mathbf{b})$ and, finally, $\mathbf{b} \in Q^{\mathbb{D}}$, as required.

For the backward implication, assume that $\text{MuHom}(\mathbb{A}, \mathbb{B}) \subseteq \text{MuHom}(\mathbb{C}, \mathbb{D})$, denote σ the signature of \mathbb{A} and \mathbb{B} , and consider an n -ary relational symbol Q in the signature of \mathbb{C} and \mathbb{D} . To prove the claim, we need to find a formula $\psi(x_1, \dots, x_n)$ that defines, in \mathbb{A} , a relation containing $Q^{\mathbb{C}}$ and, in \mathbb{B} , a relation contained in $Q^{\mathbb{D}}$.

For simplicity, assume $A = [k]$ and consider the formula

$$\phi(x_{1,1}, \dots, x_{1,n}, x_{2,1}, \dots, x_{2,n}, \dots, x_{k,n}) := \bigwedge_{R \in \sigma} \bigwedge_{\mathbf{r} \in R^k} \bigwedge_{\mathbf{j} \in [n]^{\text{ar}(R)}} R(x_{r_1, j_1}, \dots, x_{r_{\text{ar}(R)}, j_{\text{ar}(R)}}) \quad (2)$$

It follows immediately from definitions that, for any structure \mathbb{E} in the signature σ , we have $\mathbb{E} \models \phi(e_{1,1}, \dots, e_{k,n})$ if and only if the mapping $i \mapsto \{e_{i,1}, \dots, e_{i,n}\}$, $1 \leq i \leq k$ is a multi-homomorphism from \mathbb{A} to \mathbb{E} . Therefore, for any $\mathbf{a} \in A^n$, the formula $\tau_{\mathbf{a}}(x_1, \dots, x_n)$, obtained from ϕ by renaming $x_{a_i,i}$ to x_i and existentially quantifying the remaining variables, defines in \mathbb{E} the union of $f(\mathbf{a})$ over $f \in \text{MuHom}(\mathbb{A}, \mathbb{E})$ of multiplicity at most n . This relation is clearly equal to the union of $f(\mathbf{a})$ over all $f \in \text{MuHom}(\mathbb{A}, \mathbb{E})$. The sought after formula ψ is then the disjunction of $\tau_{\mathbf{a}}$ over all $\mathbf{a} \in Q^C$: it defines in \mathbb{A} a relation containing Q^C (because of the identity “multi”-homomorphism $\mathbb{A} \rightarrow \mathbb{A}$) and, in \mathbb{B} , a relation contained in Q^D (because every multi-homomorphism from \mathbb{A} to \mathbb{B} is a multi-homomorphism from \mathbb{C} to \mathbb{D} , whence $f(\mathbf{a}) \subseteq Q^D$ for any $\mathbf{a} \in Q^C$ and any $f \in \text{MuHom}(\mathbb{A}, \mathbb{B})$). ◀

4.2 Complexity classification

Since $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ reduces to $\mathcal{L}\text{-MC}(\mathbb{A})$ (or $\mathcal{L}\text{-MC}(\mathbb{B})$) by the trivial reduction which does not change the input, and the latter problem is clearly in NP, then the former problem is in NP as well. Theorem 9 shows that $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is NP-hard in all the “nontrivial” cases, as in the non-promise setting. However, our proof of hardness requires (in addition to Theorem 7) a much more involved hardness result than in the non-promise case: NP-hardness of c -coloring rainbow k -colorable $2k$ -uniform hypergraphs from [9] (here $c, k \geq 2$).

To state the result in our formalism, we introduce the n -ary “rainbow coloring” and “not-all-equal” relations on a set D as follows.

$$\text{Rb}_D^n = \{\mathbf{d} \in D^n : \{d_1, d_2, \dots, d_n\} = D\}, \quad \text{NAE}_D^n = \{\mathbf{d} \in D^n : \neg(d_1 = d_2 = \dots = d_n)\}$$

In the statement of Theorem 8 and further, we use $(A; S_1, \dots, S_k)$ to denote a structure with universe A and relations S_1, \dots, S_k .

▶ **Theorem 8** (Corollary 1.2 in [9]). *For any A and B of size at least 2, the problem $\{\exists, \wedge\}$ -PMC($(A; \text{Rb}_A^{2|A|}), (B; \text{NAE}_B^{2|A|})$) is NP-complete.*

Given this hardness result, the complexity classification is a simple consequence of Theorem 7.

▶ **Theorem 9** ($\mathcal{L} = \{\exists, \wedge, \vee\}$). *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template. If there is a constant homomorphism from \mathbb{A} to \mathbb{B} , then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is in L (in fact, decidable in constant time), otherwise $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is NP-complete.*

Proof. If there exists a constant homomorphism $f : \mathbb{A} \rightarrow \mathbb{B}$, say with image $\{b\}$, then all the relations $R^{\mathbb{B}}$ in \mathbb{B} contain the constant tuple (b, b, \dots, b) . It follows that every input sentence is satisfied in \mathbb{B} by evaluating the existentially quantified variables to b ; therefore, **Yes** is always a correct output.

If there is no constant homomorphism $\mathbb{A} \rightarrow \mathbb{B}$, we observe that no multi-homomorphism from \mathbb{A} to \mathbb{B} contains a constant homomorphism (as the set of multi-homomorphisms of a PMC template is closed under containment). It follows that the image of any “rainbow” tuple of A under any multi-homomorphism from \mathbb{A} to \mathbb{B} does not contain any constant tuple, and so any multi-homomorphism from \mathbb{A} to \mathbb{B} is a multi-homomorphism from $(A; \text{Rb}_A^{2|A|})$ to $(B; \text{NAE}_B^{2|A|})$. The reduction from Theorem 7 and the hardness from Theorem 8 conclude the proof. ◀

5 Positive fragment

We now turn our attention to the more complex case – the positive equality-free logic, that is, the \mathcal{L} -logic with $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$. We again fix this \mathcal{L} for the entire section.

5.1 Witnesses for quantified formulas

It will be convenient to work with \mathcal{L} -formulas of the special form

$$\phi(x_1, \dots, x_n) = \forall y_1 \exists z_1 \forall y_2 \exists z_2 \dots \forall y_m \exists z_m \phi'(\mathbf{x}, \mathbf{y}, \mathbf{z}), \quad (3)$$

where ϕ' is quantifier-free. Note that each formula is equivalent to a formula in this form (by transforming to prenex normal form and adding dummy quantification as needed) and the conversion can be done in logarithmic space.

Observe that for a structure \mathbb{A} and a tuple $\mathbf{a} \in A^n$, we have $\mathbb{A} \models \phi(\mathbf{a})$ if and only if there exist functions $\alpha_1 : A \rightarrow A$, $\alpha_2 : A^2 \rightarrow A$, \dots , $\alpha_m : A^m \rightarrow A$ which give us evaluations of the existentially quantified variables given the value of the previous universally quantified variables, i.e., these functions satisfy $\mathbb{A} \models \phi'(\mathbf{a}, \mathbf{c}, \alpha_1(c_1), \alpha_2(c_1, c_2), \dots, \alpha_m(c_1, \dots, c_m))$ for every $\mathbf{c} \in A^m$. We call such functions *witnesses* for $\mathbb{A} \models \phi(\mathbf{a})$.

We state a simple consequence of this viewpoint, a version of Lemma 5.

► **Lemma 10.** *Let f be a surjective multi-homomorphism from \mathbb{A} to \mathbb{B} , let $\phi(x_1, \dots, x_n)$ be an \mathcal{L} -formula in the same signature as \mathbb{A} and \mathbb{B} , and let $\mathbf{a} \in A^n$, $\mathbf{b} \in B^n$. If $\mathbb{A} \models \phi(\mathbf{a})$ and $\mathbf{b} \in f(\mathbf{a})$, then $\mathbb{B} \models \phi(\mathbf{b})$.*

In particular, if there exists a surjective multi-homomorphism from \mathbb{A} to \mathbb{B} , and ϕ is an \mathcal{L} -sentence such that $\mathbb{A} \models \phi$, then $\mathbb{B} \models \phi$.

Proof. The claim holds for quantifier-free \mathcal{L} -formulas by Lemma 5.

Next, we assume that ϕ is of the form (3) and select witnesses $\alpha_1, \dots, \alpha_m$ for $\mathbb{A} \models \phi(\mathbf{a})$. Let $g : B \rightarrow A$ be any function such that $b \in f(g(b))$ for every $b \in B$, which exists as f is surjective. We claim that any functions β_1, \dots, β_m such that $\beta_i(b_1, \dots, b_i) \in f(\alpha_i(g(b_1), \dots, g(b_i)))$ for every $i \in [m]$, are witnesses for $\mathbb{B} \models \phi(\mathbf{b})$. Indeed, for all $\mathbf{d} \in B^m$, we have $\mathbb{A} \models \phi'(\mathbf{a}, g(\mathbf{d}), \alpha_1(g(d_1)), \dots, \alpha_m(g(d_1), \dots, g(d_m)))$, and also $\mathbf{b} \in f(\mathbf{a})$, $\mathbf{d} \in f(g(\mathbf{d}))$, and $\beta_i(d_1, \dots, d_i) \in f(\alpha_i(g(d_1), \dots, g(d_i)))$ (by the assumption, choice of g , and choice of β_i , respectively); therefore, $\mathbb{B} \models \phi'(\mathbf{b}, \mathbf{d}, \beta_1(d_1), \dots, \beta_m(d_1, \dots, d_m))$ by the first paragraph. ◀

5.2 Characterization of templates and p- \mathcal{L} -definability

Unlike in the existential case, both characterizations require surjective and multi-valued functions. The core of these characterizations is an adjustment of (2) for surjective homomorphisms.

► **Lemma 11.** *Let \mathbb{A} be a structure with $A = [k]$ and m, n be arbitrary positive integers. Then there exists a formula $\phi(x_{1,1}, \dots, x_{1,n}, x_{2,1}, \dots, \dots, x_{k,n})$ such that, for any structure \mathbb{E} similar to \mathbb{A} with $|E| \leq m$, we have $\mathbb{E} \models \phi(e_{1,1}, \dots, e_{k,n})$ if and only if the mapping $i \mapsto \{e_{i,1}, \dots, e_{i,n}\}$, $i \in [k]$ is contained in a surjective multi-homomorphism from \mathbb{A} to \mathbb{E} .*

Proof. For every function h from $[m]$ to $[k]$ we take a formula $\phi_h(x_{1,1}, \dots, x_{k,n}, z_1, \dots, z_m)$ such that, for any structure \mathbb{E} in the signature of \mathbb{A} , we have $\mathbb{E} \models \phi_h(e_{1,1}, \dots, e_{k,n}, e'_1, \dots, e'_m)$ if and only if the mapping $i \mapsto \{e_{i,1}, \dots, e_{i,n}\} \cup \bigcup_{h(l)=i} e'_l$, $1 \leq i \leq k$, is a multi-homomorphism from \mathbb{A} to \mathbb{E} . Such a formula can be obtained by directly translating the definition of a multi-homomorphism into the language of logic, similarly to (2).

We claim that the formula ϕ obtained by taking the disjunction of ϕ_h over all $h : [m] \rightarrow [k]$ and universally quantifying the variables z_1, \dots, z_m satisfies the requirement of the lemma, provided $|E| \leq m$. Indeed, on the one hand, if $\mathbb{E} \models \phi(e_{1,1}, \dots, e_{k,n})$, then for every evaluation of the z variables, some ϕ_h must be satisfied. We choose any evaluation that covers the whole set E (which is possible since $|E| \leq m$) and the satisfied disjunct ϕ_h then gives us

the required surjective multi-homomorphism from \mathbb{A} to \mathbb{B} (by the choice of ϕ_h). On the other hand, if $i \mapsto \{e_{i,1}, \dots, e_{i,n}\}$ is contained in a surjective multi-homomorphism f , then for any evaluation $\varepsilon_E(z_1), \dots, \varepsilon_E(z_m)$ of the universally quantified variables, a disjunct ϕ_h is satisfied whenever $\varepsilon_E(z_l) \in f(h(l))$ for every $l \in [m]$. Such an h exists since f is surjective. \blacktriangleleft

► **Proposition 12.** *A pair (\mathbb{A}, \mathbb{B}) of similar structures is an \mathcal{L} -PMC template if and only if there exists a surjective multi-homomorphism from \mathbb{A} to \mathbb{B} .*

Proof. For the forward implication, consider the sentence obtained by existentially quantifying all the variables in the formula ϕ provided by Lemma 11 (with $m \geq |A|, |B|$). This sentence is true in \mathbb{A} (as there exists a surjective multi-homomorphism from \mathbb{A} to \mathbb{A} – the identity), so it must be true in \mathbb{B} , giving us a surjective multi-homomorphism from \mathbb{A} to \mathbb{B} . The backward implication follows from Lemma 10. \blacktriangleleft

An example which shows that one cannot replace in Proposition 12 “surjective multi-homomorphism” by “(multi-)homomorphism” is the input formula $\varphi = \forall x \exists y R(x, y)$ (“there are no sinks”) for a template where \mathbb{A} is a digraph with no sinks and \mathbb{B} is, say, \mathbb{A} plus an isolated vertex.

The following characterization of promise definability is also a straightforward consequence of Lemmata 10 and 11.

► **Theorem 13.** *Let (\mathbb{A}, \mathbb{B}) and (\mathbb{C}, \mathbb{D}) be \mathcal{L} -PMC templates such that $A = C$ and $B = D$. Then (\mathbb{C}, \mathbb{D}) is p - \mathcal{L} -definable from (\mathbb{A}, \mathbb{B}) if and only if $\text{SMuHom}(\mathbb{A}, \mathbb{B}) \subseteq \text{SMuHom}(\mathbb{C}, \mathbb{D})$. Moreover, in such a case, $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$.*

Proof. The theorem is proved in the same way as Theorem 7; using Lemma 10 instead of Lemma 5 for the forward implication, and the formula provided by Lemma 11 instead of (2) for the backward implication. \blacktriangleleft

5.3 Membership

Clearly, every \mathcal{L} -MC, as well as \mathcal{L} -PMC, is in PSPACE. We now give a generalization of the remaining membership results from [12] using an appropriate generalization of “A-shops” and “E-shops” from that paper. We say that a surjective multi-homomorphism f from \mathbb{A} to \mathbb{B} is an A-smuhom if there exists $a^* \in A$ such that $f(a^*) = B$. We also say that (\mathbb{A}, \mathbb{B}) admits an A-smuhom in such a case. We call f an E-smuhom if $f^{-1}(b^*) = A$ for some $b^* \in B$. Finally, we call f an AE-smuhom if it is simultaneously an A-smuhom and an E-smuhom.

An additional simple reduction will be useful in the proof of the membership result (Theorem 15) and later as well. We say that an \mathcal{L} -PMC template (\mathbb{C}, \mathbb{D}) is a *relaxation* of an \mathcal{L} -PMC template (\mathbb{A}, \mathbb{B}) if (\mathbb{C}, \mathbb{A}) and (\mathbb{B}, \mathbb{D}) are \mathcal{L} -PMC templates. Recall that, by Proposition 12, the property is equivalent to the existence of surjective multi-homomorphisms from \mathbb{C} to \mathbb{A} and from \mathbb{B} to \mathbb{D} .

► **Proposition 14.** *Let (\mathbb{A}, \mathbb{B}) and (\mathbb{C}, \mathbb{D}) be \mathcal{L} -PMC templates. If (\mathbb{C}, \mathbb{D}) is a relaxation of (\mathbb{A}, \mathbb{B}) , then $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$.*

Proof. The trivial reduction, which does not change the input, works. Indeed, **Yes**-instances of $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D})$ are **Yes**-instances of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ since (\mathbb{C}, \mathbb{A}) is an \mathcal{L} -PMC template, and **No**-instances of $\mathcal{L}\text{-PMC}(\mathbb{C}, \mathbb{D})$ are **No**-instances of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ since (\mathbb{B}, \mathbb{D}) is an \mathcal{L} -PMC template. \blacktriangleleft

► **Theorem 15.** *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template. Then the following holds.*

1. *If (\mathbb{A}, \mathbb{B}) admits an \mathbb{A} -smuhom, then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is in NP.*
2. *If (\mathbb{A}, \mathbb{B}) admits an \mathbb{E} -smuhom, then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is in coNP.*
3. *If (\mathbb{A}, \mathbb{B}) admits an \mathbb{AE} -smuhom, then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is in L.*

Proof. For the first item, let f be an \mathbb{A} -smuhom from \mathbb{A} to \mathbb{B} with $f(a^*) = B$, and consider an input ϕ in the special form (3), i.e., $\phi = \forall y_1 \exists z_1 \forall y_2 \exists z_2 \dots \forall y_m \exists z_m \phi'(\mathbf{y}, \mathbf{z})$, where ϕ' is quantifier-free. We answer **Yes** if there exists $\mathbf{a} \in A^m$ such that $\mathbb{A} \models \phi'(a^*, a^*, \dots, a^*, \mathbf{a})$; this can be clearly decided in NP. It is clear that the answer is **Yes** whenever ϕ is a **Yes**-instance of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$. On the other hand, if $\mathbb{A} \models \phi'(a^*, \dots, a^*, \mathbf{a})$, then any functions $\beta_1 : B \rightarrow B$, \dots , $\beta_m : B^m \rightarrow B$ such that $\beta_i(b_1, \dots, b_i) \in f(a_i)$ (for all $i \in [m]$ and $b_1, \dots, b_m \in B$) provide witnesses for $\mathbb{B} \models \phi$ by Lemma 5. Therefore, if ϕ is a **No**-instance, then the answer is **No**, as needed.

The second item follows by the duality argument.

In the case $\mathbb{A} = \mathbb{B}$, the third item can be proved in an analogous way (by eliminating both quantifiers instead of just one), see Corollary 9 in [12]. For the general case, we will construct \mathbb{C} such that there is an \mathbb{AE} -smuhom from \mathbb{C} to \mathbb{C} and there are surjective multi-homomorphisms from \mathbb{A} to \mathbb{C} and from \mathbb{C} to \mathbb{B} . Then (\mathbb{A}, \mathbb{B}) will be a relaxation of (\mathbb{C}, \mathbb{C}) by Proposition 12, and then membership of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ in L will follow from Proposition 14 and the mentioned Corollary 9 in [12]. Let f be an \mathbb{AE} -smuhom from \mathbb{A} to \mathbb{B} with $f(a^*) = B$ and $f^{-1}(b^*) = A$, and define a surjective multi-valued function f' from A to B by $f'(a^*) = B$ and $f'(a) = \{b^*\}$ if $a \neq a^*$. Note that f' is contained in f , so f' is a surjective multi-homomorphism from \mathbb{A} to \mathbb{B} . We define \mathbb{C} as the “image” of \mathbb{A} under f' , that is, $C = B$ and $R^C = \cup_{\mathbf{a} \in R^A} f'(\mathbf{a})$ for each relation symbol R . Clearly, f' is a surjective multi-homomorphism from \mathbb{A} to \mathbb{C} and the identity is a surjective homomorphism from \mathbb{C} to \mathbb{B} . It remains to find an \mathbb{AE} -smuhom from \mathbb{C} to \mathbb{C} . We claim that g defined by $g(b^*) = \{b^*\}$ and $g(c) = C$ for $c \neq b^*$ is such an \mathbb{AE} -smuhom. Indeed, if $\mathbf{c} \in R^C$, then $\mathbf{c} \in f'(\mathbf{a})$ for some $\mathbf{a} \in R^A$. By the definition of f' , we necessarily have $a_i = a^*$ whenever $c_i \neq b^*$; therefore, $f'(\mathbf{a}) \supseteq g(\mathbf{c})$. But $f'(\mathbf{a}) \subseteq R^C$ as $f' \in \text{SMuHom}(\mathbb{A}, \mathbb{C})$, and we are done. ◀

These membership results together with the (more involved) hardness results were sufficient for the tetrachotomy in [12]. One problem with generalizing this tetrachotomy is that, unlike in the non-promise setting, an \mathcal{L} -PMC template can admit an \mathbb{A} -smuhom and an \mathbb{E} -smuhom, but no \mathbb{AE} -smuhom. However, such a situation cannot happen for digraphs.

► **Proposition 16.** *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template such that \mathbb{A} and \mathbb{B} are digraphs. If (\mathbb{A}, \mathbb{B}) admits an \mathbb{A} -smuhom and an \mathbb{E} -smuhom, then it admits an \mathbb{AE} -smuhom.*

Proof. See Appendix A. ◀

5.4 Hardness

As a consequence of Theorems 8 and 13, we obtain the following hardness result.

► **Theorem 17.** *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template.*

1. *If there is no \mathbb{E} -smuhom from \mathbb{A} to \mathbb{B} , then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is NP-hard.*
2. *If there is no \mathbb{A} -smuhom from \mathbb{A} to \mathbb{B} , then $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$ is coNP-hard.*

Proof. If there exists no \mathbb{E} -smuhom from \mathbb{A} to \mathbb{B} , then $\text{SMuHom}(\mathbb{A}, \mathbb{B})$ is contained in $\text{SMuHom}((A; \text{Rb}_A^{2|A|}), (B; \text{NAE}_B^{2|A|}))$. Theorem 8 and Theorem 13 then imply the first item. The second item follows by the duality argument. ◀

In the non-promise setting, the absence of A -smuhoms and E -smuhoms is sufficient for PSPACE-hardness [12, 13]. This most involved part of the tetrachotomy result seems much more challenging in the promise setting and we do not have strong reasons to believe that templates without A -smuhoms and E -smuhoms are necessarily PSPACE-hard. Nevertheless, we are able to prove some additional hardness results which will cover all the extensions of \mathcal{L} .

► **Proposition 18.** $\mathcal{L}\text{-PMC}((A; =_A), (B; =_B))$ is PSPACE-hard for any A, B such that $|A| \geq |B| \geq 2$.

Note here that surjective multi-homomorphisms from $(A; =_A)$ to $(B; =_B)$ are exactly the surjective multi-valued functions from A to B of multiplicity one. In particular, if $|A| < |B|$, then $((A; =_A), (B; =_B))$ is not an \mathcal{L} -PMC template.

Proof. We start by noticing that the template $((A; =_A), ([2]; =_{[2]}))$ is a relaxation of $(\mathbb{A}, \mathbb{B}) := ((A; =_A), (B; =_B))$. So by Proposition 14, it is enough to prove the claim in the case $B = [2]$. For simplicity, we assume that $A = [k]$ ($k \geq 2$). We prove the PSPACE-hardness by a reduction from $\mathcal{L}\text{-MC}(\mathbb{B})$, a PSPACE-hard problem by, e.g., [14]. Consider an input ϕ to $\mathcal{L}\text{-MC}(\mathbb{B})$ in the special form (3), i.e., $\phi = \forall y_1 \exists z_1 \forall y_2 \exists z_2 \dots \forall y_m \exists z_m \phi'(\mathbf{y}, \mathbf{z})$, where ϕ' is quantifier-free. We need to find a log-space computable formula ψ such that $\mathbb{B} \models \phi$ implies $\mathbb{A} \models \psi$ (so that **Yes**-instances of $\text{MC}(\mathbb{B})$ are transformed to **Yes**-instances of $\mathcal{L}\text{-PMC}(\mathbb{A}, \mathbb{B})$) and $\mathbb{B} \models \psi$ implies $\mathbb{B} \models \phi$ (so that **No**-instances are transformed to **No**-instances).

The rough idea to construct ψ is to reinterpret the values in $A = [k]$ as values in $B = [2]$ via a mapping $A \rightarrow B$. We set

$$\psi = \forall x_1 \forall x_2 \exists x_3 \exists x_4 \dots \exists x_k (x_1 = x_2) \vee \bigwedge_{f:A \rightarrow B} \rho_f, \quad \text{where} \quad (4)$$

$$\rho_f = (\forall y'_1 \exists z_1 \dots \forall y'_m \exists z_m) (\exists y_1 \dots \exists y_m) \left(\bigwedge_{i=1}^m \sigma[f, y'_i, y_i] \right) \wedge \phi'(\mathbf{y}, \mathbf{z}) \quad (5)$$

$$\sigma[f, y'_i, y_i] = \bigvee_{a \in A} ((y'_i = x_a) \wedge (y_i = x_{f(a)})) \quad (6)$$

Observe first that ψ can be constructed from ϕ in logarithmic space.

Next, we verify that $\mathbb{B} \models \psi$ implies $\mathbb{B} \models \phi$. So, we suppose $\mathbb{B} \models \psi$ and aim to find witnesses β_1, \dots, β_m for $\mathbb{B} \models \phi$; to this end, let \mathbf{c} be some tuple in B^m that corresponds to evaluations of universally quantified variables in ϕ . We evaluate the variables x_1 and x_2 in ψ as $\varepsilon_B(x_1) = 1$ and $\varepsilon_B(x_2) = 2$, and pick an evaluation $\varepsilon_B(x_3), \dots, \varepsilon_B(x_k)$ making ψ true in \mathbb{B} . Set $f(a) = \varepsilon_B(x_a)$, $a \in A$. The first disjunct of (4) is not satisfied, so ρ_f is satisfied with this choice of ε_B . When it is the turn to evaluate y'_i , we set $\varepsilon_B(y'_i) = c_i$ and define $\beta_i(c_1, \dots, c_i) = \varepsilon_B(z_i)$, where $\varepsilon_B(z_i)$ is a satisfactory evaluation of z_i . Inspecting the definition (6), we see that y_1, \dots, y_m are necessarily evaluated as $\varepsilon_B(y_1) = c_1, \dots, \varepsilon_B(y_m) = c_m$: indeed, if a disjunct $(y'_i = x_a) \wedge (y_i = x_{f(a)})$ is satisfied, then $c_i = \varepsilon_B(y'_i) = \varepsilon_B(x_a)$ and $\varepsilon_B(y_i) = \varepsilon_B(x_{f(a)}) = \varepsilon_B(x_{\varepsilon_B(x_a)}) = \varepsilon_B(x_a)$; in particular, $\varepsilon_B(y_i) = c_i$. Therefore, the conjunct $\phi'(\mathbf{y}, \mathbf{z})$ in (5) ensures $\mathbb{B} \models \phi'(\mathbf{c}, \beta_1(c_1), \dots, \beta_m(c_1, \dots, c_m))$. As \mathbf{c} was chosen arbitrarily, we get that β_1, \dots, β_m are witnesses for $\mathbb{B} \models \phi$, as required.

We now suppose that β_1, \dots, β_m are witnesses for $\mathbb{B} \models \phi$, and aim to show that $\mathbb{A} \models \psi$. Because of the first disjunct of (4), it is enough to consider only evaluations of x_1 and x_2 with $\varepsilon_A(x_1) \neq \varepsilon_A(x_2)$. Since any bijection, regarded as a surjective multi-homomorphism from \mathbb{A} to \mathbb{A} of multiplicity one, preserves \mathcal{L} -formulas (in the sense of Lemma 10), then we

can as well assume that $\varepsilon_A(x_1) = 1$ and $\varepsilon_A(x_2) = 2$. We evaluate the remaining x variables as $\varepsilon_A(x_a) = a$, $a = 3, 4, \dots, k$. We take a function $f : A \rightarrow B$ and argue that ρ_f is satisfied in \mathbb{A} . Given a selection of $\varepsilon_A(y'_i)$, we evaluate z_i as $\varepsilon_A(z_i) = \beta_i(f(\varepsilon_A(y'_1)), \dots, f(\varepsilon_A(y'_i)))$, and we define the evaluation of the remaining variables by $\varepsilon_A(y_i) = f(\varepsilon_A(y'_i))$. With these choices, each $\sigma[f, y'_i, y_i]$ is satisfied because of the disjunct $a = \varepsilon_A(y'_i)$ in (6). The second conjunct in (5), $\phi'(\mathbf{y}, \mathbf{z})$, is also satisfied: we know $\mathbb{B} \models \phi'(\mathbf{c}, \beta_1(c_1), \dots, \beta_m(c_1, \dots, c_m))$ in particular for $c_1 = f(\varepsilon_A(y'_1)), \dots, c_m = f(\varepsilon_A(y'_m))$ and, with this \mathbf{c} , it is apparent from the choice of evaluations that $\mathbb{B} \models \phi'(\mathbf{c}, \beta_1(c_1), \dots, \beta_m(c_1, \dots, c_m))$ is equivalent to $\mathbb{A} \models \phi'(\varepsilon_A(y_1), \dots, \varepsilon_A(y_m), \varepsilon_A(z_1), \dots, \varepsilon_A(z_m))$. The proof of $\mathbb{A} \models \psi$ is concluded. \blacktriangleleft

It follows that $\{\exists, \forall, \wedge, \vee, =\}$ -PMC over any template is PSPACE-hard and so is, by the duality argument, $\{\exists, \forall, \wedge, \vee, \neq\}$ -PMC. The next proposition implies PSPACE-hardness for $\{\exists, \forall, \wedge, \vee, \neg\}$ -PMC.

► Proposition 19. *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template which is closed under complementation. Then \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}) is PSPACE-hard.*

Proof. Suppose that (\mathbb{A}, \mathbb{B}) is closed under complementation. We define an equivalence relation \sim_A on A by considering two elements equivalent if they play the same role in every relation of \mathbb{A} . Formally, $a \sim a'$ if for every symbol R from the signature, every coordinate $i \in [\text{ar}(R)]$, and every $\mathbf{c}, \mathbf{c}' \in A^{\text{ar}(R)}$, if $c_i = a$, $c'_i = a'$, $c_j = c'_j$ for all $j \in [\text{ar}(R)] \setminus \{i\}$, and $\mathbf{c} \in R^{\mathbb{A}}$, then $\mathbf{c}' \in R^{\mathbb{A}}$. We define an equivalence relation \sim_B on B analogously. Notice that \sim_A (resp., \sim_B) is indeed an equivalence relation; let m and n denote the number of equivalence classes of \sim_A and \sim_B , respectively.

Observe that $m, n \geq 2$. Indeed, otherwise any nonempty relation in the corresponding template contains all the tuples, and we do not allow such structures in this paper.

Let $\mathbb{C} = (A; \sim_A)$ and $\mathbb{D} = (B; \sim_B)$. We claim that every surjective multi-homomorphism f from \mathbb{A} to \mathbb{B} preserves \sim , i.e., is a surjective multi-homomorphism from \mathbb{C} to \mathbb{D} . Consider $a, a' \in A$, and $b, b' \in B$ such that $a \sim_A a'$, $b \in f(a)$, and $b' \in f(a')$. In order to prove $b \sim_B b'$, take arbitrary $R, i, \mathbf{d}, \mathbf{d}'$ such that $d_i = b$, $d'_i = b'$, $d_j = d'_j$ for all $j \neq i$, and $\mathbf{d} \in R^{\mathbb{B}}$. Let $\mathbf{c}, \mathbf{c}' \in A^{\text{ar}(R)}$ be tuples such that $c_i = a$, $c'_i = a'$, and $c_j = c'_j \in f^{-1}(d_j)$ for all $j \neq i$ (which exist as f is surjective). If $\mathbf{c} \notin R^{\mathbb{A}}$, then $\mathbf{c} \in \overline{R^{\mathbb{A}}}$ and, consequently, $\mathbf{d} \in f(\mathbf{c}) \subseteq \overline{R^{\mathbb{B}}}$ (as f is a surjective multi-homomorphism from \mathbb{A} to \mathbb{B}), a contradiction with $\mathbf{d} \in R^{\mathbb{B}}$. Therefore, $\mathbf{c} \in R^{\mathbb{A}}$ and also $\mathbf{c}' \in R^{\mathbb{A}}$ as $a \sim_A a'$. Now $\mathbf{d}' \in f(\mathbf{c}') \subseteq R^{\mathbb{B}}$, and $b \sim_B b'$ follows.

By Theorem 13, \mathcal{L} -PMC(\mathbb{C}, \mathbb{D}) \leq \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}). Since there exists a surjective multi-valued function from A to B that preserves \sim (namely, any $f \in \text{SMuHom}(\mathbb{A}, \mathbb{B})$), we also know that $m \geq n$. The template $(\mathbb{E}, \mathbb{F}) := (([m]; =_{[m]}), ([n]; =_{[n]}))$ is a relaxation of (\mathbb{C}, \mathbb{D}) , because there exists a surjective multi-homomorphism from \mathbb{E} to \mathbb{C} (a multi-valued function that maps i to the i -th equivalence class of \sim_A under an arbitrary linear ordering of classes) and a surjective multi-homomorphism from \mathbb{D} to \mathbb{F} (a “multi”-valued function that maps every element in the i -th equivalence class of \sim_B to $\{i\}$). By Proposition 14, \mathcal{L} -PMC(\mathbb{E}, \mathbb{F}) \leq \mathcal{L} -PMC(\mathbb{C}, \mathbb{D}); therefore, \mathcal{L} -PMC(\mathbb{E}, \mathbb{F}) \leq \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}). The former \mathcal{L} -PMC is PSPACE-hard by Proposition 18, so \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}) is PSPACE-hard, too. \blacktriangleleft

5.5 Summary and examples

The claims stated in Table 2 are now immediate consequences of the obtained results. Note that the claims remain true without the imposed restrictions on structures (i.e., we can allow singleton universes, nullary relations, etc.); the only nontrivial ingredient is the L-membership of the Boolean Sentence Value Problem [10].

We observe that the results imply a complete complexity classification in the case that one of the two template structures is *Boolean*, i.e., has a two-element universe.

► **Corollary 20** ($\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$). *Let (\mathbb{A}, \mathbb{B}) be an \mathcal{L} -PMC template.*

1. *If \mathbb{B} is Boolean, then \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}) is in L, or is NP-complete, or PSPACE-complete.*
2. *If \mathbb{A} is Boolean, then \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}) is in L, or is coNP-complete, or PSPACE-complete.*
3. *If \mathbb{A} and \mathbb{B} are Boolean, then \mathcal{L} -PMC(\mathbb{A}, \mathbb{B}) is in L, or is PSPACE-complete.*

Proof. If \mathbb{B} is Boolean, then every \mathbb{E} -smuhom (from \mathbb{A} to \mathbb{B}) is an \mathbb{AE} -smuhom. Moreover, if there is no \mathbb{A} -smuhom, then every surjective multi-homomorphism is of multiplicity one, so it is also a multi-homomorphism from $(A; =_A)$ to $(B; =_B)$. The first item now follows from Proposition 18 and Theorem 13. The other items are easy as well. ◀

There are two wide gaps left for further investigation. First, it is unclear what the complexity is for the \mathcal{L} -PMC over templates that admit both an \mathbb{A} -smuhom and an \mathbb{E} -smuhom, but no \mathbb{AE} -smuhom. While there is no such a digraph template, there are examples with one ternary or two binary relations, e.g., the following. We use ij as a shortcut for the pair (i, j) .

$$\begin{aligned} \mathbb{A} &= ([3]; \{(1, 2, 3)\}), & \mathbb{B} &= ([3]; \{1, 2, 3\} \times \{2\} \times \{3\} \cup \{1, 2\} \times \{2\} \times \{2, 3\}) \\ \mathbb{A} &= ([3]; \{12\}, \{13\}), & \mathbb{B} &= ([3]; \{12, 22, 32\}, \{12, 13, 22, 23, 33\}) \end{aligned}$$

The second gap is between simultaneous NP- and coNP-hardness, and PSPACE-hardness, when the template admits neither an \mathbb{A} -smuhom nor an \mathbb{E} -smuhom. Examples with unknown complexity include the following.

$$\begin{aligned} \mathbb{A} &= ([3]; \{(1, 2, 3)\}), & \mathbb{B} &= ([3]; \{2, 3\} \times \{1, 3\} \times \{1, 2\}) \\ \mathbb{A} &= ([3]; \{(1, 2, 3)\}), & \mathbb{B} &= ([3]; \{1, 2\} \times \{1, 2\} \times \{3\} \cup \{1, 3\} \times \{2\} \times \{2\}) \\ \mathbb{A} &= ([4]; \{12, 34\}), & \mathbb{B} &= ([4]; \{12, 13, 14, 23, 24, 34, 32\}) \end{aligned}$$

In an ongoing work, we have developed some more general PSPACE-hardness criteria, but the examples above remain elusive. The following equivalent *unary* version of the first example is an especially interesting template, whose \mathcal{L} -PMC is the problem described in the introduction.

$$\mathbb{A} = ([3]; \{1\}, \{2\}, \{3\}), \quad \mathbb{B} = ([3]; \{2, 3\}, \{1, 3\}, \{1, 2\})$$

6 Conclusion

We gave a full complexity classification of $\{\exists, \wedge, \vee\}$ -PMC, initiated an algebraic approach to $\{\exists, \forall, \wedge, \vee\}$ -PMC, and applied it to provide several complexity results about this class of problems.

An interesting concrete problem, whose complexity is currently open, is the $\{\exists, \forall, \wedge, \vee\}$ -PMC over the unary template above. As for the theory-building, the next natural step is to capture more complex reductions by means of surjective multi-homomorphisms; namely, the analogue of pp-constructions, which proved to be so useful in the theory of (Promise) CSPs [3, 2]. It may be also helpful to characterize and study the sets of surjective multi-homomorphisms in the spirit of [15, 7].

References

- 1 Per Austrin, Venkatesan Guruswami, and Johan Håstad. $(2 + \epsilon)$ -Sat is NP-hard. *SIAM J. Comput.*, 46(5):1554–1573, 2017. doi:10.1137/15M1006507.
- 2 Libor Barto, Jakub Bulín, Andrei A. Krokhin, and Jakub Opršal. Algebraic approach to promise constraint satisfaction. *J. ACM*, 68(4):28:1–28:66, 2021. doi:10.1145/3457606.
- 3 Libor Barto, Andrei Krokhin, and Ross Willard. Polymorphisms, and How to Use Them. In Andrei Krokhin and Stanislav Živný, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 1–44. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017. doi:10.4230/DFU.Vo17.15301.1.
- 4 Ferdinand Börner. Basics of Galois Connections. In Nadia Creignou, Phokion G. Kolaitis, and Heribert Vollmer, editors, *Complexity of Constraints - An Overview of Current Research Themes [Result of a Dagstuhl Seminar]*, volume 5250 of *Lecture Notes in Computer Science*, pages 38–67. Springer, 2008. doi:10.1007/978-3-540-92800-3_3.
- 5 Joshua Brakensiek and Venkatesan Guruswami. Promise Constraint Satisfaction: Structure Theory and a Symmetric Boolean Dichotomy. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'18*, pages 1782–1801, Philadelphia, PA, USA, 2018. Society for Industrial and Applied Mathematics. doi:10.1137/1.9781611975031.117.
- 6 A. A. Bulatov. A dichotomy theorem for nonuniform CSPs. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 319–330, October 2017. doi:10.1109/FOCS.2017.37.
- 7 Catarina Carvalho and Barnaby Martin. The lattice and semigroup structure of multipermutations. *International Journal of Algebra and Computation*, 0(0):1–25, 2021. doi:10.1142/S0218196722500096.
- 8 Hubie Chen. Meditations on quantified constraint satisfaction. In Robert L. Constable and Alexandra Silva, editors, *Logic and Program Semantics - Essays Dedicated to Dexter Kozen on the Occasion of His 60th Birthday*, volume 7230 of *Lecture Notes in Computer Science*, pages 35–49. Springer, 2012. doi:10.1007/978-3-642-29485-3_4.
- 9 Venkatesan Guruswami and Euiwoong Lee. Strong inapproximability results on balanced rainbow-colorable hypergraphs. *Comb.*, 38(3):547–599, 2018. doi:10.1007/s00493-016-3383-0.
- 10 Nancy Lynch. Log space recognition and translation of parenthesis languages. *J. ACM*, 24(4):583–590, October 1977. doi:10.1145/322033.322037.
- 11 Florent Madelaine and Barnaby Martin. The complexity of positive first-order logic without equality. *ACM Trans. Comput. Logic*, 13(1), January 2012. doi:10.1145/2071368.2071373.
- 12 Florent R. Madelaine and Barnaby Martin. A tetrachotomy for positive first-order logic without equality. In *Proceedings of the 26th Annual IEEE Symposium on Logic in Computer Science, LICS 2011, June 21-24, 2011, Toronto, Ontario, Canada*, pages 311–320. IEEE Computer Society, 2011. doi:10.1109/LICS.2011.27.
- 13 Florent R. Madelaine and Barnaby Martin. On the complexity of the model checking problem. *SIAM J. Comput.*, 47(3):769–797, 2018. doi:10.1137/140965715.
- 14 Barnaby Martin. First-order model checking problems parameterized by the model. In Arnold Beckmann, Costas Dimitracopoulos, and Benedikt Löwe, editors, *Logic and Theory of Algorithms, 4th Conference on Computability in Europe, CiE 2008, Athens, Greece, June 15-20, 2008, Proceedings*, volume 5028 of *Lecture Notes in Computer Science*, pages 417–427. Springer, 2008. doi:10.1007/978-3-540-69407-6_45.
- 15 Barnaby Martin. The lattice structure of sets of surjective hyper-operations. In David Cohen, editor, *Principles and Practice of Constraint Programming - CP 2010 - 16th International Conference, CP 2010, St. Andrews, Scotland, UK, September 6-10, 2010. Proceedings*, volume 6308 of *Lecture Notes in Computer Science*, pages 368–382. Springer, 2010. doi:10.1007/978-3-642-15396-9_31.
- 16 Barnaby Martin. Quantified Constraints in Twenty Seventeen. In Andrei Krokhin and Stanislav Živný, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 327–346. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017. doi:10.4230/DFU.Vo17.15301.327.

- 17 Barnaby Martin and Jos Martin. The complexity of positive first-order logic without equality II: the four-element case. In Anuj Dawar and Helmut Veith, editors, *Computer Science Logic, 24th International Workshop, CSL 2010, 19th Annual Conference of the EACSL, Brno, Czech Republic, August 23-27, 2010. Proceedings*, volume 6247 of *Lecture Notes in Computer Science*, pages 426–438. Springer, 2010. doi:10.1007/978-3-642-15205-4_33.
- 18 Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM*, 67(5):30:1–30:78, August 2020. doi:10.1145/3402029.
- 19 Dmitriy Zhuk and Barnaby Martin. QCSP monsters and the demise of the chen conjecture. In Konstantin Makarychev, Yury Makarychev, Madhur Tulsiani, Gautam Kamath, and Julia Chuzhoy, editors, *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing, STOC 2020, Chicago, IL, USA, June 22-26, 2020*, pages 91–104. ACM, 2020. doi:10.1145/3357713.3384232.

A Proof of Proposition 16

Proof. Denote by R the unique binary symbol in the signature. Let f be an \mathbb{A} -smuhom from \mathbb{A} to \mathbb{B} with $f(a^*) = B$ and let g be an \mathbb{E} -smuhom from \mathbb{A} to \mathbb{B} with $g^{-1}(b^*) = A$.

If a^* is isolated in \mathbb{A} (i.e., $(a, a^*), (a^*, a) \notin R^{\mathbb{A}}$ for every $a \in A$), then we define a surjective multi-valued function h by $h(a^*) = B$ and $h(a) = \{b^*\}$ for every $a \neq a^*$. It is a multi-homomorphism from \mathbb{A} to \mathbb{B} since for any $(a, a') \in R^{\mathbb{A}}$, we have $h(a, a') = \{(b^*, b^*)\}$, which is contained in $R^{\mathbb{B}}$ because $R^{\mathbb{A}}$ is nonempty, so $g(R^{\mathbb{A}}) \ni (b^*, b^*)$.

Suppose next that there is an edge $(a_1, a^*) \in R^{\mathbb{A}}$ but a^* has no outgoing edges in \mathbb{A} . Let b_1 be an arbitrary element from $f(a_1)$ and define h by $h(a^*) = B$ and $h(a) = \{b_1\}$ for every $a \neq a^*$. To verify that $h \in \text{SMuHom}(\mathbb{A}, \mathbb{B})$, consider an edge $(a, a') \in R^{\mathbb{A}}$. As a^* has no outgoing edges in \mathbb{A} , we get $a \neq a^*$, so $h(a) = \{b_1\}$. Now $h(a, a') \subseteq \{b_1\} \times B$, which is contained in $R^{\mathbb{B}}$ because $R^{\mathbb{B}} \supseteq f(a_1, a^*) \supseteq \{b_1\} \times B$.

If a^* has an outgoing edge $(a^*, a_1) \in R^{\mathbb{A}}$ but no incoming edges, we proceed similarly, defining $h(a^*) = B$ and $h(a) = \{b_1\}$ for all $a \neq a^*$, where b_1 is an arbitrary element from $f(a_1)$.

Finally, suppose that $(a_1, a^*) \in R^{\mathbb{A}}$ and $(a^*, a_2) \in R^{\mathbb{A}}$ for some $a_1, a_2 \in A$. If there is an element $a_3 \in A$ with no outgoing (resp., incoming) edges, define h by $h(a_3) = B$ and $h(a) = \{b'\}$ for all $a \neq a_3$, where b' is an arbitrary element from $f(a_1)$ (resp., $f(a_2)$). If there is no such element a_3 , then we define $h(a^*) = B$ and $h(a) = \{b^*\}$ for all $a \neq a^*$. Since g is surjective, and every $a \in A$ has both an incoming and an outgoing edge, then $(b, b^*) \in R^{\mathbb{B}}$ and $(b^*, b) \in R^{\mathbb{B}}$ for all $b \in B$, therefore, $h \in \text{SMuHom}(\mathbb{A}, \mathbb{B})$.

The proof of Proposition 16 is concluded. ◀