FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

DOCTORAL THESIS

Kristina Asimi

Promises in Satisfaction Problems

Department of Algebra

Supervisor of the doctoral thesis: doc. Mgr. Libor Barto, Ph.D.

Study programme: Algebra, Number Theory and Mathematical Logic

Prague 2023

I declare that I carried out this doctoral thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....
Author's signature

Dedication: Inspired by recent acknowledgments sections of my colleagues and friends, I wanted to make a nice, funny, emotional acknowledgments/dedication section myself, but unfortunately, as my friends know, I am not good in writing and expressing my emotions. In fact, I am known for replying only with *yes* or *no* or not replying at all, also known in our community as *asimiing*[1]. However, I *promise* to try my best to elaborate.

I would like to express my sincere gratitude and appreciation to everybody who has been a part of my PhD journey, however, in order to keep this section shorter than the actual thesis, I will mention just some persons individually.

I am deeply grateful to my supervisor, Libor Barto, for his invaluable guidance, support, and encouragement throughout my PhD journey. Thank you, Libor, for your insightful comments, constructive feedback, patience; for all the beers we drank, all the songs we sang and all the foosball games we played. I will still remember you *when you're* 64.

I am also very grateful to Victor Dalmau, for having me over in Barcelona and for being so kind. Thank you, Victor, for building monster structures with me, and for all the funny stories.

I would like to thank my MSc supervisor, Jovanka Pantović, for encouraging me to apply for this PhD position and for introducing me to the *wonderland* of CSP, during my studies back in Novi Sad. Thank you, Vanja, for believing in me and for sending me to this wonderful journey, where, besides PCSP, I also got to know a lot of amazing people.

The first time I walked into the PhD office of the Department of Algebra, I met my PhD father Libor and my PhD brother Diego. Diego offered me liquorice candy, I took one and regretted it later. That was the only time he gave me something bitter. Since then, he was only making my bitterness a little bit sweeter.

I would like to express my sincere appreciation to all the other CSP colleagues who made my PhD journey better. Thank you, Michael, Kuba, Alex, Dima, Attila, for all the coffees, beers, movie nights, amazing trips and board games nights. Thank you, William, for listening to my first results and for telling me I was doing a great job, even though I was not so sure. Thank you, Silvia, for being a great friend and coauthor, for coming to Prague when I was stuck after Covid, both emotionally and research-wise, I did not even know I needed you. Thank you, Caterina, for your support and encouragement in all spheres of my life. Thank you, Albert and Jakub, for popping up in Prague every now and then, and for following my crazy trip ideas. Thank you, Pinsker, for all the songs.

I would also like to thank my other algebra friends. Thank you, Filippo and Chiara, for filling the PhD office with joy and happiness. Thank you, Mikuláš, for being my bro.

I also have to thank my biology friends, for being interested in my research topics, for having me in their flat and accepting me in their lovely family. Especially thanks to Sanja, who was there since the beginning of my journey. Thank you, Sanja, for your friendship, love, kindness, generosity, loyalty, encouragement, for always believing in me, even when I felt small; thank you for all the memories we have shared, the laughter we enjoyed, and the tears we have shed together.

[1] *Asimiing* is an example of *pinskerism*, for more information on this topic please ask me in person.

Finally, I would like to express my gratitude to my support from Serbia - to my friends, who were listening to all my thoughts and feelings, and who would always welcome me with such a joy; and to my family - Mom, Dad, brother Minja and sister Katarina, for their unconditional love, unwavering support, and constant encouragement throughout my academic journey. Their faith in me has been a source of strength and inspiration, and I am forever grateful for their presence in my life. Hvala, mama, tata, Minja i Juco, što ste uvek bili tu za mene, što sam se posle svakog našeg susreta osećala bolje, to mi je dalo snagu da nastavim. I am also grateful to our dogs, who helped me through the most stressful periods.

I would like to extend my sincere apologies to those who have supported me throughout my journey but have not been mentioned explicitly. Please know that your support has been deeply appreciated, and I apologize for any unintentional omissions.

There, *I did it my way.*

Title: Promises in Satisfaction Problems

Author: Kristina Asimi

Department: Department of Algebra

Supervisor: doc. Mgr. Libor Barto, Ph.D., Department of Algebra

Abstract: This thesis focuses on the complexity of the promise version of Constraint Satisfaction Problem (CSP) and its variants.

The first study concerns the Promise Constraint Satisfaction Problem (PCSP), which extends the traditional CSP to include approximation variants of satisfiability and graph coloring. A specific PCSP, referred to as finding a valid Not-All-Equal solution to a 1-in-3-SAT instance, has been shown by Barto [LICS '19] to lack finite tractability. While it can be reduced to a tractable CSP, the latter is necessarily over an infinite domain (unless P=NP). We say that such a PCSP is not finitely tractable and we initiate a systematic study of this phenomenon by giving a general necessary condition for finite tractability. Additionally, we characterize finite tractability within a class of templates.

In the second study, we focus on the CSP in the context of first-order logic. The fixed-template 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 (making the promise version). 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 call these problems Promise Model Checking Problems, and they are a generalization of Model Checking Problems. We classify the computational complexity of Promise Model Checking 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, $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$.

In addition to the aforementioned studies, we introduce the framework of the Left-Hand Side Restricted PCSP (a generalization of the Left-Hand Side Restricted CSP) and study its complexity.

Keywords: Constraint satisfaction problem, promise constraint satisfaction problem, finite tractability, model checking problem

Contents

Introduction

In this chapter we introduce the *constraint satisfaction problem* (CSP) and its *promise* variants, which will be the focus of this thesis. At the end of this chapter we also describe the organization of the thesis.

Many computational problems, including various versions of logical satisfiability, graph coloring, and systems of equations can be phrased as Constraint Satisfaction Problems (CSPs) over fixed templates (see [BKW17a]). There are several (equivalent) formulations of the notion of the CSP. One of them is via homomorphisms of relational structures: a *template* $\mathbb{A}$ is a relational structure with finitely many relations and the CSP over $\mathbb{A}$, written CSP($\mathbb{A}$), is the problem to decide whether a given finite relational structure $\mathbb{X}$ admits a homomorphism to $\mathbb{A}$. Another formulation 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 existential quantifier $\exists$.

The complexity of CSPs over finite templates (i.e., templates whose domain is a finite set) is now completely classified by a celebrated dichotomy theorem independently obtained by Bulatov [Bul17] and Zhuk [Zhu17, Zhu20]: every CSP($\mathbb{A}$) is either tractable (that is, solvable in polynomial-time) or NP-complete. The landmark results leading to the complete classification include Schaefer's dichotomy theorem [Sch78] for CSPs over Boolean structures (i.e., structures with a two-element domain), Hell and Nešetřil's dichotomy theorem [HN90] for CSPs over graphs, and Feder and Vardi's thorough study [FV98] through Datalog and group theory. The latter paper also inspired the development of a mathematical theory of finite-template CSPs [Jea98, BJK05, BOP18], the so called *algebraic approach*, that provided guidance and tools for the general dichotomy theorem by Bulatov and Zhuk.

The algebraic approach has been successfully applied in many variants and generalizations of the CSP such as the infinite-template CSP [Bod08] or valued CSP [KKR17]. The object of the thesis is the study of the computational complexity of promise variants of the CSP.

Constraint Satisfaction Problem (CSP)

A common formal definition of an instance of the CSP over a finite domain is as follows.

Definition 1. *An* instance *of the CSP is a triple* $(V, D, \mathcal{C})$ *where*

- V *is a finite set, called the set of* variables*,*
- D *is a finite set, called the* domain*,*
- $\mathcal{C}$ *is a finite list of constraints, where each constraint is a pair* $C = (x, R)$ *with*
  - x *a tuple of variables of length* n*, called the* scope *of* C*, and*

– *R an n-ary relation on D, called the* constraint relation *of C.*

An assignment*, that is, a mapping $f : V \to D$,* satisfies *a constraint $C = (x, R)$ if $f(x) \in R$, where f is applied component-wise. An assignment f is a* solution *if it satisfies all constraints. The problem is to decide whether the given instance have a solution.*

There are various computational problems that arise from the CSP framework. The problem as defined above is the *decision problem*. (A related problem, the search problem, is to find a solution if at least one solution exists.)

One can define the same problem in different ways. In Chapters 1 and 3 of the thesis we will regard it as a homomorphism problem, while in Chapter 2 as a model checking problem.

A *homomorphism* is a relation-preserving map between two relational structures and it only makes sense if the structures have the same number of relations and the corresponding relations are of the same arity, in which case we say that the structures are *similar*. We will write $\mathbb{A} \to \mathbb{B}$ to denote that there is a homomorphism from $\mathbb{A}$ to $\mathbb{B}$.

Definition 2. *The* CSP *over a relational structure $\mathbb{A}$, written* CSP($\mathbb{A}$)*, is the following problem. Given a finite relational structure $\mathbb{X}$ similar to $\mathbb{A}$, decide whether there is a homomorphism from $\mathbb{X}$ to $\mathbb{A}$.*

The search version of the problem is to find such a homomorphism if you know that at least one exists.

Finally, we give the logical formulation of CSP.

Definition 3. *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 $\exists$.

It is not difficult to see that all these definitions of the CSP are equivalent. Not only that the three definitions of decision versions of CSP are equivalent, but also the decision and the search version of CSP are equivalent ([BJK05]).

An example of a CSP is *k-coloring*. It is a problem of deciding whether the vertices of a given graph can be colored by k different colors so that no adjacent vertices are assigned the same color. The search version of this problem is to find such a coloring given that it is possible. Formulated as a homomorphism problem, k-coloring is the problem of deciding whether a given graph admits a homomorphism to the k-clique (complete graph on k vertices). The complexity of this problem depends on k. More specifically, it is solvable in polynomial time for $k = 2$ and it is NP-complete for $k > 2$.

Promise Constraint Satisfaction Problem (PCSP)

In the previous subsection we saw that finding a 100-coloring of a graph is, in general, hard. A natural question is whether we can do something to get a possibly simpler problem. For example, if we knew that a 3-coloring exists, would this make finding a 100-coloring easier? This gives rise to the Promise Constraint Satisfaction Problem (PCSP).

A *template* for the PCSP is a pair $(\mathbb{A}, \mathbb{B})$ of similar structures such that $\mathbb{A}$ has a homomorphism to $\mathbb{B}$, and the PCSP over $(\mathbb{A}, \mathbb{B})$ is the following problem: given a finite relational structure $\mathbb{X}$ such that $\mathbb{X} \to \mathbb{A}$, find a homomorphism $\mathbb{X} \to \mathbb{B}$. (Notice here that we are not given a homomorphism $\mathbb{X} \to \mathbb{A}$, we just know that it exists.) As you may notice, this is the search version of the problem. As for the CSP, here we also have a decision version, which is the problem to distinguish between the case that a given finite structure $\mathbb{X}$ admits a homomorphism to $\mathbb{A}$ and the case that $\mathbb{X}$ does not have a homomorphism to $\mathbb{B}$ (the promise is that one of the cases takes place). Unlike in the non-promise setting, here we do not know whether the decision and the search versions are equivalent. There is always a reduction from decision to search, but the other direction is open. Throughout the thesis we will consider the decision version of PCSP and we will write $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ for the PCSP over $(\mathbb{A}, \mathbb{B})$. The PCSP framework generalizes that of CSP (take $\mathbb{A} = \mathbb{B}$) and includes important approximation problems. For example, if $\mathbb{A} = \mathbb{K}_k$ (the clique on k vertices) and $\mathbb{B} = \mathbb{K}_l$, $k \leq l$, then $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ is a version of the *approximate graph coloring problem*, namely, the problem to distinguish graphs that are k-colorable from those that are not even l-colorable. The classification of the complexity of this problem is an open problem after more than 40 years of research. On the other hand, the basics of the algebraic approach to CSPs can be generalized to PCSPs [AGH17, BG18, BKO19, BBKO18].

The previous example shows that a full classification of the complexity of PCSPs over graph templates is still open and so is the analogue of Schaefer's Boolean CSP, PCSPs over pairs of Boolean structures. However, strong partial results have already been obtained. Brakensiek and Guruswami [BG18] proved a dichotomy theorem for all *symmetric Boolean templates allowing negations*, i.e., templates $(\mathbb{A}, \mathbb{B})$ such that $\mathbb{A} = (\{0,1\}; R_0, R_1, \dots)$, $\mathbb{B} = (\{0,1\}; S_0, S_1, \dots)$, each relation R_i, S_i is invariant under permutations of coordinates, and $R_0 = S_0$ is the binary disequality relation $\neq$. Ficak, Kozik, Olšák, and Stankiewicz [FKOS19] later generalized this result to all symmetric Boolean templates.

To prove tractability or hardness results for PCSPs, a very simple but useful reduction is often applied: If $(\mathbb{A}, \mathbb{B})$ and $(\mathbb{A}', \mathbb{B}')$ are similar PCSP templates and there exist homomorphisms $\mathbb{A}' \to \mathbb{A}$ and $\mathbb{B} \to \mathbb{B}'$, then the trivial reduction (which does not change the instance) reduces $\mathrm{PCSP}(\mathbb{A}', \mathbb{B}')$ to $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$; we say that $(\mathbb{A}', \mathbb{B}')$ is a *homomorphic relaxation* of $(\mathbb{A}, \mathbb{B})$. In fact, all the tractable symmetric Boolean PCSPs can be reduced in this way to a tractable CSP over a structure with a *possibly infinite domain*.

An interesting example of a PCSP that can be naturally reduced to a tractable CSP over an infinite domain is the following problem: an instance is a list of triples of variables and the problem is to distinguish instances that are satisfiable as positive 1-in-3-SAT instances from those that are not even satisfiable as Not-All-Equal-3-SAT instances. This computational problem can be formulated as $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ where $\mathbb{A}$ consists of the ternary 1-in-3 relation over $\{0,1\}$ and $\mathbb{B}$ consists of the ternary not-all-equal relation over $\{0,1\}$. It is easy to see that $\mathbb{A} \to \mathbb{C} \to \mathbb{B}$ where $\mathbb{C}$ is the relation "$x + y + z = 1$" over the set of all integers. Therefore, $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ is reducible (by means of the trivial reduction) to $\mathrm{PCSP}(\mathbb{C}, \mathbb{C}) = \mathrm{CSP}(\mathbb{C})$ which is a tractable problem. The main result of [Bar19] is that no finite structure can be used in place of $\mathbb{C}$ for this particular template – we call such a PCSP *not finitely tractable*.

In Chapter 1, we initiate a systematic study of this phenomenon. As the main technical contribution, we determine which of the "basic tractable cases" in Brakensiek and Guruswami's classification [BG18] are finitely tractable. It turns out that finite tractability is quite rare, so the infinite nature of the 1-in-3 versus Not-All-Equal problem is not exceptional at all.

Model checking problem

Another way to generalize CSP is to allow different sets of connectives and quantifiers.

The model checking problem [MM18] takes as input a structure $\mathbb{A}$ (often called a model) and a sentence ϕ in a specified logic and asks whether $\mathbb{A} \vDash \phi$, 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 $\mathcal{L}$-MC($\mathbb{A}$).

For the case $\mathcal{L} = \{\exists, \wedge\}$, the problem $\mathcal{L}$-MC($\mathbb{A}$) is exactly CSP($\mathbb{A}$). For the case $\mathcal{L} = \{\exists, \forall, \wedge\}$, the problem $\mathcal{L}$-MC($\mathbb{A}$) is the so called *quantified CSP*, another well-studied class of problems, see the survey [Mar17]. It was widely believed that this class exhibits a P/NP-complete/PSPACE-complete trichotomy [Che12]. A recent breakthrough [ZM20] 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 $\{\exists, \forall, \wedge\}$-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, $\{\exists, \wedge, =\}$-MC($\mathbb{A}$) is no harder than $\{\exists, \wedge\}$-MC($\mathbb{A}$) because equalities can be propagated out in this case, and $\{\forall, \vee\}$-MC($\mathbb{A}$) is dual to $\{\exists, \wedge\}$-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 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 $\mathcal{L}$-MC($\mathbb{A}$), namely $\mathcal{L} = \{\exists, \wedge, \vee\}$ and $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ [Mar08].

The former fragment was addressed in [Mar08]: except for a simple case solvable in polynomial time (in fact, in 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 [Mar08, MM12, MM10] (see also [Mar10, CM21]), the full complexity classification was given in [MM11, MM18]: each problem in this class is in P (even L), or is NP-complete, coNP-complete, or PSPACE-complete.

Promise model checking problem

As one may assume from the name, the promise model checking problem is a generalization of the CSP in the two directions considered above. 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. Chapter 2 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. 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 [Mar08].

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 53.

Other side

So far we have considered CSPs over fixed templates, namely the right-hand side structure is fixed, the input is on the left-hand side, and the question is whether there is a homomorphism from the left-hand side structure to the right-hand side one, or, in the search version, to find such a homomorphism. Actually, CSP in its full generality has neither of the two structures fixed, i.e., the input is a pair of structures and the question is whether there is a homomorphism from the left-hand one to the right-hand one. This problem is also known as the *homomorphism problem*.

Several well-known NP-complete problems can be viewed as restrictions of the homomorphism problem.

For example, the 3-coloring problem is equivalent to the homomorphism problem where the right-hand side input structure is a 3-clique (triangle), and it is NP-complete. On the other hand, the homomorphism problem where the left-hand side input structure is a k-clique is equivalent to the so-called k-`Clique` *problem*, the problem of deciding whether a given graph contains a k-clique. For every fixed k the k-`Clique` problem is solvable in polynomial time, but if we restrict the left-hand side to the class of all cliques, we get the so-called `Clique` problem, which is NP-complete.

For classes $\mathcal{C}$ and $\mathcal{D}$ of structures, let $\mathrm{Hom}(\mathcal{C}, \mathcal{D})$ denote the restriction of the homomorphism problem to input structures $\mathbb{A} \in \mathcal{C}$ and $\mathbb{B} \in \mathcal{D}$. If either $\mathcal{C}$ or $\mathcal{D}$ is the class of all structures, we will use $'-'$.

Left-hand side restricted CSP

If we restrict the left-hand side structure, we get the so-called *left-hand side restricted CSP*. It is a problem of the form $\mathrm{Hom}(\mathcal{C}, -)$. It is proved in [Gro07] that under some complexity theoretic assumption from *parameterized complexity theory*, $\mathrm{Hom}(\mathcal{C}, -)$ is solvable in polynomial time if and only if $\mathcal{C}$ has *bounded tree width modulo homomorphic equivalence*.

Left-hand side restricted PCSP

As mentioned before, an example of a CSP seen from the other side is `Clique`. We can approximate this problem by asking the following question: for an input graph $\mathbb{G}$, decide whether it has a k-clique or not even an l-clique, where $l \leq k$. Furthermore, we generalize this problem to general relational structures in the following way. Let $\mathcal{C}$ be a class of pairs of structures $(\mathbb{A}, \mathbb{B})$ such that there is a homomorphism from $\mathbb{A}$ to $\mathbb{B}$. Then $\mathrm{PHom}(\mathcal{C}, -)$ is the following problem: for $(\mathbb{A}, \mathbb{B}) \in \mathcal{C}$ and a relational structure $\mathbb{D}$ similar to $\mathbb{A}$ (and $\mathbb{B}$), decide whether there is a homomorhpism from $\mathbb{B}$ to $\mathbb{D}$ or not even from $\mathbb{A}$ to $\mathbb{D}$. Chapter 3 deals with complexity of this newly defined problem.

Thesis outline

The main part of the thesis is divided into three independent chapters.

In Chapter 1, we define *finitely tractable* PCSPs and *not finitely tractable* PCSPs, we initiate a systematic study of this phenomenon by giving a general necessary condition for finite tractability and characterizing finite tractability within a class of templates - the "basic" tractable cases in the dichotomy theorem for symmetric Boolean PCSPs allowing negations by Brakensiek and Guruswami [BG18]. These results have been published in [AB21].

In Chapter 2, 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 call these problems *promise model checking problems*.

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 results in this chapter have been published in [ABB22].

In Chapter 3, we introduce the framework of left-hand side restricted PCSPs and we provide some initial results. The main technical contribution is a sufficient condition for W[1]-hardness. This is an unpublished joint work with Libor Barto and Victor Dalmau.

1. Finitely tractable PCSPs

1.1 Introduction

Throughout this chapter we will regard the constraint satisfaction problem as a homomorphism problem: a *template* $\mathbb{A}$ is a relational structure with finitely many relations and the CSP over $\mathbb{A}$, written CSP($\mathbb{A}$), is the problem to decide whether a given finite relational structure $\mathbb{X}$ (similar to $\mathbb{A}$) admits a homomorphism to $\mathbb{A}$.

This chapter concerns one generalization of the basic CSP framework, the Promise CSP (PCSP).

A *template* for the PCSP is a pair $(\mathbb{A}, \mathbb{B})$ of similar structures such that $\mathbb{A}$ has a homomorphism to $\mathbb{B}$, and the PCSP over $(\mathbb{A}, \mathbb{B})$, written PCSP($\mathbb{A}, \mathbb{B}$), is the problem to distinguish between the case that a given finite structure $\mathbb{X}$ admits a homomorphism to $\mathbb{A}$, written $\mathbb{X} \to \mathbb{A}$, and the case that $\mathbb{X}$ does not have a homomorphism to $\mathbb{B}$, written $\mathbb{X} \not\to \mathbb{B}$, (the promise is that one of the cases takes place). Obviously, PCSP($\mathbb{A}, \mathbb{A}$) is CSP($\mathbb{A}$).

A powerful tool used in determining the computational complexity of a PCSP is *homomorphic relaxation*: $(\mathbb{A}', \mathbb{B}')$ is a homomorphic relaxation of $(\mathbb{A}, \mathbb{B})$ if $\mathbb{A}' \to \mathbb{A} \to \mathbb{B} \to \mathbb{B}'$, and in that case PCSP($\mathbb{A}', \mathbb{B}'$) is not harder than PCSP($\mathbb{A}, \mathbb{B}$).

Since we already know "a lot" about CSP, a common way to prove that some PCSP($\mathbb{A}, \mathbb{B}$) is tractable is to find a structure $\mathbb{C}$ such that $\mathbb{A} \to \mathbb{C} \to \mathbb{B}$ and such that CSP($\mathbb{C}$) is tractable. In [Bar19] Barto found an example of a PCSP template such that such a "sandwitched" structure has to be infinite – this PCSP is not finitely tractable in the sense of the following definition.

Definition 4. *We say that* PCSP($\mathbb{A}, \mathbb{B}$) *is* finitely tractable *if there exists a finite relational structure* $\mathbb{C}$ *such that* $\mathbb{A} \to \mathbb{C} \to \mathbb{B}$ *and* CSP($\mathbb{C}$) *is tractable. Otherwise we call* PCSP($\mathbb{A}, \mathbb{B}$) not finitely tractable*. (We assume P $\neq$ NP throughout the thesis.)*

In this chapter, we initiate a systematic study of this phenomenon. As the main technical contribution, we determine which of the "basic tractable cases" in Brakensiek and Guruswami's classification [BG18] are finitely tractable.

1.1.1 Symmetric Boolean PCSPs allowing negations

We now discuss the classification of symmetric Boolean templates allowing negations from [BG18]. It will be convenient to describe these templates by listing the corresponding relation pairs, that is, instead of $(\mathbb{A} = (\{0,1\}; R_1, R_2, \dots, R_n), \mathbb{B} = (\{0,1\}; S_1, S_2, \dots, S_n))$ we describe this template by the list (R_1, S_1), (R_2, S_2), $\dots$, (R_n, S_n). Recall that the template is *symmetric* if all the involved relations are symmetric, i.e., invariant under any permutation of coordinates, and the template *allows negations* if $(\neq, \neq)$ is among the relation pairs, where $\neq = \{(0,1),(1,0)\}$ is the disequality relation.

It may be also helpful to think of an instance of PCSP($\mathbb{A}, \mathbb{B}$) as a list of constraints of the form R_i(variables) and the problem is to distinguish between instances where each constraint is satisfiable and those which are not satisfiable even when we replace each R_i by the corresponding "relaxed version" S_i. Allowing negations then means that we can use constraints $x \neq y$ – we can effectively negate variables.

The following relations are important for the classification.

- odd-in-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{i=1}^{s} x_i \text{ is odd}\}$,
  even-in-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{n=1}^{s} x_i \text{ is even}\}$
- r-in-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{n=1}^{s} x_i = r\}$
- $\leq r$-in-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{i=1}^{s} x_i \leq r\}$,
  $\geq r$-in-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{i=1}^{s} x_i \geq r\}$
- not-all-equal-$s = \{\mathbf{x} \in \{0,1\}^s : \sum_{i=1}^{s} x_i \notin \{0, s\}\}$

The next theorem lists some of the tractable cases of the classification, which are "basic" in the sense explained below.

Theorem 1. PCSP$((P,Q),(\neq,\neq))$ *is tractable if* (P,Q) *is equal to*

(a) $($*odd-in-*s, *odd-in-*$s)$, *or* $($*even-in-*s, *even-in-*$s)$, *or*

(b) $(\leq r$*-in-*$s, \leq(2r-1)$*-in-*$s)$ *and* $r \leq s/2$, *or*
$(\geq r$*-in-*$s, \geq(2r-s+1)$*-in-*$s)$ *and* $r \geq s/2$, *or*

(c) $(r$*-in-*s, *not-all-equal-*$s)$

for some positive integers r, s.

It is proved in Appendix 1.A that every tractable symmetric Boolean PCSP allowing negations can be obtained by

- taking any number of relation pairs from one of the following three items:
  (a) (odd-in-s, odd-in-s), or (even-in-s, even-in-s)
  (b) ($\leq r$-in-s, $\leq(2r-1)$-in-s) and $r \leq s/2$, or
  ($\geq r$-in-s, $\geq(2r-s+1)$-in-s) and $r \geq s/2$, or
  ($\frac{s}{2}$-in-s, not-all-equal-s) and s is even
  (c) (r-in-s, not-all-equal-s)

  where r and s are positive integers,
- adding any number of "trivial" relation pairs (P,Q) such that $P \subseteq Q$, and Q is the full relation or P contains only constant tuples, and
- taking a homomorphic relaxation of the obtained template.

In this sense, Theorem 1 provides building blocks for all tractable templates.

1.1.2 Contributions

Some of the cases in Theorem 1 are finitely tractable: templates in item (a) are tractable CSPs (they can be decided by solving systems of linear equations of the two-element field), templates in item (c) for r odd and s even are homomorphic relaxations of (odd-in-s, odd-in-s), and templates in item (b) for $r = 1$ or $r = s - 1$ as well as all templates with $s \leq 2$ are tractable CSPs (reducible to 2-SAT) [Sch78, BKW17a]. Our main theorem proves that all the remaining cases are not finitely tractable. In fact, we prove this property even for some relaxations of these templates:

Theorem 2. *The PCSP over any of the following templates is not finitely tractable.*

(1) $(r\text{-}in\text{-}s, \leq(2r-1)\text{-}in\text{-}s), (\neq, \neq)$ *where* $1 < r < s/2$,
$(r\text{-}in\text{-}s, \geq(2r-s+1)\text{-}in\text{-}s), (\neq, \neq)$ *where* $s/2 < r < s-1$

(2) $(\leq r\text{-}in\text{-}s, \leq(2r-1)\text{-}in\text{-}s), (\neq, \neq)$ *where* s *is even,* $1 < r = s/2$
$(\geq r\text{-}in\text{-}s, \geq(2r-s+1)\text{-}in\text{-}s), (\neq, \neq)$ *where* s *is even,* $1 < r = s/2$

(3) $(r\text{-}in\text{-}s, \leq(2r-1)\text{-}in\text{-}s), (\neq, \neq)$ *where* s *is even,* $1 < r = s/2$*, and* r *is even*
$(r\text{-}in\text{-}s, \geq(2r-s+1)\text{-}in\text{-}s), (\neq, \neq)$ *where* s *is even,* $1 < r = s/2$*, and* r *is even*

(4) $(r\text{-}in\text{-}s, not\text{-}all\text{-}equal\text{-}s)$ *where* $s > r$*,* $s > 2$*, and* r *is even or* s *is odd*

Note that the templates in the last item do not contain the disequality pair; the special case with $r = 1$ and $s = 3$ is the main result of [Bar19]. Disequalities in the other items are necessary, since otherwise the templates are homomorphic relaxations of CSPs over one-element structures.

In Theorem 8 we provide a general necessary condition for finite tractability of an arbitrary finite-template PCSP in terms of so called h1 identities. Showing that templates in Theorem 2 do not satisfy this necessary condition forms the bulk of the chapter.

The necessary condition in Theorem 8 seems very unlikely to be sufficient for finite tractability. Nevertheless, we observe in Theorem 5 that finite tractability *does* depend only on h1 identities, just like standard tractability [BKO19], see Theorem 3 and the discussion following the theorem.

1.2 Preliminaries

1.2.1 PCSP

For every positive integer n we let $[n] = \{1, 2, \ldots, n\}$.

A *relational structure* (of finite signature) is a tuple $\mathbb{A} = (A; R_1, R_2, \ldots, R_n)$ where A is a set, called the *domain*, and each R_i is a relation on A of arity $\mathrm{ar}(R_i) \geq 1$, that is, $R_i \subseteq A^{\mathrm{ar}(R_i)}$. The structure $\mathbb{A}$ is finite if A is finite. Two relational structures $\mathbb{A} = (A; R_1, R_2, \ldots, R_n)$ and $\mathbb{B} = (B; S_1, S_2, \ldots, S_n)$ are *similar* if they have the same number of relations and $\mathrm{ar}(R_i) = \mathrm{ar}(S_i)$ for each $i \in [n]$. In this case, a *homomorphism* from $\mathbb{A}$ to $\mathbb{B}$ is a mapping $f : A \to B$ such that $(f(a_1), f(a_2), \ldots, f(a_k)) \in S_i$ whenever $i \in [n]$ and $(a_1, a_2, \ldots, a_k) \in R_i$

where $k = \text{ar}(R_i)$. If there exists a homomorphism from $\mathbb{A}$ to $\mathbb{B}$, we write $\mathbb{A} \to \mathbb{B}$, and if there is none, we write $\mathbb{A} \not\to \mathbb{B}$.

Definition 5. *A* PCSP template *is a pair* $(\mathbb{A}, \mathbb{B})$ *of similar relational structures such that* $\mathbb{A} \to \mathbb{B}$.

The PCSP *over* $(\mathbb{A}, \mathbb{B})$*, written* $\text{PCSP}(\mathbb{A}, \mathbb{B})$*, is the following problem. Given a finite relational structure* $\mathbb{X}$ *similar to* $\mathbb{A}$ *(and* $\mathbb{B}$*), output "Yes." if* $\mathbb{X} \to \mathbb{A}$ *and output "No." if* $\mathbb{X} \not\to \mathbb{B}$.

We define $\text{CSP}(\mathbb{A}) = \text{PCSP}(\mathbb{A}, \mathbb{A})$.

Definition 6. *Let* $(\mathbb{A}, \mathbb{B})$ *and* $(\mathbb{A}', \mathbb{B}')$ *be similar* PCSP *templates. We say that* $(\mathbb{A}', \mathbb{B}')$ *is a* homomorphic relaxation *of* $(\mathbb{A}, \mathbb{B})$ *if* $\mathbb{A}' \to \mathbb{A}$ *and* $\mathbb{B} \to \mathbb{B}'$.

Recall that if $(\mathbb{A}', \mathbb{B}')$ is a homomorphic relaxation of $(\mathbb{A}, \mathbb{B})$, then the trivial reduction, which does not change the input structure $\mathbb{X}$, reduces $\text{PCSP}(\mathbb{A}', \mathbb{B}')$ to $\text{PCSP}(\mathbb{A}, \mathbb{B})$.

1.2.2 Polymorphisms

A crucial concept for the algebraic approach to (P)CSP is a polymorphism.

Definition 7. *Let* $R \subseteq A^k$ *and* $S \subseteq B^k$ *be relations. A function* $c : A^n \to B$ *is a* polymorphism *of* (R, S) *if*

$$\begin{pmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{k1} \end{pmatrix} \in R, \begin{pmatrix} a_{12} \\ a_{22} \\ \vdots \\ a_{k2} \end{pmatrix} \in R, \dots, \begin{pmatrix} a_{1n} \\ a_{2n} \\ \vdots \\ a_{kn} \end{pmatrix} \in R \Rightarrow \begin{pmatrix} c(a_{11}, a_{12}, \dots, a_{1n}) \\ c(a_{21}, a_{22}, \dots, a_{23}) \\ \vdots \\ c(a_{k1}, a_{k2}, \dots, a_{kn}) \end{pmatrix} \in S.$$

Definition 8. *Let* $\mathbb{A} = (A; R_1, R_2, \dots, R_m)$ *and* $\mathbb{B} = (B; S_1, S_2, \dots, S_m)$ *be two similar relational structures. A function* $c : A^n \to B$ *is a* polymorphism *from* $\mathbb{A}$ *to* $\mathbb{B}$ *if it is a polymorphism of* (R_i, S_i) *for every* $i \in \{1, 2, \dots, m\}$.

We denote the set of all polymorphisms from $\mathbb{A}$ *to* $\mathbb{B}$ *by* $\text{Pol}(\mathbb{A}, \mathbb{B})$ *and define* $\text{Pol}(\mathbb{C}) = \text{Pol}(\mathbb{C}, \mathbb{C})$.

The computational complexity of a PCSP depends only on the set of polymorphisms of its template [BG18]. We note that tractability of the PCSPs in Theorem 1 stems from nice polymorphisms: parities (item (a)), majorities (item (b)), and alternating thresholds (item (c)).

The set of polymorphisms is an algebraic object named minion in [BKO19], which we define in Definition 10 below.

Definition 9. *An n-ary function* $f^\pi : A^n \to B$ *is called a* minor *of an m-ary function* $f : A^m \to B$ *given by a map* $\pi : [m] \to [n]$ *if*

$$f^\pi(x_1, x_2, \dots, x_n) = f(x_{\pi(1)}, x_{\pi(2)} \dots, x_{\pi(m)})$$

for all $x_1, x_2, \dots, x_n \in A$.

Definition 10. *Let* $\mathcal{O}(A, B) = \{f : A^n \to B : n \geq 1\}$*. A* minion *on* (A, B) *is a non-empty subset* $\mathcal{M}$ *of* $\mathcal{O}(A, B)$ *that is closed under taking minors. For fixed* $n \geq 1$*, let* $\mathcal{M}^{(n)}$ *denote the set of n-ary functions from* $\mathcal{M}$.

As mentioned, $\mathcal{M} = \mathrm{Pol}(\mathbb{A}, \mathbb{B})$ is always a minion and the complexity of $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ depends only on $\mathcal{M}$. This result was strengthened in [BKO19, BBKO18] (generalizing the same result for CSPs [BOP18]) as follows.

Definition 11. *Let $\mathcal{M}$ and $\mathcal{N}$ be two minions. A mapping $\xi : \mathcal{M} \to \mathcal{N}$ is called a* minion homomorphism *if it preserves arities and preserves taking minors, i.e., $\xi(f^\pi) = (\xi(f))^\pi$ for every $f \in \mathcal{M}^{(m)}$ and every $\pi : [m] \to [n]$.*

Theorem 3. *Let $(\mathbb{A}, \mathbb{B})$ and $(\mathbb{A}', \mathbb{B}')$ be PCSP templates. If there exists a minion homomorphism $\mathrm{Pol}(\mathbb{A}', \mathbb{B}') \to \mathrm{Pol}(\mathbb{A}, \mathbb{B})$, then $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ is log-space reducible to $\mathrm{PCSP}(\mathbb{A}', \mathbb{B}')$.*

An *h1 identity* (h1 stands for height one) is a meaningful expression of the form function(variables) $\approx$ function(variables), e.g., if $f : A^3 \to B$ and $g : A^4 \to B$, then $f(x, y, x) \approx g(y, x, x, z)$ is an h1 identity. Such an h1 identity is *satisfied* if the corresponding equation holds universally, e.g., $f(x, y, x) \approx g(y, x, x, z)$ is satisfied if and only if $f(x, y, x) = g(y, x, x, z)$ for every $x, y, z \in A$.

Every minion homomorphism $\xi : \mathcal{M} \to \mathcal{N}$ preserves h1 identities in the sense that if functions $f, g \in \mathcal{M}$ satisfy an h1 identity, then so do their ξ-images $\xi(f), \xi(g) \in \mathcal{N}$. In fact, an arity-preserving ξ between minions is a minion homomorphism if and only if it preserves h1 identities (see [BOP18] for details). In this sense, Theorem 3 shows that the complexity of a PCSP depends only on h1 identities satisfied by polymorphisms.

1.2.3 Notation for tuples

Repeated entries in tuples will be indicated by $\times$, e.g. $(2 \times a, 3 \times b)$ stands for the tuple (a, a, b, b, b).

The i-th *cyclic shift* of a tuple $(x_1, \dots, x_m)$ is the tuple $(x_{(m-i \mod m)+1}, \dots, x_m, x_1, \dots, x_{(m-i-1 \mod m)+1})$. A *cyclic shift* is the i-th cyclic shift for some i. We will use cyclic shifts both for tuples of zeros and ones and tuples of variables.

We will often use special p-tuples and $n = p^2$-tuples of zeros and ones as arguments for Boolean functions, where p will be a fixed prime number. For $0 \le k \le p$, $0 \le l \le p^2$, and $0 \le k^1, \dots, k^p \le p$ we write

$$\langle k \rangle_p = (k \times 1, (p-k) \times 0) = (\underbrace{1, 1, \dots, 1}_{k}, \underbrace{0, 0, \dots, 0}_{p-k}), \quad \langle l \rangle_n = (\underbrace{1, 1, \dots, 1}_{l}, \underbrace{0, 0, \dots, 0}_{n-l})$$

and

$$\langle k^1, k^2, \dots, k^p \rangle_p = \langle k^1 \rangle_p \langle k^2 \rangle_p \dots \langle k^p \rangle_p$$

for the concatenation of $\langle k^1 \rangle_p$, $\langle k^2 \rangle_p \dots$, $\langle k^p \rangle_p$. (Note here that the "i" in k^i is an index, not an exponent.) The subscripts p and n in $\langle\rangle_p$ and $\langle\rangle_n$ will be usually clear from the context and we omit them. We will sometimes need to shift n-ary tuples $\langle k^1, k^2, \dots, k^p \rangle$ blockwise, e.g., to $\langle k^2, \dots, k^p, k^1 \rangle$. In such a situation we talk about a *p-ary cyclic shift* to avoid confusion.

It will be often convenient to think of an n-tuple $\mathbf{k} = \langle k^1, k^2, \dots, k^p \rangle$ as a $p \times p$ zero-one matrix with columns $\langle k^1 \rangle$, $\langle k^2 \rangle$,..., $\langle k^p \rangle$. For example, the ones in $\langle p \times 5 \rangle$ form a $5 \times p$ "rectangle" and $\langle (p-2) \times 5, 2 \times 4 \rangle$ is "almost" a $5 \times p$ rectangle – the bottom right 1×2 corner is removed. A p-ary cyclic shift of $\mathbf{k}$ corresponds to cyclic permutation of columns.

The area of a zero-one n-tuple $\mathbf{k}$ is defined as the fraction of ones and is denoted $\lambda(\mathbf{k})$.

$$\lambda(\mathbf{k}) = \left(\sum_{i=1}^{n} k_i\right) / p^2$$

The area of $\langle k^1, k^2, \ldots, k^p \rangle$ is thus $(k^1 + k^2 + \cdots + k^p)/p^2$.

If t is a p-ary function, we simply write $t\langle k \rangle$ instead of $t(\langle k \rangle)$. Similar shorthand is used for n-ary functions and tuples $\langle k^1, k^2, \ldots, k^p \rangle_p$.

1.3 Finitely tractable PCSPs

1.3.1 Finite tractability depends only on h1 identities

We start by observing that finite tractability also depends only on h1 identities satisfied by polymorphisms, just like standard tractability (recall the discussion about h1 identities and minion homomorphisms below Theorem 3). This result, Theorem 5, is an immediate consequence of the following lemma and Theorem 3.

Lemma 4. *Let* $(\mathbb{A}, \mathbb{B})$ *be a PCSP template. Then the following are equivalent.*

- $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ *is finitely tractable.*
- *There exists a finite relational structure* $\mathbb{C}$ *such that* $\mathrm{CSP}(\mathbb{C})$ *is solvable in polynomial time and there exists a minion homomorphism* $\mathrm{Pol}(\mathbb{C}) \to \mathrm{Pol}(\mathbb{A}, \mathbb{B})$.

Proof. This lemma is a consequence of known results and we only sketch the argument here. In Section II.B of [Bar19] it is argued that the first item is equivalent to the claim that a finite tractable template $(\mathbb{C}, \mathbb{C})$ pp-constructs $(\mathbb{A}, \mathbb{B})$. The latter claim is equivalent to the second item by Theorem 4.12 in [BBKO18]. □

Theorem 5. *Let* $(\mathbb{A}, \mathbb{B})$ *and* $(\mathbb{A}', \mathbb{B}')$ *be PCSP templates. If there exists a minion homomorphism* $\mathrm{Pol}(\mathbb{A}', \mathbb{B}') \to \mathrm{Pol}(\mathbb{A}, \mathbb{B})$ *and* $\mathrm{PCSP}(\mathbb{A}', \mathbb{B}')$ *is finitely tractable, then so is* $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$.

1.3.2 Necessary condition for finite tractability

In this subsection, we derive the necessary condition for finite tractability that will be used to prove Theorem 2. A cyclic polymorphism is a starting point for the condition.

Definition 12. *A function* $c : A^p \to B$ *is called* cyclic *if it satisfies the h1 identity*

$$c(x_1, x_2, \ldots, x_p) \approx c(x_2, \ldots, x_p, x_1).$$

Cyclic polymorphisms can be used [BK12] to characterize the borderline between tractable and NP-complete CSPs proposed in [BJK05] and confirmed in [Bul17, Zhu17, Zhu20]. We only state the direction needed in this paper.

Theorem 6 ([BK12])**.** *Let* $\mathbb{C}$ *be a CSP template over a finite domain* C. *If* $\mathrm{CSP}(\mathbb{C})$ *is not NP-complete, then* $\mathbb{C}$ *has a cyclic polymorphism of arity* p *for every prime number* $p > |C|$.

Polymorphism minions of CSP templates are closed under arbitrary composition (cf. [BKW17a]). In particular, if $\mathrm{CSP}(\mathbb{C})$ is not NP-complete, then $\mathrm{Pol}(\mathbb{C})$ contains the function

$$\begin{aligned} &t(x_{11}, x_{21}, \dots, x_{p1},\ x_{12}, x_{22}, \dots, x_{p2},\ \dots,\ x_{1p}, x_{2p}, \dots, x_{pp}) \\ &\quad = c(c(x_{11}, x_{21}, \dots, x_{p1}), c(x_{12}, x_{22}, \dots, x_{p2}), \dots, c(x_{1p}, x_{2p}, \dots, x_{pp})), \end{aligned} \tag{1.1}$$

where c is a p-ary cyclic function and $p > |C|$. Such a function satisfies strong h1 identities which are not satisfied by the templates in Theorem 2. We now (in two steps) describe one such collection of strong enough identities.

Definition 13. *A function $t : A^{p^2} \to B$ is* doubly cyclic *if it satisfies every identity of the form $t(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_p) \approx t(\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_p)$, where $\mathbf{x}_i$ is a p-tuple of variables and $\mathbf{y}_i$ is a cyclic shift of $\mathbf{x}_i$ for every $i \in [p]$, and every identity of the form $t(\mathbf{x}_1, \mathbf{x}_2 \dots, \mathbf{x}_p) \approx t(\mathbf{x}_2, \dots, \mathbf{x}_p, \mathbf{x}_1)$, where each $\mathbf{x}_i$ is a p-tuple of variables.*

Observe that t from 1.1 is doubly cyclic – the first type of identities come from the cyclicity of the inner c while the second type from the outer c. It will be also useful for us to observe in Lemma 9 that, after rearranging the arguments (we read them row-wise), t is a cyclic function of arity p^2. From the finiteness of the domain C we get one more property of function t. In the next definition, by an *x/y-tuple* we mean a tuple containing only variables x and y.

Definition 14. *A doubly cyclic function $t : A^{p^2} \to B$ is b-*bounded *if there exists an equivalence relation $\sim$ on the set of all p-ary x/y-tuples with at most b equivalence classes such that t satisfies every identity of the form $t(\mathbf{u}_1, \mathbf{u}_2, \dots \mathbf{u}_p) \approx t(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_p)$ where $\mathbf{u}_i$ and $\mathbf{v}_i$ are x/y-tuples such that $\mathbf{u}_i \sim \mathbf{v}_i$ for every $i \in [p]$.*

Lemma 7. *Let $c : C^p \to C$ be a cyclic function. Then the function t defined by 1.1 is a b-bounded doubly cyclic function for $b = |C|^{|C|^2}$.*

Proof. We define $\sim$ by declaring two p-ary x/y-tuples $\mathbf{u}$ and $\mathbf{v}$ $\sim$-equivalent if $c(\mathbf{u}) \approx c(\mathbf{v})$. As there are $b = |C|^{|C|^2}$ binary functions $C^2 \to C$, this equivalence has at most b equivalence classes. By definitions, t is then b-bounded and doubly cyclic. □

The promised necessary condition for finite tractability is now a simple consequence:

Theorem 8. *Let $(\mathbb{A}, \mathbb{B})$ be a finite PCSP template that is finitely tractable. Then there exists b such that $(\mathbb{A}, \mathbb{B})$ has a p^2-ary b-bounded doubly cyclic polymorphism for every sufficiently large prime p.*

Proof. If $(\mathbb{A}, \mathbb{B})$ is finitely tractable, then, by Lemma 4, there exists a minion homomorphism $\xi : \mathrm{Pol}(\mathbb{C}) \to \mathrm{Pol}(\mathbb{A}, \mathbb{B})$, where $\mathbb{C}$ is finite and $\mathrm{CSP}(\mathbb{C})$ is tractable. By Theorem 6, $\mathbb{C}$ has a p-ary cyclic polymorphism for every sufficiently large prime. Then, by Lemma 7, the polymorphism t of $\mathbb{C}$ defined by 1.1 is a b-bounded and doubly cyclic (with the appropriate b). As ξ preserves h1 identities, $\xi(t)$ is a b-bounded doubly cyclic polymorphism of $(\mathbb{A}, \mathbb{B})$. □

1.3.3 Sketch of the proof of Theorem 2

Finally, we are ready to start proving Theorem 2. Without loss of generality, we consider only templates on the first lines of Cases (1)–(3) of Theorem 2 (in particular, $r \leq s/2$) and assume that $r \leq s/2$ in Case (4) (the remaining templates can be obtained by swapping zero and one in the domains). The general idea for all the cases is the same. In the next section we will give a complete proof for the Case (1), and then the proofs for other cases will follow, but in much less detail, because a big part of the proofs is very similar to the proof for the Case (1), the parts where the proofs significantly differ will be written in detail. But here we sketch the proof for a general fixed template $(\mathbb{A}, \mathbb{B})$ from Cases (1)-(4), where $r \leq s/2$.

Striving for a contradiction, suppose that $\mathrm{PCSP}(\mathbb{A}, \mathbb{B})$ is finitely tractable. By Theorem 8 there exists b such that $(\mathbb{A}, \mathbb{B})$ has a p^2-ary b-bounded doubly cyclic polymorphism t for every sufficiently large arity p^2. We fix such a b and t, where p is fixed to a sufficiently large prime p congruent to 1 modulo s. How large must p be will be seen in due course. We denote $n = p^2$ and observe that $n \equiv 1 \pmod{s}$ as well.

Using the cyclicity and double cyclicity we will show that certain n-tuples $\mathbf{z}$ are tame in that $t(\mathbf{z}) = t\langle 0 \rangle$ (recall here the notation in Subsection 1.2.3) iff the area of $\mathbf{z}$ is below a threshold θ. The *threshold* is defined as $\theta = 1/2$ for all the templates but the (r-in-s, not-all-equal-s) template in Case (4), where we set $\theta = r/s$ (observe that $\theta = r/s$ also in Case (2) and (3)).

The evaluations that we use are called near-threshold almost rectangles defined as follows.

Definition 15. *A tuple $\mathbf{z} \in \{0,1\}^n$ is an* almost rectangle *if it is a p-ary cyclic shift of a tuple of the form $\langle z^1, \dots, z^1, z^2, \dots, z^2 \rangle_p$, where $0 \leq z^1, z^2 \leq p$, the number of z^1's is arbitrary, and $|z^1 - z^2| < 5b$. The quantity $\Delta z = |z^1 - z^2|$ is referred to as the* step size. *We say that $\mathbf{z}$ is* near-threshold *if $|\lambda(\mathbf{z}) - \theta| < 1/s^{\Delta z + 3}$.*

The proof can now be finished by finding two near-threshold almost rectangles $\mathbf{z}_1$ and $\mathbf{z}_2$ such that $\lambda(\mathbf{z}_1) < \theta < \lambda(\mathbf{z}_2)$ and $t(\mathbf{z}_1) = t(\mathbf{z}_2)$; but the tameness of near-threshold almost rectangles gives us $t(\mathbf{z}_1) = t\langle 0 \rangle_n \neq 1 - t\langle 0 \rangle_n = t(\mathbf{z}_2)$, a contradiction.

1.4 Case (1): $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where $1 < r < s/2$

In this section we will prove that $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where $1 < r < s/2$ is not finitely tractable, which is (half of the) Case (1) in Theorem 2. Striving for a contradiction, suppose that it is finitely tractable. By Theorem 8 there exists b such that $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ has a p^2-ary b-bounded doubly cyclic polymorphism t for every sufficiently large arity p^2. We fix such a b and t, where p is fixed to a sufficiently large prime p congruent to 1 modulo s (which is possible by the Dirichlet prime number theorem). How large must p be will be seen in due course. We denote $n = p^2$ and observe that $n \equiv 1 \pmod{s}$ as well.

We will use the cyclicity of an operation obtained from t by an appropriate rearrangnment of its arguments, stated in the following lemma.

Lemma 9. *Let $t : A^{p^2} \to B$ be a doubly cyclic function. Then the function t^σ defined by*

$$t^\sigma \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \cdots & x_{pp} \end{pmatrix} = t \begin{pmatrix} x_{11} & x_{21} & \cdots & x_{p1} \\ x_{12} & x_{22} & \cdots & x_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1p} & x_{2p} & \cdots & x_{pp} \end{pmatrix}$$

is a cyclic function.

Proof. By cyclically shifting the arguments we get the same result:

$$\begin{aligned}
&t^\sigma(x_{21}, x_{31}, \ldots, x_{p1}, x_{12}, x_{22}, x_{32}, \ldots, x_{p2}, x_{13}, \ldots, x_{2p}, x_{3p}, \ldots, x_{pp}, x_{11}) \\
&= t^\sigma \begin{pmatrix} x_{21} & \cdots & x_{2,p-1} & x_{2p} \\ \vdots & \ddots & \vdots & \vdots \\ x_{p1} & \cdots & x_{p,p-1} & x_{pp} \\ x_{12} & \cdots & x_{1p} & x_{11} \end{pmatrix} = t \begin{pmatrix} x_{21} & \cdots & x_{p1} & x_{12} \\ \vdots & \ddots & \vdots & \vdots \\ x_{2,p-1} & \cdots & x_{p,p-1} & x_{1p} \\ x_{2p} & \cdots & x_{pp} & x_{11} \end{pmatrix} \\
&= t \begin{pmatrix} x_{21} & \cdots & x_{p1} & x_{11} \\ \vdots & \ddots & \vdots & \vdots \\ x_{2,p-1} & \cdots & x_{p,p-1} & x_{1,p-1} \\ x_{2p} & \cdots & x_{pp} & x_{1p} \end{pmatrix} = t \begin{pmatrix} x_{11} & x_{21} & \cdots & x_{p1} \\ x_{12} & x_{22} & \cdots & x_{p2} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1p} & x_{2p} & \cdots & x_{pp} \end{pmatrix} \\
&= t^\sigma \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \cdots & x_{pp} \end{pmatrix} \\
&= t^\sigma(x_{11}, x_{21}, \ldots, x_{p1}, x_{12}, x_{22}, \ldots, x_{p2}, \ldots, x_{1p}, x_{2p}, \ldots, x_{pp}).
\end{aligned}$$

□

The following lemma is a consequence of the fact that t^σ is a polymorphism (as t is) which is, additionally, cyclic by Lemma 9.

Lemma 10. *Let $\langle k_1 \rangle, \langle k_2 \rangle, \ldots, \langle k_s \rangle$, where $0 \le k_i \le n$, be an s-tuple of n-tuples such that $\sum_{i=1}^s k_i = rn$. Then $(t^\sigma \langle k_1 \rangle, t^\sigma \langle k_2 \rangle, \ldots, t^\sigma \langle k_s \rangle) \in \le (2r-1)$-in-$s$.*

Moreover, we have $t^\sigma \langle n-k \rangle = 1 - t^\sigma \langle k \rangle$ for every $0 \le k \le n$.

Proof. For the first part, form an $s \times rn$ matrix M whose first row is $\langle k_1 \rangle_{rn}$ and the j-th row is the $(\sum_{l=1}^{j-1} k_l)$-th cyclic shift of $\langle k_j \rangle_{rn}$ for $j \in \{2, \ldots, s\}$. Note that each column of M contains exactly 1 one. Split this matrix into r-many $s \times n$ blocks $M^1, M^2, \ldots, M^r$. Their sum $X = \sum_{j=1}^r M^j$ is an $s \times n$ zero-one matrix whose each column contains exactly r ones. Moreover, for all $j \in [s]$, the j-th row of X is a cyclic shift of $\langle k_j \rangle$, therefore its t^σ-image is $t^\sigma \langle k_j \rangle$ by cyclicity of t^σ. Each column belongs to the relation r-in-s, therefore, as t^σ is a polymorphism, we get that t^σ applied to the rows gives a tuple in $\le (2r-1)$-in-s. This implies the first claim.

For the second part, we take $\langle k \rangle$ together with the k-th cyclic shift of $\langle n-k \rangle$ and use the fact that t^σ preserves the disequality relation pair. □

Lemma 11. *Denote $a = \lfloor n/2 \rfloor$. For every $0 \leq k \leq 2a$, we have*

$$t^\sigma \langle k \rangle_n = \begin{cases} t^\sigma \langle 0 \rangle_n & \text{if } 0 \leq k \leq a \\ 1 - t^\sigma \langle 0 \rangle_n & \text{if } 1 + a \leq k \leq 2a \end{cases}$$

Proof. We prove $t^\sigma \langle k \rangle = 0$ and $t^\sigma \langle n - k \rangle = 1$ for any $0 \leq k \leq a$ by induction on $i = a - k$, $i = 0, 1, \ldots, a$. For the first step, $k = (n-1)/2$, we apply Lemma 10 to the s-tuple $2r \times \langle k \rangle, \langle r \rangle, (s - 2r - 1) \times \langle 0 \rangle$. (We can apply Lemma 10 because $2rk + r = rn$.) Since $\leq (2r-1)$-in-s contains no tuple with more than $(2r-1)$ ones, we get $t^\sigma \langle k \rangle = 0$. Then also $t^\sigma \langle n - k \rangle = 1$ by the second part of the lemma. For the induction step, we use the tuple

$$r \times \langle k \rangle, r \times \langle n - k - 1 \rangle, \langle r \rangle, (s - 2r - 1) \times \langle 0 \rangle$$

in a similar way, additionally using that $t^\sigma \langle n - k - 1 \rangle = 1$ by the induction hypothesis.

□

Definition 16. *A tuple $\mathbf{z} \in \{0,1\}^n$ is* tame *if*

$$t(\mathbf{z}) = \begin{cases} t\langle 0 \rangle_n & \text{if } \lambda(\mathbf{z}) < 1/2 \\ 1 - t\langle 0 \rangle_n & \text{if } \lambda(\mathbf{z}) > 1/2 \end{cases}$$

(Note here that $\lambda(\mathbf{z})$ is never equal to $1/2$ since n is odd and $n \equiv 1 \pmod{s}$.)

The evaluations that we use are called near-threshold almost rectangles defined as follows.

Definition 17. *A tuple $\mathbf{z} \in \{0,1\}^n$ is an* almost rectangle *if it is a p-ary cyclic shift of a tuple of the form $\langle z^1, \ldots, z^1, z^2, \ldots, z^2 \rangle_p$, where $0 \leq z^1, z^2 \leq p$, the number of z^1's is arbitrary, and $|z^1 - z^2| < 5b$. The quantity $\Delta z = |z^1 - z^2|$ is referred to as the* step size. *We say that $\mathbf{z}$ is* near-threshold *if $|\lambda(\mathbf{z}) - 1/2| < 1/s^{\Delta z + 3}$.*

Observe that an almost rectangle $\mathbf{z} = \langle z^2 + 1, \ldots, z^2 + 1, z^2, \ldots, z^2 \rangle_p$ regarded as a $p \times p$ matrix is, when read row-wise, equal to a sequence of consecutive ones, followed by zeros. In other words, using the notation t^σ from Lemma 9, we have $t(\mathbf{z}) = t^\sigma \langle k \rangle_n$ for some k. Also note that every almost rectangle of step size at most one has a p-ary cyclic shift of this form. Having this in mind, the following lemma is an easy consequence of the previous one.

Lemma 12. *Every near-threshold almost rectangle of step size at most one is tame.*

Proof. Let $\mathbf{z}$ be a near-threshold almost rectangle of step size at most one. Without loss of generality, assume it is of the form $\mathbf{z} = \langle z^2 + 1, \ldots, z^2 + 1, z^2, \ldots, z^2 \rangle_p$. Let k be the number of ones in $\mathbf{z}$. We know that $t(\mathbf{z}) = t^\sigma \langle k \rangle_n$. Since $\mathbf{z}$ is a near-threshold almost rectangle, we have

$$\left| \lambda(\mathbf{z}) - \frac{1}{2} \right| < \frac{1}{s^{\Delta z + 3}},$$

so

$$\lambda(\mathbf{z}) < \frac{1}{s^{\Delta z + 3}} + \frac{1}{2},$$

but we know

$$\lambda(\mathbf{z}) = \frac{k}{n}.$$

Since the step size is at most one, i.e., $\Delta z \in \{0, 1\}$, we have

$$\frac{k}{n} < \frac{1}{s^{\Delta z+3}} + \frac{1}{2} \leq \frac{1}{s^3} + \frac{1}{2},$$

so

$$k < 2\left\lfloor \frac{n}{2} \right\rfloor.$$

Now the result follows immediately from Lemma 11 □

Definition 18. *We say that an m-tuple of evaluations $\mathbf{k}_1 = \langle k_1^1, k_1^2, \ldots, k_1^p \rangle$, $\mathbf{k}_2 = \langle k_2^1, k_2^2, \ldots, k_2^p \rangle, \ldots, \mathbf{k}_m = \langle k_m^1, k_m^2, \ldots, k_m^p \rangle$, where $m \in [s]$, is* plausible *if $\sum_{j=1}^m k_j^i = rp$ for all $i \in [p]$.*

In other words, by arranging the integers defining $\mathbf{k}_1$, $\mathbf{k}_2$, …, $\mathbf{k}_m$ as rows of an $m \times p$ matrix, we get a matrix whose every column sums up to rp. Note that the sum of the areas of the evaluations is then equal to r.

The following lemma is a "2-dimensional analogue" of Lemma 10. The proof applies the first type of doubly cyclic identities from Definition 13.

Lemma 13. *If a tuple $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_s$ is plausible, then $(t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_s)) \in\leq (2r-1)$-in-$s$.*

Moreover, we have $t\langle p - k^1, p - k^2, \ldots, p - k^p \rangle = 1 - t\langle k^1, k^2, \ldots, k^p \rangle$ for any evaluation $\langle k^1, k^2, \ldots, k^p \rangle$.

Proof. Let $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_s$ be a plausible tuple. Fix, for a while, an arbitrary $i \in [p]$. Form a $s \times rp$ matrix M_i whose first row is $\langle k_1^i \rangle_{rp}$ and j-th row is the $(\sum_{l=1}^{j-1} k_l^i)$-th cyclic shift of $\langle k_j^i \rangle_{rp}$ for $j \in \{2, \ldots, s\}$. Split this matrix into r-many $s \times p$ blocks $M_i^1, M_i^2, \ldots, M_i^r$. Their sum $X_i = \sum_{j=1}^r M_i^j$ is an $s \times p$ matrix whose each column contains exactly r ones. Moreover, for all $j \in [s]$, the j-th row of the matrix X_i is a cyclic shift of $\langle k_j^i \rangle_p$. Put the matrices X_1,X_2, …, X_p aside to form an $s \times n$ matrix Y. Its rows have the same t-images as $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_s$, respectively, because t is doubly cyclic. Each column belongs to the relation r-in-s, therefore, as t is a polymorphism, we get that t applied to the rows gives a tuple in $\leq(2r-1)$-in-s. This tuple is equal to $(t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_s))$.

The second part can be proved in a similar way as the second part of Lemma 10 using the disequality relation pair. □

The next lemma will be applied to produce plausible sequence of evaluations. The proof uses the other type of doubly cyclic identities. First we provide a brief sketch and then the whole proof.

Lemma 14. *Let $\mathbf{z}$ be an almost rectangle of step size $\Delta z \geq 2$ with $|\lambda(\mathbf{z}) - 1/2| \leq 1/s^3$ and let p be sufficiently large. Then*

- *there exists a plausible $2r$-tuple $\mathbf{k}_1$, $\mathbf{k}_2$,…, $\mathbf{k}_{2r-1}$, $\mathbf{l}$ of almost rectangles such that $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{2r-1})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \cdots = \lambda(\mathbf{k}_{2r-1})$, and $\mathbf{l}$ has the same step size Δz as $\mathbf{z}$;*

- *there exists a plausible $2r$-tuple $\mathbf{k}_1$, $\mathbf{k}_2$,…, $\mathbf{k}_{2r-2}$, $\mathbf{l}_1, \mathbf{l}_2$ of almost rectangles such that $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{2r-2})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \lambda(\mathbf{k}_2) = \cdots = \lambda(\mathbf{k}_{2r-2})$, both $\mathbf{l}_1$ and $\mathbf{l}_2$ have step size strictly smaller than Δz, and $|\lambda(\mathbf{l}_1) - \lambda(\mathbf{l}_2)| \leq 1/p$.*

Proof sketch. We can assume that $\mathbf{z} = \langle c \times z^1, d \times z^2 \rangle$ for some c, d, z^1, z^2. For the first item, we consider the $(2r-1) \times p$ matrix X whose first row is $\mathbf{z}$ and the i-th row is the c-th cyclic shift of the $(i-1)$-st row for each $i \in \{2, \ldots, 2r-1\}$. Let Y be the $2r \times p$ matrix obtained from X by adding a row $(l^1, l^2, \ldots, l^p)$ so that each column sums up to rp and we define $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_m, \mathbf{l}$ as the n-tuples determined by the rows of Y via $\langle\rangle$, e.g., $\mathbf{l} = \langle l^1, l^2, \ldots, l^p \rangle$. The inequality $|\lambda(\mathbf{z}) - 1/2| \leq 1/s^3$ (and p being sufficiently large) ensures that $\mathbf{l}$ is correctly defined (i.e., all the l^i are between 0 and p), the construction gives that $\mathbf{l}$ is an almost rectangle with step size Δz and that $\mathbf{z}$ and $\mathbf{k}_i$ have equal areas, and the double cyclicity of t implies $t(\mathbf{z}) = t(\mathbf{k}_i)$. For the second item we additionally split the l row in two roughly equal rows. This will guarantee the two properties of $\mathbf{l}_1$ and $\mathbf{l}_2$. □

Proof. Without loss of generality we can assume that $\mathbf{z} = \langle c \times z^1, d \times z^2 \rangle$ for some c, d and $z^1 > z^2$. Let $m = 2r - 1$ for the first item and $m = 2r - 2$ for the second one. We define an integer $m \times p$ matrix X so that the first row is $(c \times z^1, d \times z^2)$ and the i-th row is the c-th cyclic shift of the $(i-1)$-st row for each $i \in \{2, \ldots, m\}$. Let Y be the $(m+1) \times p$ matrix obtained from X by adding a row $(l^1, l^2, \ldots, l^p)$ so that each column sums up to rp. It is easily seen by induction on $i \leq m$ that the sum of the first i rows is a cyclic shift of a tuple of the form $(e, \ldots, e, e', \ldots, e')$, where $|e - e'| = \Delta z$ and the "step down" is at position ci mod p (when columns are indexed from 0). It follows that $(l^1, l^2, \ldots, l^p)$ is also a cyclic shift of a tuple of the form $(e, \ldots, e, e', \ldots, e')$ where e and e' differ by Δz.

Next we observe that each $l^i > 0$ if p is sufficiently large. Indeed, note that since $|z^1 - z^2|/p$ can be made arbitrarily small (recall $|z^1 - z^2| < 5b$), we have $p(\lambda(\mathbf{z}) - \epsilon) < z^1, z^2 < p(\lambda(\mathbf{z}) + \epsilon)$, where $\epsilon > 0$ can be made arbitrarily small. Since $z^1, z^2 < p(\lambda(\mathbf{z}) + \epsilon)$ and for each i we have $l^i = rp - mz^1$ or $l^i = rp - mz^2$, then we have, for each i,

$$\begin{aligned}
l^i &> rp - mp(\lambda(\mathbf{z}) + \epsilon) \\
&\geq rp - (2r-1)p\left(\frac{1}{2} + \frac{1}{s^3} + \epsilon\right) \\
&= p\left(r - (2r-1)\left(\frac{1}{2} + \frac{1}{s^3} + \epsilon\right)\right) \\
&= p\left(r - \left(\frac{2r}{2} - \frac{1}{2}\right) - (2r-1)\left(\frac{1}{s^3} + \epsilon\right)\right) \\
&= p\left(\frac{1}{2} - (2r-1)\left(\frac{1}{s^3} + \epsilon\right)\right) \\
&> p\left(\frac{1}{2} - 2r\left(\frac{1}{s^3} + \epsilon\right)\right),
\end{aligned}$$

which is, for a sufficiently small ϵ, greater than 0 since $2r/s^3 < s/s^3 = 1/s^2 < 1/2$. Similarly, now using $p(\lambda(\mathbf{z}) - \epsilon) < z^1, z^2$, we get that each $l^i < p$ if $m = 2r - 1$ and $l^i < 3p/2$ if $m = 2r - 2$.

Now we can finish the proof of the first item. We set $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_m, \mathbf{l}$ to be the n-tuples determined by the rows of Y via $\langle\rangle$, e.g., $\mathbf{l} = \langle l^1, l^2, \ldots, l^p \rangle$. The inequalities $0 \leq l^i \leq p$ guarantee that $\mathbf{l}$ is correctly defined and we see, using also the double cyclicity of t (for $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_m)$), that these n-tuples have all the required properties.

To finish the proof of the second item, we define the $\mathbf{k}_i$ as above and set $\mathbf{l}_1 = \langle \lfloor l^1/2 \rfloor, \lfloor l^2/2 \rfloor, \ldots, \lfloor l^p/2 \rfloor \rangle$, $\mathbf{l}_2 = \langle \lceil l^1/2 \rceil, \lceil l^2/2 \rceil, \ldots, \lceil l^p/2 \rceil \rangle$. Since $0 \leq \lfloor l^i/2 \rfloor \leq \lceil l^i/2 \rceil \leq 3p/4 < p$, for $i = 1, 2, \ldots, p$, these tuples are correctly defined almost rectangles. Their areas clearly differ by at most $1/p$. As $\Delta z \geq 2$, their step sizes are strictly smaller than Δz, and we are done in this case as well. $\square$

Equipped with these lemmata we are ready to prove the following lemma.

Lemma 15. *Every near-threshold almost rectangle is tame.*

Proof. The proof is by induction on the step size. Step sizes zero and one are dealt with in Lemma 12, so we assume that $\mathbf{z}$ is a near-threshold almost rectangle of step size $2 \leq \Delta z < 5b$.

Assume first that $\lambda(\mathbf{z})$ is not *too close to* $1/2$, say, $|\lambda(\mathbf{z}) - 1/2| \geq 1/s^{5b+4}$. We apply the second item in Lemma 14 and get a plausible $2r$-tuple $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_{2r-2}, \mathbf{l}_1, \mathbf{l}_2$ such that $\mathbf{z}, \mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_{2r-2}$ all have the same t-images and areas, and $\mathbf{l}_1$ and $\mathbf{l}_2$ are almost rectangles with step sizes strictly smaller than Δz, whose areas differ by at most $1/p$.

The average area of almost rectangles $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_{2r-2}$, $\mathbf{l}_1$, $\mathbf{l}_2$ is $1/2$, the first $2r - 2$ of them have the same area as $\mathbf{z}$, bounded away from $1/2$ by a constant (namely $1/s^{5b+4}$), and the last two have almost the same area (the difference is at most $1/p$). By choosing a large enough p we get

$$\operatorname{sgn}\left(\lambda(\mathbf{l}_1) - \frac{1}{2}\right) = \operatorname{sgn}\left(\lambda(\mathbf{l}_2) - \frac{1}{2}\right) \neq \operatorname{sgn}\left(\lambda(\mathbf{z}) - \frac{1}{2}\right)$$

and

$$\left|\lambda(\mathbf{l}_i) - \frac{1}{2}\right| \leq 2r \cdot \left|\lambda(\mathbf{z}) - \frac{1}{2}\right|;$$

in particular, both $\mathbf{l}_i$ are near-threshold since

$$2r \cdot \left|\lambda(\mathbf{z}) - \frac{1}{2}\right| \leq s \cdot \left|\lambda(\mathbf{z}) - \frac{1}{2}\right| \leq \frac{1}{s^{\Delta z+3-1}} \leq \frac{1}{s^{\Delta l_i+3}}.$$

By the induction hypothesis, both $\mathbf{l}_i$ are tame. We complete the $2r$-tuple to an s-tuple by adding $s - 2r$ zeroes. The new s-tuple is also plausible, so we can apply Lemma 13. We get that $(t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_{2r-2}), t(\mathbf{l}_1), t(\mathbf{l}_2), t(\mathbf{0}), t(\mathbf{0}), \ldots, t(\mathbf{0})) \in \leq (2r-1)$-in-$s$. But $t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{2r-2})$ and $t(\mathbf{l}_1) = t(\mathbf{l}_2)$, and there are $2r$ of them, so they cannot all be one. Let $\overline{\mathbf{k}}$ be the "complementary" tuple of $\mathbf{k} = \langle k^1, k^2, \ldots, k^p \rangle$, i.e., $\overline{\mathbf{k}} = \langle p - k^1, p - k^2, \ldots, p - k^p \rangle$. The tuple $\overline{\mathbf{k}}_1$, $\overline{\mathbf{k}}_2$, $\ldots$, $\overline{\mathbf{k}}_{2r-2}$, $\overline{\mathbf{l}}_1$, $\overline{\mathbf{l}}_2$ is also plausible, and if we add $s - 2r$ zeroes, we again get a plausible s-tuple, on which we can apply Lemma 13, so we get $(t(\overline{\mathbf{k}}_1), t(\overline{\mathbf{k}}_2), \ldots, t(\overline{\mathbf{k}}_{2r-2}), t(\overline{\mathbf{l}}_1), t(\overline{\mathbf{l}}_2), t(\mathbf{0}), t(\mathbf{0}), \ldots, t(\mathbf{0})) \in \leq (2r-1)$-in-$s$. By the second part of Lemma 13, we know $t(\mathbf{k}_1) \neq t(\overline{\mathbf{k}}_1)$, $t(\mathbf{k}_2) \neq t(\overline{\mathbf{k}}_2)$, $\ldots$, $t(\mathbf{k}_{2r-2}) \neq t(\overline{\mathbf{k}}_{2r-2})$, $t(\mathbf{l}_1) \neq t(\overline{\mathbf{l}}_1)$, $t(\mathbf{l}_2) \neq t(\overline{\mathbf{l}}_2)$, so we also get $t(\overline{\mathbf{k}}_1) = t(\overline{\mathbf{k}}_2) = \cdots = t(\overline{\mathbf{k}}_{2r-2})$ and $t(\overline{\mathbf{l}}_1) = t(\overline{\mathbf{l}}_2)$. As there are $2r$ of them, they cannot all be one, which implies that $t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_{2r-2}), t(\mathbf{l}_1), t(\mathbf{l}_2)$ cannot all be zero. Hence, $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{2r-2}) \neq t(\mathbf{l}_1) = t(\mathbf{l}_2)$. We do the same thing to the "complementary" tuple of the tuple $\mathbf{k}_1$, $\mathbf{k}_2$, $\ldots$, $\mathbf{k}_{2r-2}$, $\mathbf{l}_1$, $\mathbf{l}_2$ - we add zeroes to get an s-tuple, we apply Lemma 13. Since $\mathrm{sgn}(\lambda(\mathbf{z}) - 1/2) \neq \mathrm{sgn}(\lambda(\mathbf{l}_1) - 1/2)$, it follows that $\mathbf{z}$ is tame, as required.

It remains to deal with the case that $\lambda(\mathbf{z})$ is too close to $1/2$. In this case we will find an almost rectangle $\mathbf{l}$ with the same step size as $\mathbf{z}$ such that $t(\mathbf{l}) = 1 - t(\mathbf{z})$ and $\lambda(\mathbf{l}) - 1/2 = -s'(\lambda(\mathbf{z}) - 1/2)$, where s' is such that $2 \leq s' \leq s$. If $\lambda(\mathbf{l})$ is already not too close to $1/2$, then we observe that $\mathbf{l}$ is near-threshold (indeed, $|\lambda(\mathbf{l}) - 1/2| \leq s|\lambda(\mathbf{z}) - 1/2| \leq s/s^{5b+4} \leq 1/s^{\Delta z+3}$) and apply to $\mathbf{l}$ the first part of the proof, thus obtaining that $\mathbf{l}$ is tame and, consequently, $\mathbf{z}$ is tame as well. If $\lambda(\mathbf{l})$ is still too close to $1/2$, then we simply repeat the process until we get a rectangle that is not too close.

To find such an almost rectangle $\mathbf{l}$ we apply the first item of Lemma 14 and get a plausible $2r$-tuple $\mathbf{k}_1, \ldots, \mathbf{k}_{2r-1}, \mathbf{l}$ such that $t(\mathbf{z}) = t(\mathbf{k}_1) = \cdots = t(\mathbf{k}_{2r-1})$ and $\mathbf{l}$ is an almost rectangle of the same step size as $\mathbf{z}$. Since the area of each $\mathbf{k}_i$ is equal to $\lambda(\mathbf{z})$ and the average area in the plausible $2r$-tuple is $1/2$, we get that $\lambda(\mathbf{l}) - 1/2 = -(2r-1)(\lambda(\mathbf{z}) - 1/2)$. By the same trick as previously, adding zeroes, using Lemma 13 and "complementary" tuples, we get that $t(\mathbf{l})$ and $t(\mathbf{z})$ are not equal. This concludes the construction of $\mathbf{l}$ and the proof of the lemma. $\square$

The proof can now be finished by using the tameness of near-threshold almost rectangles together with the b-boundedness of t as follows.

Let $m = (p-1)/2$ and choose positive integers $z^{2,1}$ and $z^{2,2}$ so that $p/2 - 2b < z^{2,1} < z^{2,2} < p/2$ and the x/y-tuples $(z^{2,1} \times x, (p - z^{2,1}) \times y)$ and $(z^{2,2} \times x, (p - z^{2,2}) \times y)$ are $\sim$-equivalent (see Definition 14 of boundedness). This is possible by the pigeonhole principle since there are more than b integers in the interval and $\sim$ has at most b classes.

By the choice of $z^{2,1}$ and $z^{2,2}$, for any meaningful choice of z^1, we have $t(\mathbf{z}_1) = t(\mathbf{z}_2)$ where $\mathbf{z}_i = \langle m \times z^1, (p-m) \times z^{2,i} \rangle_p$, $i = 1, 2$. We choose z^1 as the maximum number such that $\lambda(\mathbf{z}_1) < 1/2$. (Note here that for $z^1 = p$ the area of $\mathbf{z}_1$ can be made arbitrarily close to $(1 + 1/2)/2 > 1/2$ by choosing a sufficiently large p, so we may assume $z^1 < p$.) From $m < p/2$ it follows that increasing $z^{2,1}$ by one makes the area of $\mathbf{z}_1$ greater than increasing z^1 by one, therefore $\lambda(\mathbf{z}_2) > 1/2$.

Note that $z^1 > p/2$ since otherwise the area of $\mathbf{z}_2$ is less than $1/2$. On the other hand, $z^1 < p/2 + 3b$, otherwise the area of $\mathbf{z}_1$ is greater (assuming $p > 5$):

$$\lambda(\mathbf{z}_1) = \frac{mz^1 + (p-m)z^{2,1}}{p^2} \geq \frac{\frac{p-1}{2}(\frac{p}{2} + 3b) + \frac{p+1}{2}(\frac{p}{2} - 2b)}{p^2} = \frac{\frac{p^2}{2} + \frac{b(p-5)}{2}}{p^2} > \frac{1}{2}.$$

It follows that the step size of both $\mathbf{z}_1$ and $\mathbf{z}_2$ is less than $5b$, so both $\mathbf{z}_i$ are almost rectangles. By choosing a sufficiently large p, the difference $\lambda(\mathbf{z}_2) - \lambda(\mathbf{z}_1)$ can be made arbitrarily small, and since $\lambda(\mathbf{z}_1) < 1/2 < \lambda(\mathbf{z}_2)$ both $\mathbf{z}_i$ are then near-threshold.

Now the tameness of near-threshold almost rectangles (Lemma 15) gives us $t(\mathbf{z}_1) = t\langle 0 \rangle_n \neq 1 - t\langle 0 \rangle_n = t(\mathbf{z}_2)$. On the other hand, we also have $t(\mathbf{z}_1) = t(\mathbf{z}_2)$, a contradiction.

1.5 The other cases

1.5.1 Case (2): $\mathrm{PCSP}((\leq r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where s is even, $1 < r = s/2$

In this subsection we will prove that $\mathrm{PCSP}((\leq r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where s is even, $1 < r = s/2$ is not finitely tractable, which is (half of the) Case (2) in Theorem 2. Striving for a contradiction, suppose that it is finitely tractable. There exists b such that $\mathrm{PCSP}((\leq r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ has a p^2-ary b-bounded doubly cyclic polymorphism t for every sufficiently large arity p^2. We fix such a b and t, where p is fixed to a sufficiently large prime p congruent to 1 modulo s. How large must p be will be seen in due course. We denote $n = p^2$ and observe that $n \equiv 1 \pmod{s}$ as well.

We define t^σ as in Lemma 9, and by the same lemma we have that t^σ is cyclic.

The following lemma is analogous to Lemma 10 for Case (1), and the proof is almost the same.

Lemma 16. *Let $\langle k_1 \rangle, \langle k_2 \rangle, \ldots, \langle k_s \rangle$, where $0 \leq k_i \leq n$, be an s-tuple of n-tuples such that $\sum_{i=1}^{s} k_i \leq rn$. Then $(t^\sigma\langle k_1\rangle, t^\sigma\langle k_2\rangle, \ldots, t^\sigma\langle k_s\rangle) \in \leq (2r-1)$-in-$s$.*

Moreover, we have $t^\sigma\langle n-k\rangle = 1 - t^\sigma\langle k\rangle$ for every $0 \leq k \leq n$.

Proof. For the first part, form an $s \times rn$ matrix M whose first row is $\langle k_1\rangle_{rn}$ and the j-th row is the $(\sum_{l=1}^{j-1} k_l)$-th cyclic shift of $\langle k_j\rangle_{rn}$ for $j \in \{2, \ldots, s\}$. Note that each of the first $\sum k_i$ columns of M contains exactly 1 one and the remaining columns are all zero. Split this matrix into r-many $s \times n$ blocks $M^1, M^2, \ldots, M^r$. Their sum $X = \sum_{j=1}^{r} M^j$ is an $s \times n$ zero-one matrix whose each column contains at most r ones. Moreover, for all $j \in [s]$, the j-th row of X is a cyclic shift of $\langle k_j\rangle$, therefore its t^σ-image is $t^\sigma\langle k_j\rangle$ by cyclicity of t^σ. Each column belongs to the relation $\leq r$-in-s, therefore, as t^σ is a polymorphism, we get that t^σ applied to the rows gives a tuple in $\leq (2r-1)$-in-s. This implies the first claim.

The second part is proved in the same way as the second part of the analogous lemma for the Case (1) (Lemma 10), using only the disequality relation pair. □

The following lemma is analogous to Lemma 11 for Case (1), and the proof is significantly different.

Lemma 17. *Denote $a = \lfloor n/2 \rfloor$. For every $0 \leq k \leq 2a$, we have*

$$t^\sigma\langle k\rangle_n = \begin{cases} t^\sigma\langle 0\rangle_n & \text{if } 0 \leq k \leq a \\ 1 - t^\sigma\langle 0\rangle_n & \text{if } 1 + a \leq k \leq 2a \end{cases}$$

Proof. If $0 \leq k \leq a$, then we apply Lemma 16 to the s-tuple $\langle k\rangle, \langle k\rangle, \ldots, \langle k\rangle$ (we can do that because $sk \leq sa = s\lfloor n/2\rfloor \leq sn/2$, which is equal to rn since $s = 2r$) and we get that the tuple $(t^\sigma\langle k\rangle, t^\sigma\langle k\rangle, \ldots, t^\sigma\langle k\rangle)$ is in $\leq (2r-1)$-in-s; therefore $t^\sigma\langle k\rangle = 0$. For the remaining values $2a \geq k \geq a+1$ we apply the second part of the same lemma and get $t^\sigma\langle k\rangle = 1$. □

Tameness and almost rectangles are in this case defined in the same way as in Case (1). (Note here that the definitions are not really the same, because t depends on the template, that is, the case.)

Definition 19. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is* tame *if*

$$t(\mathbf{z}) = \begin{cases} t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) < 1/2 \\ 1 - t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) > 1/2 \end{cases}$$

Definition 20. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is an* almost rectangle *if it is a p-ary cyclic shift of a tuple of the form* $\langle z^1, \ldots, z^1, z^2, \ldots, z^2\rangle_p$*, where* $0 \le z^1, z^2 \le p$*, the number of* z^1*'s is arbitrary, and* $|z^1 - z^2| < 5b$*. The quantity* $\Delta z = |z^1 - z^2|$ *is referred to as the* step size. *We say that* $\mathbf{z}$ *is* near-threshold *if* $|\lambda(\mathbf{z}) - 1/2| < 1/s^{\Delta z + 3}$.

The following lemma is analogous to Lemma 12 in Case (1) and is proved in the same way, using Lemma 17 instead of 11.

Lemma 18. *Every near-threshold almost rectangle of step size at most one is tame.*

Now, as before, we want to generalize this result to arbitrary step size.

We define a plausible tuple as before.

Definition 21. *We say that an m-tuple of evaluations* $\mathbf{k}_1 = \langle k_1^1, k_1^2, \ldots, k_1^p\rangle$, $\mathbf{k}_2 = \langle k_2^1, k_2^2, \ldots, k_2^p\rangle, \ldots, \mathbf{k}_m = \langle k_m^1, k_m^2, \ldots,\ k_m^p\rangle$*, where* $m \in [s]$*, is* plausible *if* $\sum_{j=1}^m k_j^i = rp$ *for all* $i \in [p]$.

The following lemma is analogous to the Lemma 13 for Case (1) and the proof is the same, because r-in-$s \subseteq\le r$-in-s.

Lemma 19. *If a tuple* $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_s$ *is plausible, then* $(t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_s)) \in\le (2r-1)$*-in-*$s$.

Moreover, we have $t\langle p - k^1, p - k^2, \ldots, p - k^p\rangle = 1 - t\langle k^1, k^2, \ldots, k^p\rangle$ *for any evaluation* $\langle k^1, k^2, \ldots, k^p\rangle$.

The following lemma is analogous to the Lemma 14 for Case (1) and is proved in the same way, just put s instead of $2r$.

Lemma 20. *Let* $\mathbf{z}$ *be an almost rectangle of step size* $\Delta z \ge 2$ *with* $|\lambda(\mathbf{z}) - 1/2| \le 1/s^3$ *and let p be sufficiently large. Then*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \ldots,$ $\mathbf{k}_{s-1}$, $\mathbf{l}$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{s-1})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \cdots = \lambda(\mathbf{k}_{s-1})$*, and* $\mathbf{l}$ *has the same step size* Δz *as* $\mathbf{z}$*;*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \ldots,$ $\mathbf{k}_{s-2}$, $\mathbf{l}_1, \mathbf{l}_2$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{s-2})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \lambda(\mathbf{k}_2) = \cdots = \lambda(\mathbf{k}_{s-2})$*, both* $\mathbf{l}_1$ *and* $\mathbf{l}_2$ *have step size strictly smaller than* Δz*, and* $|\lambda(\mathbf{l}_1) - \lambda(\mathbf{l}_2)| \le 1/p$.

Now we are ready to prove the following lemma, analogous to Lemma 15 for Case (1).

Lemma 21. *Every near-threshold almost rectangle is tame.*

Proof. There is no need to write the whole proof here, because it is similar to the proof of Lemma 15, using the corresponding analogous lemmata that we just proved. It is just a bit simpler, because in this case $2r = s$, so we do not complete $2r$-tuples to s-tuples by adding zeroes, because they already are s-tuples, so we can immediately apply Lemma 19. $\square$

Now that we proved the tameness of almost rectangles, we can finish the proof in the same way as for Case (1), of course, now using Lemma 21 instead of Lemma 15.

1.5.2 Case (3): $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where s is even, $1 < r = s/2$, and r is even

In this subsection we will prove that $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ where s is even, $1 < r = s/2$, and r is even, is not finitely tractable, which is (half of the) Case (3) in Theorem 2. Striving for a contradiction, suppose that it is finitely tractable. There exists b such that $\mathrm{PCSP}((r\text{-in-}s, \leq (2r-1)\text{-in-}s), (\neq, \neq))$ has a p^2-ary b-bounded doubly cyclic polymorphism t for every sufficiently large arity p^2. We fix such a b and t, where p is fixed to a sufficiently large prime p congruent to 1 modulo s. How large must p be will be seen in due course. We denote $n = p^2$ and observe that $n \equiv 1 \pmod{s}$ as well.

We define t^σ as in Lemma 9, and by the same lemma we have that t^σ is cyclic.

The following lemma is analogous to Lemma 10 for Case (1), and the proof is the same.

Lemma 22. *Let $\langle k_1 \rangle, \langle k_2 \rangle, \ldots, \langle k_s \rangle$, where $0 \leq k_i \leq n$, be an s-tuple of n-tuples such that $\sum_{i=1}^{s} k_i = rn$. Then $(t^\sigma\langle k_1 \rangle, t^\sigma\langle k_2 \rangle, \ldots, t^\sigma\langle k_s \rangle) \in \leq (2r-1)$-in-$s$.*

Moreover, we have $t^\sigma\langle n-k \rangle = 1 - t^\sigma\langle k \rangle$ for every $0 \leq k \leq n$.

The following lemma is analogous to Lemma 11 for Case (1), and the proof is significantly different.

Lemma 23. *Denote $a = \lfloor n/2 \rfloor$. For every $0 \leq k \leq 2a$, we have*

$$t^\sigma\langle k \rangle_n = \begin{cases} t^\sigma\langle 0 \rangle_n & \text{if } 0 \leq k \leq a \\ 1 - t^\sigma\langle 0 \rangle_n & \text{if } 1 + a \leq k \leq 2a \end{cases}$$

Proof. We will prove, starting from the left, the following chain of disequalities.

$$t^\sigma\langle a \rangle \neq t^\sigma\langle a+1 \rangle \neq t^\sigma\langle a-1 \rangle \neq t^\sigma\langle a+2 \rangle \neq t^\sigma\langle a-2 \rangle \neq \ldots \neq t^\sigma\langle 2a \rangle \neq t^\sigma\langle 0 \rangle$$

This will imply $t^\sigma\langle a \rangle = t^\sigma\langle a-1 \rangle = \cdots = t^\sigma\langle 0 \rangle \neq t^\sigma\langle a+1 \rangle = t^\sigma\langle a+2 \rangle = \cdots = t^\sigma\langle 2a \rangle$. We start with the first disequality $t^\sigma\langle a \rangle \neq t^\sigma\langle a+1 \rangle$. The sequence of arguments

$$(s-r) \times \langle a \rangle,\ r \times \langle a+1 \rangle$$

has length s and is plausible as $(s-r)a + r(a+1) = sa + r$ and $sa + r$ is equal to rn. (Indeed, $n \equiv 1 \pmod{s}$, so $n = ms + 1$ for some integer m; then $a = \lfloor n/2 \rfloor = \lfloor (ms+1)/2 \rfloor = mr$ and $sa + r = smr + r = (n-1)r + r = rn$.) By Lemma 22, the t^σ image of this tuple belongs to $\leq (2r-1)$-in-s, so $t^\sigma\langle a \rangle$ and $t^\sigma\langle a+1 \rangle$ are not both ones. Also

$$(s-r) \times \langle n-a \rangle,\ r \times \langle n-(a+1) \rangle$$

is plausible, so by applying the same lemma to this tuple we get that $t^\sigma\langle n-a\rangle$ and $t^\sigma\langle n-(a+1)\rangle$ are not both ones. By the second part of Lemma 22 we can conclude that $t^\sigma\langle a\rangle \neq t^\sigma\langle a+1\rangle$.

For the second disequality $t^\sigma\langle a+1\rangle \neq t^\sigma\langle a-1\rangle$, we use the sequence

$$(s-r)/2 \times \langle a-1\rangle,\ (s+r)/2 \times \langle a+1\rangle$$

and derive $t^\sigma\langle a+1\rangle \neq t^\sigma\langle a-1\rangle$ using Lemma 22 as before.

To prove $t^\sigma\langle a-i+1\rangle \neq t^\sigma\langle a+i\rangle$ for $i \in \{2,3,\ldots,a\}$, we observe that, by the already established disequalities, we have $t^\sigma\langle a-i+1\rangle = \cdots = t^\sigma\langle a\rangle$, and then use

- $(s+r)/4 \times \langle a+i\rangle,\ (s-r)/2 \times \langle a-1\rangle,\ (s+r)/4 \times \langle a-i+2\rangle$ if $(s+r)/2$ is even;
- $(s+r+2)/4\times\langle a+i\rangle,\ (s-r-2)/2\times\langle a-1\rangle,\ 2\times\langle a-i+1\rangle,\ (s+r-6)/4\times\langle a-i+2\rangle$ if $(s+r)/2$ is odd.

Finally, for proving $t^\sigma\langle a+i\rangle \neq t^\sigma\langle a-i\rangle$ we use

$$(s-r)/2 \times \langle a-i\rangle,\ (s-r)/2 \times \langle a+i\rangle,\ r \times \langle a+1\rangle$$

This completes the proof. □

We define tameness and almost rectangles in the same way as before.

Definition 22. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is* tame *if*

$$t(\mathbf{z}) = \begin{cases} t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) < 1/2 \\ 1-t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) > 1/2 \end{cases}$$

Definition 23. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is an* almost rectangle *if it is a p-ary cyclic shift of a tuple of the form $\langle z^1,\ldots,z^1,z^2,\ldots,z^2\rangle_p$, where $0 \le z^1, z^2 \le p$, the number of z^1's is arbitrary, and $|z^1-z^2| < 5b$. The quantity $\Delta z = |z^1-z^2|$ is referred to as the* step size. *We say that* $\mathbf{z}$ *is* near-threshold *if* $|\lambda(\mathbf{z})-1/2| < 1/s^{\Delta z+3}$.

The following lemma is analogous to Lemma 12 in Case (1) and is proved in the same way, using Lemma 23 instead of Lemma 11.

Lemma 24. *Every near-threshold almost rectangle of step size at most one is tame.*

We define a plausible tuple as before.

Definition 24. *We say that an m-tuple of evaluations $\mathbf{k}_1 = \langle k_1^1,k_1^2,\ldots,k_1^p\rangle$, $\mathbf{k}_2 = \langle k_2^1,k_2^2,\ldots,k_2^p\rangle$, …, $\mathbf{k}_m = \langle k_m^1,k_m^2,\ldots,k_m^p\rangle$, where $m \in [s]$, is* plausible *if $\sum_{j=1}^m k_j^i = rp$ for all $i \in [p]$.*

The following lemma is analogous to the Lemma 13 for Case (1) and the proof is the same.

Lemma 25. *If a tuple* $\mathbf{k}_1, \mathbf{k}_2, \ldots, \mathbf{k}_s$ *is plausible, then* $(t(\mathbf{k}_1), t(\mathbf{k}_2), \ldots, t(\mathbf{k}_s)) \in \leq (2r-1)$*-in-s.*

Moreover, we have $t\langle p-k^1, p-k^2, \ldots, p-k^p\rangle = 1 - t\langle k^1, k^2, \ldots, k^p\rangle$ *for any evaluation* $\langle k^1, k^2, \ldots, k^p\rangle$.

The following lemma is analogous to Lemma 20 for Case (2) and is proved in the same way.

Lemma 26. *Let* $\mathbf{z}$ *be an almost rectangle of step size* $\Delta z \geq 2$ *with* $|\lambda(\mathbf{z}) - 1/2| \leq 1/s^3$ *and let* p *be sufficiently large. Then*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \ldots,$ $\mathbf{k}_{s-1}$, $\mathbf{l}$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{s-1})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \cdots = \lambda(\mathbf{k}_{s-1})$, *and* $\mathbf{l}$ *has the same step size* Δz *as* $\mathbf{z}$*;*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \ldots,$ $\mathbf{k}_{s-2}$, $\mathbf{l}_1, \mathbf{l}_2$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_{s-2})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \lambda(\mathbf{k}_2) = \cdots = \lambda(\mathbf{k}_{s-2})$, *both* $\mathbf{l}_1$ *and* $\mathbf{l}_2$ *have step size strictly smaller than* Δz, *and* $|\lambda(\mathbf{l}_1) - \lambda(\mathbf{l}_2)| \leq 1/p$.

Now we are ready to prove the following lemma, analogous to Lemma 15 for Case (1) and Lemma 21 for Case (2). The proof is the same as for Case (2).

Lemma 27. *Every near-threshold almost rectangle is tame.*

Now that we proved the tameness of almost rectangles, we can finish the proof in the same way as before, of course, now using Lemma 27.

1.5.3 Case (4): PCSP(r-in-s, not-all-equal-s) where $r \leq s/2$, $s > 2$, and r is even or s is odd

The whole Case (4) is PCSP(r-in-s, not-all-equal-s) where $r < s$, $s > 2$, and r is even or s is odd, but here we will prove not finite tractability only for when $r \leq s/2$, and the rest will follow as explained in Subsection 1.3.3.

Striving for a contradiction, suppose that it is finitely tractable. There exists b such that PCSP(r-in-s, not-all-equal-s) has a p^2-ary b-bounded doubly cyclic polymorphism t for every sufficiently large arity p^2. We fix such a b and t, where p is fixed to a sufficiently large prime p congruent to 1 modulo s. How large must p be will be seen in due course. We denote $n = p^2$ and observe that $n \equiv 1 \pmod{s}$ as well.

We define t^σ as before (Lemma 9), and by the same lemma we have that t^σ is cyclic.

The following lemma is analogous to the first part of Lemma 10 for Case (1), and the proof is the same. Note here that we do not have the second part of the lemma, as there is no disequality pair in the template.

Lemma 28. *Let* $\langle k_1\rangle, \langle k_2\rangle, \ldots, \langle k_s\rangle$, *where* $0 \leq k_i \leq n$, *be an s-tuple of n-tuples such that* $\sum_{i=1}^{s} k_i = rn$. *Then* $(t^\sigma\langle k_1\rangle, t^\sigma\langle k_2\rangle, \ldots, t^\sigma\langle k_s\rangle) \in$ *not-all-equal-s.*

The following lemma is analogous to Lemma 23 for Case (3), and the proof is similar.

Lemma 29. *Denote $a = \lfloor rn/s \rfloor$. For every $0 \leq k \leq 2a$, we have*

$$t^\sigma \langle k \rangle_n = \begin{cases} t^\sigma \langle 0 \rangle_n & \text{if } 0 \leq k \leq a \\ 1 - t^\sigma \langle 0 \rangle_n & \text{if } 1 + a \leq k \leq 2a \end{cases}$$

Proof. We will prove, starting from the left, the following chain of disequalities.

$$t^\sigma \langle a \rangle \neq t^\sigma \langle a+1 \rangle \neq t^\sigma \langle a-1 \rangle \neq t^\sigma \langle a+2 \rangle \neq t^\sigma \langle a-2 \rangle \neq \ldots \neq t^\sigma \langle 2a \rangle \neq t^\sigma \langle 0 \rangle$$

This will imply $t^\sigma \langle a \rangle = t^\sigma \langle a-1 \rangle = \cdots = t^\sigma \langle 0 \rangle \neq t^\sigma \langle a+1 \rangle = t^\sigma \langle a+2 \rangle = \cdots = t^\sigma \langle 2a \rangle$. We start with the first disequality $t^\sigma \langle a \rangle \neq t^\sigma \langle a+1 \rangle$. The sequence of arguments

$$(s-r) \times \langle a \rangle,\ r \times \langle a+1 \rangle$$

has length s and is plausible as $(s-r)a + r(a+1) = sa + r$ and $sa + r$ is equal to rn. (Indeed, $n \equiv 1 \pmod{s}$, so $n = ms + 1$ for some integer m; then $a = mr$ and $sa + r = smr + r = (n-1)r + r = rn$.) By Lemma 28, $t^\sigma \langle a \rangle \neq t^\sigma \langle a+1 \rangle$.

For the second disequality $t^\sigma \langle a+1 \rangle \neq t^\sigma \langle a-1 \rangle$, as well as for the further disequalities we need to distinguish two cases: Case (4a) r and s have the same parity and Case (4b) r is even and s is odd. In Case (4a) we directly use the sequence

$$(s-r)/2 \times \langle a-1 \rangle,\ (s+r)/2 \times \langle a+1 \rangle$$

and derive $t^\sigma \langle a+1 \rangle \neq t^\sigma \langle a-1 \rangle$ using Lemma 28 as before. In Case (4b) we first use

$$(s-1) \times \langle a \rangle,\ \langle a+r \rangle$$

to deduce $t^\sigma \langle a+r \rangle \neq t^\sigma \langle a \rangle$ (so $t^\sigma \langle a+1 \rangle = t^\sigma \langle a+r \rangle$) and then

$$(s-1)/2 \times \langle a-1 \rangle,\ (s-1)/2 \times \langle a+1 \rangle,\ \langle a+r \rangle$$

to deduce $t^\sigma \langle a-1 \rangle \neq t^\sigma \langle a+1 \rangle$.

To prove $t^\sigma \langle a-i+1 \rangle \neq t^\sigma \langle a+i \rangle$ for $i \in \{2, 3, \ldots, a\}$, we observe that, by the already established disequalities, we have $t^\sigma \langle a-i+1 \rangle = \cdots = t^\sigma \langle a \rangle$, and then use

- $(s+r)/4 \times \langle a+i \rangle,\ (s-r)/2 \times \langle a-1 \rangle,\ (s+r)/4 \times \langle a-i+2 \rangle$ in Case (4a) and $(s+r)/2$ is even;
- $(s+r+2)/4 \times \langle a+i \rangle$, $(s-r-2)/2 \times \langle a-1 \rangle$, $2 \times \langle a-i+1 \rangle$, $(s+r-6)/4 \times \langle a-i+2 \rangle$ in Case (4a) and $(s+r)/2$ is odd;
- $r/2 \times \langle a+i \rangle,\ (s-r) \times \langle a \rangle,\ r/2 \times \langle a-i+2 \rangle$ in Case (4b).

Finally, for proving $t^\sigma \langle a+i \rangle \neq t^\sigma \langle a-i \rangle$ we use

- $(s-r)/2 \times \langle a-i \rangle,\ (s-r)/2 \times \langle a+i \rangle,\ r \times \langle a+1 \rangle$ in Case (4a) and
- $(s-1)/2 \times \langle a-i \rangle,\ (s-1)/2 \times \langle a+i \rangle,\ 1 \times \langle a+r \rangle$ in Case (4b).

□

We define tameness and near-threshold almost rectangles in a slightly different way than before.

Definition 25. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is* tame *if*

$$t(\mathbf{z}) = \begin{cases} t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) < r/s \\ 1 - t\langle 0\rangle_n & \text{if } \lambda(\mathbf{z}) > r/s \end{cases}$$

Definition 26. *A tuple* $\mathbf{z} \in \{0,1\}^n$ *is an* almost rectangle *if it is a p-ary cyclic shift of a tuple of the form* $\langle z^1, \dots, z^1, z^2, \dots, z^2\rangle_p$, *where* $0 \leq z^1, z^2 \leq p$, *the number of* z^1*'s is arbitrary, and* $|z^1 - z^2| < 5b$. *The quantity* $\Delta z = |z^1 - z^2|$ *is referred to as the* step size. *We say that* $\mathbf{z}$ *is* near-threshold *if* $|\lambda(\mathbf{z}) - r/s| < 1/s^{\Delta z+3}$.

The following lemma is analogous to Lemma 12 in Case (1) and is proved in the same way, using Lemma 29 instead of Lemma 11.

Lemma 30. *Every near-threshold almost rectangle of step size at most one is tame.*

We define a plausible tuple as before.

Definition 27. *We say that an m-tuple of evaluations* $\mathbf{k}_1 = \langle k_1^1, k_1^2, \dots, k_1^p\rangle$, $\mathbf{k}_2 = \langle k_2^1, k_2^2, \dots, k_2^p\rangle$, $\dots$, $\mathbf{k}_m = \langle k_m^1, k_m^2, \dots, k_m^p\rangle$, *where* $m \in [s]$, *is* plausible *if* $\sum_{j=1}^{m} k_j^i = rp$ *for all* $i \in [p]$.

The following lemma is analogous to the first part of Lemma 13 for Case (1) and the proof is almost the same. Again, notice that we do not have the second part of the lemma, because there is no disequality pair in the template.

Lemma 31. *If a tuple* $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_s$ *is plausible, then* $(t(\mathbf{k}_1), t(\mathbf{k}_2), \dots, t(\mathbf{k}_s)) \in$ *not-all-equal-*s.

The following lemma is analogous to Lemma 20 for Case (2) and Lemma 26 for Case (3) and is proved in a similar way.

Lemma 32. *Let* $\mathbf{z}$ *be an almost rectangle of step size* $\Delta z \geq 2$ *with* $|\lambda(\mathbf{z}) - r/s| \leq 1/s^3$ *and let* p *be sufficiently large. Then*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \dots$, $\mathbf{k}_{s-1}$, $\mathbf{l}$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \dots = t(\mathbf{k}_{s-1})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \dots = \lambda(\mathbf{k}_{s-1})$, *and* $\mathbf{l}$ *has the same step size* Δz *as* $\mathbf{z}$*;*

- *there exists a plausible s-tuple* $\mathbf{k}_1$, $\mathbf{k}_2, \dots$, $\mathbf{k}_{s-2}$, $\mathbf{l}_1, \mathbf{l}_2$ *of almost rectangles such that* $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \dots = t(\mathbf{k}_{s-2})$, $\lambda(\mathbf{z}) = \lambda(\mathbf{k}_1) = \lambda(\mathbf{k}_2) = \dots = \lambda(\mathbf{k}_{s-2})$, *both* $\mathbf{l}_1$ *and* $\mathbf{l}_2$ *have step size strictly smaller than* Δz, *and* $|\lambda(\mathbf{l}_1) - \lambda(\mathbf{l}_2)| \leq 1/p$.

Proof. Without loss of generality we can assume that $\mathbf{z} = \langle c \times z^1, d \times z^2\rangle$ for some c, d and $z^1 > z^2$. Let $m = s-1$ for the first item and $m = s-2$ for the second one. We define an integer $m \times p$ matrix X so that the first row is $(c \times z^1, d \times z^2)$ and the i-th row is the c-th cyclic shift of the $(i-1)$-st row for each $i \in \{2, \dots, m\}$. Let Y be the $(m+1) \times p$ matrix obtained from X by adding a row $(l^1, l^2, \dots, l^p)$ so that each column sums up to rp. It is easily seen by induction on $i \leq m$ that the sum of the first i rows is a cyclic shift of a tuple of the form $(e, \dots, e, e', \dots, e')$, where $|e - e'| = \Delta z$ and the "step down" is at position $ci \mod p$ (when columns are indexed from 0). It follows that $(l^1, l^2, \dots, l^p)$ is also a cyclic shift of a tuple of the form $(e, \dots, e, e', \dots, e')$ where e and e' differ by Δz.

Next we observe that each $l^i > 0$ if p is sufficiently large. Indeed, note that since $|z^1 - z^2|/p$ can be made arbitrarily small (recall $|z^1 - z^2| < 5b$), we have $p(\lambda(\mathbf{z}) - \epsilon) < z^1, z^2 < p(\lambda(\mathbf{z}) + \epsilon)$, where $\epsilon > 0$ can be made arbitrarily small. We then have $l^i > rp - mp(\lambda(\mathbf{z}) + \epsilon) \geq rp - (s-1)p(r/s + 1/s^3 + \epsilon) = p(r/s - (s-1)(1/s^3 + \epsilon)) > p(r/s - s(1/s^3 + \epsilon))$, which is, for a sufficiently small ϵ, greater than 0 since $s/s^3 = 1/s^2 < r/s$. Similarly, each $l^i < 2rp/s \leq p$ if $m = s-1$ and $l^i < 3rp/s$ if $m = s-2$.

Now we can finish the proof of the first item. We set $\mathbf{k}_1, \mathbf{k}_2, \dots, \mathbf{k}_m, \mathbf{l}$ to be the n-tuples determined by the rows of Y via $\langle\rangle$, e.g., $\mathbf{l} = \langle l^1, l^2, \dots, l^p \rangle$. The inequalities $0 \leq l^i \leq p$ guarantee that $\mathbf{l}$ is correctly defined and we see, using also the double cyclicity of t (for $t(\mathbf{z}) = t(\mathbf{k}_1) = t(\mathbf{k}_2) = \cdots = t(\mathbf{k}_m)$), that these n-tuples have all the required properties.

To finish the proof of the second item, we define the $\mathbf{k}_i$ as above and set $\mathbf{l}_1 = \langle \lfloor l^1/2 \rfloor, \lfloor l^2/2 \rfloor, \dots, \lfloor l^p/2 \rfloor \rangle, \mathbf{l}_2 = \langle \lceil l^1/2 \rceil, \lceil l^2/2 \rceil, \dots, \lceil l^p/2 \rceil \rangle$. Since $0 \leq \lfloor l^i/2 \rfloor \leq \lceil l^i/2 \rceil \leq 3rp/2s < p$, these tuples are correctly defined almost rectangles. Their areas clearly differ by at most $1/p$. As $\Delta z \geq 2$, their step sizes are strictly smaller than Δz, and we are done in this case as well. $\square$

We are now ready to prove the following lemma, analogous to Lemma 15 for Case (1), Lemma 21 for Case (2) and Lemma 27 for Case (3).

Lemma 33. *Every near-threshold almost rectangle is tame.*

Proof. The proof is like for the Case (2), just a bit simpler, without using "complementary" tuples. $\square$

Now we can finish the proof in the same way as before, of course, now using Lemma 33.

1.6 Conclusion

We have characterized finite tractability among the basic tractable cases in the Brakensiek–Guruswami classification [BG18] of symmetric Boolean PCSPs allowing negations. A natural direction for future research is an extension to all the tractable cases (not just the basic ones), or even to all symmetric Boolean PCSPs [FKOS19], not only those allowing negations. An obstacle, where our efforts have failed so far, is already in relaxations of the basic templates (P, Q) with disequalities. For example, which (P, Q), $(\neq, \neq)$, with P a subset of $\leq r$-in-s and Q a superset of $\leq(2r-1)$-in-s, give rise to finitely tractable PCSPs?

Another natural direction is to better understand the "level of tractability." For the finitely tractable templates $(\mathbb{A}, \mathbb{B})$ considered in this paper, it is always possible to find a tractable CSP($\mathbb{C}$) with $\mathbb{A} \to \mathbb{C} \to \mathbb{B}$ and such that $\mathbb{C}$ is two-element. Is it so for all symmetric Boolean templates? For general Boolean templates, the answer is "No": [DSM+20] presents an example that requires a three-element $\mathbb{C}$. However, it is unclear whether there is an upper bound on the size of $\mathbb{C}$ for finitely tractable (Boolean) PCSPs, and if there is, how it could be computed. There are also natural concepts beyond finite tractability, still stronger than standard tractability. We refer to [Bar19] for some questions in this direction.

1.A Basic cases

In this section we will prove the following theorem, already stated in Subsection 1.1.1.

Theorem 34. *Every tractable symmetric Boolean PCSP allowing negations can be obtained by*

- *taking any number of relation pairs from one of the following three items:*

  (a) $(\textit{odd-in-}s, \textit{odd-in-}s)$, *or* $(\textit{even-in-}s, \textit{even-in-}s)$

  (b) $(\leq r\textit{-in-}s, \leq (2r-1)\textit{-in-}s)$ *and* $r \leq s/2$, *or*
  $(\geq r\textit{-in-}s, \geq (2r-s+1)\textit{-in-}s)$ *and* $r \geq s/2$, *or*
  $(\frac{s}{2}\textit{-in-}s, \textit{not-all-equal-}s)$ *and* s *is even*

  (c) $(r\textit{-in-}s, \textit{not-all-equal-}s)$

  where r *and* s *are positive integers,*

- *adding any number of "trivial" relation pairs* (P, Q) *such that* $P \subseteq Q$, *and* Q *is the full relation or* P *contains only constant tuples, and*

- *taking a homomorphic relaxation of the obtained template.*

Recall that we can describe a PCSP template $(\mathbb{A} = (\{0,1\}; P_1, P_2, \ldots, P_n), \mathbb{B} = (\{0,1\}; Q_1, Q_2, \ldots, Q_n))$ by the collection of pairs $\Gamma = \{(P_1, Q_1), (P_2, Q_2), \ldots, (P_n, Q_n)\}$.

In [BG18] Brakensiek and Guruswami give a complexity classification of symmetric Boolean templates allowing negations in terms of the polymorphisms. The functions they are considering are *Parity*, *Majority*, *Alternating-Threshold*, and their *'anti-'* functions.

Let L be a positive odd integer, and let $x = (x_1, x_2, \ldots, x_L) \in \{0,1\}^L$.

- The Parity function: $Par_L(x) = \begin{cases} 1, \text{ if } \Sigma_{i=1}^L x_i \text{ is odd} \\ 0, \text{ otherwise} \end{cases}$.

- The Majority function: $Maj_L(x) = \begin{cases} 1, \text{ if } \Sigma_{i=1}^L x_i > L/2 \\ 0, \text{ otherwise} \end{cases}$.

- The Alternating-Threshold function: $AT_L = \begin{cases} 1, \text{ if } \Sigma_{i=1}^L (-1)^{i-1} x_i > 0 \\ 0, \text{ otherwise} \end{cases}$.

The prefix 'anti-' refers to the negations of these functions. The 'anti-' function will be denoted with a horizontal bar. For example, *anti-parity* is

$$\overline{Par}_L(x) = \begin{cases} 0, \text{ if } \Sigma_{i=1}^L x_i \text{ is odd} \\ 1, \text{ otherwise} \end{cases}.$$

Theorem 35. *Let* Γ *be a symmetric Boolean* PCSP *template allowing negations. If at least one of* Par_L, Maj_L, AT_L, $\overline{Par}_L$, $\overline{Maj}_L$, $\overline{AT}_L$ *is a polymorphism of* Γ *for all odd* L, *then* PCSP(Γ) *is polynomial-time tractable. Otherwise,* PCSP(Γ) *is NP-hard.*

Each symmetric Booelan relation is uniquely determined by its arity and the Hamming weights of the elements. We let $\mathrm{Ham}_k(S) = \{x \in \{0,1\}^k : |x| \in S\}$, where $|x|$ is the number of ones in x, denote these sets. For example, $\neq = \{(0,1),(1,0)\} = \mathrm{Ham}_2(\{1\})$.

Definition 28. *Let $f : \{0,1\}^L \to \{0,1\}$ be a function and let $P \subseteq \{0,1\}^k$ be a relation. Define $O_f(P)$ to be*

$$O_f(P) := f(P^L) = \{x \in \{0,1\}^k : \textit{exist } x^1, x^2, \ldots, x^L \in P \textit{ such that } x_i = f(x_i^1, x_i^2, \ldots, x_i^L) \textit{ for all } i \in [k]\}.$$

Note that f is a polymorphism of (P,Q) if and only if $O_f(P) \subseteq Q$. Define

$$O_{AT}(P) = \bigcup_{L\in\mathbb{N},\ L \text{ odd}} O_{AT_L}(P),$$

$$O_{Maj}(P) = \bigcup_{L\in\mathbb{N},\ L \text{ odd}} O_{Maj_L}(P),$$

$$O_{Par}(P) = \bigcup_{L\in\mathbb{N},\ L \text{ odd}} O_{Par_L}(P).$$

The following two lemmas, regarding Alternating-Threshold and Majority, are proved in [BG17] (Claim 4.6. and Claim 4.8.).

Lemma 36. *Let $k \geq 2$, and let $P = \mathrm{Ham}_k(S)$.*

- *If $S = \{l\}$ where $l \in \{1,2,\ldots,k-1\}$, then $O_{AT}(P) = \mathrm{Ham}_k(\{1,2,\ldots,k-1\})$.*
- *If S contains two different elements $l_1, l_2 \in \{0,1,\ldots,k\}$ such that $\{l_1,l_2\} \neq \{0,k\}$, then $O_{AT}(P) = \{0,1\}^k$.*

Lemma 37. *Consider $k \geq 2$. Let $P = \mathrm{Ham}_k(S)$.*

- *If S contains $0 < l < k/2$, then $O_{Maj}(P) = \mathrm{Ham}_k(\{0,\ldots,2\max S - 1\})$.*
- *If S contains $k/2 < l < k$, then $O_{Maj}(P) = \mathrm{Ham}_k(\{2\min S - k + 1,\ldots,k\})$.*
- *If $S = \{k/2\}$, then $O_{Maj}(P) = \mathrm{Ham}_k(\{1,\ldots,k-1\})$.*

We prove the following analogue lemma for Parity.

Lemma 38. *Let $k \geq 1$, and let $P = \mathrm{Ham}_k(S)$.*

- *If S contains an odd $l < k$, then $O_{Par}(P) \supseteq$ odd-in-k.*
- *If S contains an even $0 < l < k$, then $O_{Par}(P) \supseteq$ even-in-k.*

Proof.
- Let $l < k$ be an odd number in S. Let $0 < m \leq k$ be an odd number.

  If $m > l$, we form a $k \times (m - l + 1)$ matrix

$$M = \begin{pmatrix} A \\ B \\ C \end{pmatrix},$$

where A is an $(l-1) \times (m-l+1)$-matrix with all ones, B is a $(m-l+1) \times (m-l+1)$-matrix with ones on the main diagonal and zeroes everywhere else, and C is a $(k-m) \times (m-l+1)$-matrix with all zeroes. Each column of the matrix M is a k-tuple with hamming weight l, so it belongs to P. When we apply Par_{m-l+1} to the rows of M, we get a k-tuple whose first m coordinates are ones. Since P is symmetric, we can permute the rows of the matrix as we want, the columns will still be in P, so we get all k-tuples with m ones.

If $m < l$, we form a $k \times (l-m+1)$-matrix

$$M = \begin{pmatrix} A \\ B \\ C \end{pmatrix},$$

where A is an $m \times (l-m+1)$-matrix with all ones, B is a $(l-m+1) \times (l-m+1)$-matrix with zeroes on the main diagonal and ones everywhere else, and C is a $(k-l-1) \times (l-m+1)$-matrix with all ones. Every column of the matrix M is a tuple with l ones, so it belongs to P. When we apply Par_{l-m+1} to the rows of M, we get a k-tuple with m ones.

So, we proved that, if S contains an odd number less than k, then $O_{Par}(P)$ contains all k-tuples with odd number of ones.

- Let $0 < l < k$ be an even number in S. Let $0 \leq m \leq k$ be an even number. The same proof as above works also in this case, we get that $O_{Par}(P)$ contains all k-tuples with m number of ones. But m was an arbitrary even number, so $O_{Par}(P)$ contains all k-tuples with even number of ones.

□

Equipped with these lemmata, we can prove Theorem 34.

Proof of Theorem 34. Let Γ be a symmetric Boolean PCSP template allowing negations such that PCSP(Γ) is solvable in polynomial time. Then by Theorem 35 (as we assumed P $\neq$ NP) we have that at least one of Par_L, Maj_L, AT_L, $\overline{Par}_L$, $\overline{Maj}_L$, $\overline{AT}_L$ is a polymorphism of Γ for all odd L.

Assume Par_L is a polymorphism of Γ for all odd L. Take a pair $(P, Q) \in \Gamma$ such that P contains a not-all-equal tuple. (If such a pair does not exist, then every relational pair in Γ is trivial.) If P contains a not-all-equal tuple x such that $|x|$ is odd, then by the first item in Lemma 38 we get that odd-in-$k \subseteq Q$. If it contains a not-all-equal tuple x such that $|x|$ is even, then by the second item in Lemma 38 we get that even-in-$k \subseteq Q$. So, if $P \subseteq$ odd-in-k, then odd-in-$k \subseteq Q$, and, if $P \subseteq$ even-in-k, then even-in-$k \subseteq Q$. Otherwise, we get that Q is full, so the pair (P, Q) is trivial. We conclude that Γ is a homomorphic relaxation of a template obtained by taking any number of relation pairs from the item a) in Theorem 34, and adding any number of trivial relation pairs, where the homomorphisms in forming the homomorphic relaxation are identity mappings.

Assume that Maj_L is a polymorphism of Γ for all odd L. Take a $(P, Q) \in \Gamma$ such that P contains a not-all-equal tuple. Let k and S be such that $P = \mathrm{Ham}_k(S)$. If S contains $0 < l < k/2$, then by the first item in Lemma 37 we have that $\leq (2\max S - 1)$-in-$k \subseteq Q$; also, by the definition of S, we have

$P \subseteq\leq (\max S)$-in-k. So, $P \subseteq\leq (\max S)$-in-$k \subseteq\leq (2\max S - 1)$-in-$k \subseteq Q$. If S contains $k/2 < l < k$, then by the second item in Lemma 37 we get $P \subseteq\geq \min S$-in-$k \subseteq\geq (2\min S - k + 1)$-in-$k \subseteq Q$. Otherwise, $S = \{k/2\}$, and then by the third item in Lemma 37 we get $P = \frac{k}{2}$-in-$k \subseteq$ not-all-equal-$k \subseteq Q$. We can conclude that Γ is a homomorphic relaxation of a template obtained by taking any number of relation pairs from the item b) in Theorem 34, and adding any number of trivial relation pairs, where the homomorphisms are identity mappings.

Assume that AT_L is a polymorphism of Γ for all odd L. Take a $(P, Q) \in \Gamma$ such that P contains a not-all-equal tuple. Let k and S be such that $P = \mathrm{Ham}_k(S)$. If S contains only one element, say l, then by the first item in Lemma 36 we get that not-all-equal-$k \subseteq Q$, so, in that case $P = l$-in-$k \subseteq$ not-all-equal-$k \subseteq Q$. If S contains more elements, by the second item in Lemma 36 we get that Q is full, so (P, Q) is a trivial pair of relations. We conclude that Γ is a homomorphic relaxation of a template obtained by taking any number of relation pairs from the item c) in Theorem 34, and adding any number of trivial relation pairs, where the homomorphisms are identity mappings.

If one of $\overline{Par}_L$, $\overline{Maj}_L$, $\overline{AT}_L$ is a polymorphism of Γ for all odd L, we use the same reasoning, just that in this case, the second homomorphism in forming the homomorphic relaxation will be negation instead of identity. For example, assume that $\overline{Par}_L$ is a polymorphism of Γ for all odd L, and take a pair $(P, Q) \in \Gamma$ such that P contains a not-all-equal tuple. If P contains a tuple x such that $|x|$ is odd, then by Lemma 38 we have that odd-in-$k \subseteq O_{Par}(P)$. The "negation" function that swaps zeroes and ones homomorphically maps $O_{Par}(P)$ to $O_{\overline{Par}}(P)$. Since $\overline{Par}_L$ is a polymorphism of (P, Q) for all odd L, we have that $O_{\overline{Par}_L}(P) \subseteq Q$ for all odd L, hence, by definition, $O_{\overline{Par}}(P) \subseteq Q$. If P contains a tuple x such that $|x|$ is even, then by Lemma 38 we have that even-in-$k \subseteq O_{Par}(P)$, which is homomorphically mapped to $O_{\overline{Par}}(P)$ by the negation function, and $O_{\overline{Par}}(P) \subseteq Q$. We conclude that Γ is a homomorphic relaxation of a template obtained by taking any number of relation pairs from the item a) in Theorem 34, and adding any number of trivial relation pairs, where the first homomorphism in forming the homomorphic relaxation is the identity function and the second one is the negation function. $\square$

2. Fixed-Template Promise Model Checking Problems

2.1 Introduction

In this chapter we will adopt the logical formulation for the constraint satisfaction problem, which we recall here. 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 $\exists$.

Motivated by recent developments in the area of CSP, we study an extension of this framework in two simultaneous directions. One direction is to enable other choices of permitted quantifiers and connectives. Another direction is to consider two versions of each relation, strong and weak (a so-called promise problem). Our contributions are described in Subsection 2.1.1.

The model checking problem [MM18] 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 $\mathcal{L}$-MC($\mathbb{A}$).

It turns out that only four fragments need to be considered in order to fully understand the complexity of $\mathcal{L}$-MC($\mathbb{A}$). The known results are summarized in Figure 2.1.

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

Figure 2.1: Known complexity results for $\mathcal{L}$-MC($\mathbb{A}$).

The Promise CSP is a recently introduced extension of the CSP framework motivated by open problems in (in)approximability of satisfiability and coloring problems [AGH17, BG18, BBKO21]. 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 $\mathbb{B}$.

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. This work concentrates on $\{\exists, \wedge, \vee\}$-PMC and $\{\exists, \forall, \wedge, \vee\}$-PMC.

2.1.1 Contributions

Theorem 42 and Theorem 48 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 [MM18]; however, we believe that our approach is somewhat more transparent. In particular, it allows us to easily characterize meaningful templates for these problems (Propositions 41 and 47).

For $\{\exists, \wedge, \vee\}$-PMC, we obtain an L/NP-complete dichotomy in Theorem 44. 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 [GL18].

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 $\mathcal{L} \supsetneq \{\exists, \forall, \wedge, \vee\}$. We also give some examples where our efforts 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 Figure 2.2, the conditions for $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ are stated in terms of special surjective multi-homomorphisms of the template, introduced in Subsection 2.5.3.

<table>
<tr><th>$\mathcal{L}$-PMC($\mathbb{A}, \mathbb{B}$)</th><th>Condition</th><th>Complexity</th></tr>
<tr><td>$\{\exists, \forall, \wedge\}$-PMC($\mathbb{A}, \mathbb{B}$)</td><td></td><td>L or NP-complete</td></tr>
<tr><td rowspan="5">$\{\exists, \forall, \wedge, \vee\}$-PMC($\mathbb{A}, \mathbb{B}$)</td><td>$\mathsf{A}\mathsf{E}$-smuhom, or
A-smuhom and E-smuhom
and $\mathbb{A}, \mathbb{B}$ digraphs</td><td>L</td></tr>
<tr><td>A-smuhom and E-smuhom</td><td>NP $\cap$ coNP</td></tr>
<tr><td>A-smuhom, no E-smuhom</td><td>NP-complete</td></tr>
<tr><td>E-smuhom, no A-smuhom</td><td>coNP-complete</td></tr>
<tr><td>no A-smuhom
and no E-smuhom</td><td>NP-hard
and coNP-hard</td></tr>
<tr><td>$\{\exists, \forall, \wedge, \vee, =\}$-PMC($\mathbb{A}, \mathbb{B}$),
$\{\exists, \forall, \wedge, \vee, \neq\}$-PMC($\mathbb{A}, \mathbb{B}$),
$\{\exists, \forall, \wedge, \vee, \neg\}$-PMC($\mathbb{A}, \mathbb{B}$)</td><td></td><td>L or
PSPACE-complete</td></tr>
</table>

Figure 2.2: Complexity results for $\mathcal{L}$-PMC($\mathbb{A}, \mathbb{B}$).

2.2 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 $\overline{S} := A^n \setminus S$. 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, ..., i.e., $\mathbf{a} = (a_1, \dots, a_n)$.

A *signature* is a nonempty collection of relation symbols each with an associated arity, denoted $\operatorname{ar}(R)$ 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 $\emptyset \subsetneq R^{\mathbb{A}} \subsetneq A^{\operatorname{ar}(R)}$ 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 $\overline{\mathbb{A}}$. 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 $i \in [\operatorname{ar}(R)] = \{1, 2, \dots, \operatorname{ar}(R)\}$. 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 $g : A \to B$ 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 $\operatorname{MuHom}(\mathbb{A}, \mathbb{B})$

[1] We deviate here from the terminology of [MM11, MM12] because it would not work well in the promise setting.

and the set of all surjective multi-homomorphisms by $\mathrm{SMuHom}(\mathbb{A}, \mathbb{B})$.

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 $(v_1, \ldots, v_n)$ which contains every free variable of ψ and a tuple $(a_1, \ldots, a_n) \in A^n$, we write $\mathbb{A} \models \psi(a_1, \ldots, a_n)$ if ψ is satisfied when $v_1, \ldots, v_n$ are evaluated as $\varepsilon_A(v_1) = a_1, \ldots, \varepsilon_A(v_n) = a_n$, respectively. Notice that variables $v_1, \ldots, v_n$ indeed need to be pairwise distinct, otherwise this notation would not make sense. The tuple $(v_1, \ldots, v_n)$ is often specified by writing $\psi = \psi(v_1, \ldots, v_n)$.

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

2.3 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.

2.3.1 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 $\mathcal{L}$-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 $\mathcal{L}$-MC($\mathbb{C}$) can be reduced in polynomial-time (even logarithmic space) to $\mathcal{L}$-MC($\mathbb{A}$). Indeed, the reduction amounts to replacing atomic formulas of the form $R(\mathbf{v})$ by their definitions.

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 $\mathcal{L} = \{\exists, \wedge, =\}$ [BKW17b], surjective polymorphisms for $\mathcal{L} = \{\exists, \forall, \wedge, =\}$ [Mar17], multi-endomorphisms for $\mathcal{L} = \{\exists, \wedge, \vee\}$, surjective multi-endomorphisms for $\mathcal{L} = \{\exists, \forall, \wedge, \vee\}$ [MM18]; see also [Bör08]). 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 $A = [k] := \{1, \ldots, k\}$ and consider the following formula.

$$\phi(x_1, \ldots, x_k) := \bigwedge_{R \in \sigma} \bigwedge_{\mathbf{r} \in R^{\mathbb{A}}} R(x_{r_1}, \ldots, x_{r_{\mathrm{ar}(R)}}) \tag{2.1}$$

It follows immediately from definitions that, for any structure $\mathbb{E}$ in the signature of $\mathbb{A}$, $\mathbb{E} \models \phi(e_1, \ldots, e_k)$ 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., $\psi(x_1, x_3, x_2) := (\exists x_4)(\exists x_5) \ldots (\exists x_k)\phi$ 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.

2.3.2 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 29. *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* $\mathcal{L}$-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 $\mathcal{L}$-PMC$(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}$-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 $\mathcal{L}$-PMC$(\mathbb{C}, \mathbb{D})$ to a `Yes`-instance ψ of $\mathcal{L}$-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 30. *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 39. *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 $\mathcal{L}$-*PMC*$(\mathbb{C}, \mathbb{D}) \leq \mathcal{L}$-*PMC*$(\mathbb{A}, \mathbb{B})$.*

Proof. The reduction is to replace each atomic $Q(\mathbf{v})$ by the corresponding formula ψ from Definition 30. 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 $\neg$). $\square$

2.3.3 Interesting fragments

We now explain why only four fragments of first-order logic need to be considered in order to fully understand the problems $\mathcal{L}$-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 $\mathcal{L} \subseteq \{\exists, \vee\}$, 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 $(\mathcal{L} \cup \{=\})$-PMC$(\mathbb{A}, \mathbb{B})$ is essentially the same as $\mathcal{L}$-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 $\overline{R}$ interpreted as $\overline{R}^{\mathbb{A}} = \overline{R^{\mathbb{A}}}$, $\overline{R}^{\mathbb{B}} = \overline{R^{\mathbb{B}}}$. 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 $\overline{\mathbb{E}} \models \phi'$ 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 $\mathcal{L}$-PMC$(\mathbb{A}, \mathbb{B})$ to a `No`-instance (resp., `Yes`-instance) of $\mathcal{L}'$-PMC$(\overline{\mathbb{B}}, \overline{\mathbb{A}})$, 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.

2.4 Existential positive fragment

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

2.4.1 Characterization of templates and p-$\{\exists, \wedge, \vee\}$-definability

We start by characterizing $\{\exists, \wedge, \vee\}$-PMC templates. One direction of the characterization follows from the discussion below (2.1), the other one from the following observation.

Lemma 40. *Let f be a multi-homomorphism from $\mathbb{A}$ to $\mathbb{B}$, let $\phi(x_1, \dots, x_n)$ be a quantifier-free $\{\exists, \wedge, \vee\}$-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 41. *A pair $(\mathbb{A}, \mathbb{B})$ of similar structures is an $\{\exists, \wedge, \vee\}$-*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 40, and it follows that $\mathbb{B} \models \phi$.

For the forward implication, observe that the sentence obtained from the formula (2.1) 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 $\{\exists, \wedge, \vee\}$ (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 42. *Let $(\mathbb{A}, \mathbb{B})$ and $(\mathbb{C}, \mathbb{D})$ be $\{\exists, \wedge, \vee\}$-*PMC *templates such that $A = C$ and $B = D$. Then $(\mathbb{C}, \mathbb{D})$ is p-$\{\exists, \wedge, \vee\}$-definable from $(\mathbb{A}, \mathbb{B})$ if and only if* $\mathrm{MuHom}(\mathbb{A}, \mathbb{B}) \subseteq \mathrm{MuHom}(\mathbb{C}, \mathbb{D})$. *Moreover, in such a case,* $\{\exists, \wedge, \vee\}$-PMC$(\mathbb{C}, \mathbb{D}) \leq \{\exists, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$.

Proof. It is enough to verify the equivalence, since then the second claim follows from Theorem 39. To prove the forward implication, assume that $(\mathbb{C}, \mathbb{D})$ is p-$\{\exists, \wedge, \vee\}$-definable from $(\mathbb{A}, \mathbb{B})$, let $f \in \mathrm{MuHom}(\mathbb{A}, \mathbb{B})$, 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 40 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 30, 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 40. Therefore, $\mathbb{B} \models \psi(\mathbf{b})$ and, finally, $\mathbf{b} \in Q^{\mathbb{D}}$, as required.

For the backward implication, assume that $\mathrm{MuHom}(\mathbb{A}, \mathbb{B}) \subseteq \mathrm{MuHom}(\mathbb{C}, \mathbb{D})$, 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

$$\phi(x_{1,1}, \dots, x_{1,n}, x_{2,1}, \dots, x_{2,n}, \dots, x_{k,1}, \dots, x_{k,n}) := \bigwedge_{R \in \sigma} \bigwedge_{\mathbf{r} \in R^{\mathbb{A}}} \bigwedge_{\mathbf{j} \in [n]^{\mathrm{ar}(R)}} R(x_{r_1, j_1}, \dots, x_{r_{\mathrm{ar}(R)}, j_{\mathrm{ar}(R)}}) \quad (2.2)$$

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 $f \in \mathrm{MuHom}(\mathbb{A}, \mathbb{E})$ of multiplicity at most n. This relation is clearly equal to the union of $f(\mathbf{a})$ over all $f \in \mathrm{MuHom}(\mathbb{A}, \mathbb{E})$. The sought after formula ψ is then the disjunction of $\tau_{\mathbf{a}}$ over all $\mathbf{a} \in Q^{\mathbb{C}}$: it defines in $\mathbb{A}$ a relation containing $Q^{\mathbb{C}}$ (because of the identity "multi"-homomorphism $\mathbb{A} \to \mathbb{A}$) and, in $\mathbb{B}$, a relation contained in $Q^{\mathbb{D}}$ (because every multi-homomorphism from $\mathbb{A}$ to $\mathbb{B}$ is a multi-homomorphism from $\mathbb{C}$ to $\mathbb{D}$, whence $f(\mathbf{a}) \subseteq Q^{\mathbb{D}}$ for any $\mathbf{a} \in Q^{\mathbb{C}}$ and any $f \in \mathrm{MuHom}(\mathbb{A}, \mathbb{B})$). $\square$

2.4.2 Complexity classification

Since $\{\exists, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ reduces to $\{\exists, \wedge, \vee\}$-MC$(\mathbb{A})$ (or $\{\exists, \wedge, \vee\}$-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 44 shows that $\{\exists, \wedge, \vee\}$-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 42) a much more involved hardness result than in the non-promise case: NP-hardness of c-coloring rainbow k-colorable $2k$-uniform hypergraphs from [GL18] (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.

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

In the statement of Theorem 43 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 43 (Corollary 1.2 in [GL18]). *For any A and B of size at least 2, the problem $\{\exists, \wedge\}$-PMC$((A; \mathrm{Rb}_A^{2|A|}), (B; \mathrm{NAE}_B^{2|A|}))$ is NP-complete.*

Given this hardness result, the complexity classification is a simple consequence of Theorem 42.

Theorem 44. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \wedge, \vee\}$-PMC *template. If there is a constant homomorphism from* $\mathbb{A}$ *to* $\mathbb{B}$*, then* $\{\exists, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is in* L *(in fact, decidable in constant time), otherwise* $\{\exists, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is* NP*-complete.*

Proof. If there exists a constant homomorphism $f : \mathbb{A} \to \mathbb{B}$, say with image $\{b\}$, then all the relations $R^{\mathbb{B}}$ in $\mathbb{B}$ contain the constant tuple $(b, b, \ldots, b)$. 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 $\mathbb{A} \to \mathbb{B}$, 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 $(A; \mathrm{Rb}_A^{2|A|})$ to $(B; \mathrm{NAE}_B^{2|A|})$. The reduction from Theorem 42 and the hardness from Theorem 43 conclude the proof. □

2.5 Positive fragment

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

2.5.1 Witnesses for quantified formulas

It will be convenient to work with $\{\exists, \forall, \wedge, \vee\}$-formulas of the special form

$$\phi(x_1, \ldots, x_n) = \forall y_1 \exists z_1 \forall y_2 \exists z_2 \ldots \forall y_m \exists z_m \; \phi'(\mathbf{x}, \mathbf{y}, \mathbf{z}), \tag{2.3}$$

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 $\alpha_1 : A \to A$, $\alpha_2 : A^2 \to A$, …, $\alpha_m : A^m \to A$ which give us evaluations of the existentially quantified variables given the value of the previous universally quantified variables, i.e., these functions satisfy $\mathbb{A} \models \phi'(\mathbf{a}, \mathbf{c}, \alpha_1(c_1), \alpha_2(c_1, c_2), \ldots, \alpha_m(c_1, \ldots, c_m))$ 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 40.

Lemma 45. *Let f be a surjective multi-homomorphism from $\mathbb{A}$ to $\mathbb{B}$, let $\phi(x_1, \ldots, x_n)$ be an $\{\exists, \forall, \wedge, \vee\}$-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 $\{\exists, \forall, \wedge, \vee\}$-sentence such that $\mathbb{A} \models \phi$, then $\mathbb{B} \models \phi$.

Proof. The claim holds for quantifier-free $\{\exists, \forall, \wedge, \vee\}$-formulas by Lemma 40.

Next, we assume that ϕ is of the form (2.3) and select witnesses α_1, …, α_m for $\mathbb{A} \models \phi(\mathbf{a})$. Let $g : B \to A$ 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, …, β_m such that $\beta_i(b_1, \ldots, b_i) \in f(\alpha_i(g(b_1), \ldots, g(b_i)))$ for every

$i \in [m]$, are witnesses for $\mathbb{B} \models \phi(\mathbf{b})$. Indeed, for all $\mathbf{d} \in B^m$, we have $\mathbb{A} \models \phi'(\mathbf{a}, g(\mathbf{d}), \alpha_1(g(d_1)), \ldots, \alpha_m(g(d_1), \ldots, g(d_m)))$, and also $\mathbf{b} \in f(\mathbf{a})$, $\mathbf{d} \in f(g(\mathbf{d}))$, and $\beta_i(d_1, \ldots, d_i) \in f(\alpha_i(g(d_1), \ldots, g(d_i)))$ (by the assumption, choice of g, and choice of β_i, respectively); therefore, $\mathbb{B} \models \phi'(\mathbf{b}, \mathbf{d}, \beta_1(d_1), \ldots, \beta_m(d_1, \ldots, d_m))$ by the first paragraph. □

2.5.2 Characterization of templates and p-$\{\exists, \forall, \wedge, \vee\}$-definability

Unlike in the existential case, both characterizations require surjective and multi-valued functions. The core of these characterizations is an adjustment of (2.2) for surjective homomorphisms.

Lemma 46. *Let $\mathbb{A}$ be a structure with $A = [k]$ and m, n be arbitrary positive integers. Then there exists a formula $\phi(x_{1,1}, \ldots, x_{1,n}, x_{2,1}, \ldots, x_{2,n}, \ldots, x_{k,n})$ such that, for any structure $\mathbb{E}$ similar to $\mathbb{A}$ with $|E| \leq m$, we have $\mathbb{E} \models \phi(e_{1,1}, \ldots, e_{k,n})$ if and only if the mapping $i \mapsto \{e_{i,1}, \ldots, e_{i,n}\}$, $i \in [k]$ 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 $\phi_h(x_{1,1}, \ldots, x_{k,n}, z_1, \ldots, z_m)$ such that, for any structure $\mathbb{E}$ in the signature of $\mathbb{A}$, we have $\mathbb{E} \models \phi_h(e_{1,1}, \ldots, e_{k,n}, e'_1, \ldots, e'_m)$ if and only if the mapping $i \mapsto \{e_{i,1}, \ldots, e_{i,n}\} \cup \bigcup_{h(l)=i} e'_l$, $1 \leq i \leq k$, 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 (2.2).

We claim that the formula ϕ obtained by taking the disjunction of ϕ_h over all $h : [m] \to [k]$ and universally quantifying the variables $z_1, \ldots, z_m$ satisfies the requirement of the lemma, provided $|E| \leq m$. Indeed, on the one hand, if $\mathbb{E} \models \phi(e_{1,1}, \ldots, e_{k,n})$, 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 $\mathbb{E}$ (by the choice of ϕ_h). On the other hand, if $i \mapsto \{e_{i,1}, \ldots, e_{i,n}\}$ is contained in a surjective multi-homomorphism f, then for any evaluation $\varepsilon_E(z_1), \ldots, \varepsilon_E(z_m)$ 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. □

Proposition 47. *A pair $(\mathbb{A}, \mathbb{B})$ of similar structures is an $\{\exists, \forall, \wedge, \vee\}$-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 46 (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 45. □

An example which shows that one cannot replace in Proposition 47 "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 45 and 46.

Theorem 48. *Let* $(\mathbb{A}, \mathbb{B})$ *and* $(\mathbb{C}, \mathbb{D})$ *be* $\{\exists, \forall, \wedge, \vee\}$-PMC *templates such that* $A = C$ *and* $B = D$. *Then* $(\mathbb{C}, \mathbb{D})$ *is p-*$\{\exists, \forall, \wedge, \vee\}$*-definable from* $(\mathbb{A}, \mathbb{B})$ *if and only if* $\mathrm{SMuHom}(\mathbb{A}, \mathbb{B}) \subseteq \mathrm{SMuHom}(\mathbb{C}, \mathbb{D})$. *Moreover, in such a case,* $\{\exists, \forall, \wedge, \vee\}$-$\mathrm{PMC}(\mathbb{C}, \mathbb{D}) \leq \{\exists, \forall, \wedge, \vee\}$-$\mathrm{PMC}(\mathbb{A}, \mathbb{B})$.

Proof. It is enough to verify the equivalence, since then the second claim follows from Theorem 39.

To prove the forward implication, assume that $(\mathbb{C}, \mathbb{D})$ is p-$\{\exists, \forall, \wedge, \vee\}$-definable from $(\mathbb{A}, \mathbb{B})$, let $f \in \mathrm{SMuHom}(\mathbb{A}, \mathbb{B})$, 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 45 as follows. We have $\mathbb{A} \models \psi(\mathbf{a})$, where $\psi(\mathbf{x})$ is a formula from Definition 30. Then for any $\mathbf{b} \in f(\mathbf{a})$ we have $\mathbb{B} \models \psi(\mathbf{b})$ by Lemma 45, and, finally, $\mathbf{b} \in Q^{\mathbb{D}}$, as required.

For the backward implication, assume that $\mathrm{SMuHom}(\mathbb{A}, \mathbb{B}) \subseteq \mathrm{SMuHom}(\mathbb{C}, \mathbb{D})$, 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 $A = [k]$. Let $m = \max\{k, |B|\}$, and let $\phi(x_{1,1}, \dots, x_{1,n}, x_{2,1}, \dots, x_{2,n}, \dots, x_{k,1}, \dots, x_{k,n})$ be the formula 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 $\mathbb{E}$, provided by Lemma 46. For any $\mathbf{a} \in A^n$, we define $\tau_{\mathbf{a}}(x_1, \dots, x_n)$ to be the formula obtained from ϕ by renaming $x_{a_i,i}$ to x_i and existentially quantifying the remaining variables. The sought after formula ψ is then the disjunction of $\tau_{\mathbf{a}}$ over all $\mathbf{a} \in Q^{\mathbb{C}}$: it defines in $\mathbb{A}$ a relation containing $Q^{\mathbb{C}}$ (because of the surjective multi-homomorphism $\mathbb{A} \to \mathbb{A}$ defined by $a \mapsto \{a\}$) and, in $\mathbb{B}$, a relation contained in $Q^{\mathbb{D}}$ (because $\mathrm{SMuHom}(\mathbb{A}, \mathbb{B}) \subseteq \mathrm{SMuHom}(\mathbb{C}, \mathbb{D})$, so if a mapping is contained in a surjective multi-homomorphism from $\mathbb{A}$ to $\mathbb{B}$, then it is contained in a surjective multi-homomorphism from $\mathbb{C}$ to $\mathbb{D}$). $\square$

2.5.3 Membership

Clearly, every $\{\exists, \forall, \wedge, \vee\}$-MC, as well as $\{\exists, \forall, \wedge, \vee\}$-PMC, is in PSPACE. We now give a generalization of the remaining membership results from [MM11] 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 50) and later as well. We say that an $\{\exists, \forall, \wedge, \vee\}$-PMC template $(\mathbb{C}, \mathbb{D})$ is a *relaxation* of an $\{\exists, \forall, \wedge, \vee\}$-PMC template $(\mathbb{A}, \mathbb{B})$ if $(\mathbb{C}, \mathbb{A})$ and $(\mathbb{B}, \mathbb{D})$ are $\{\exists, \forall, \wedge, \vee\}$-PMC templates. Recall that, by Proposition 47, 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 49. *Let* $(\mathbb{A}, \mathbb{B})$ *and* $(\mathbb{C}, \mathbb{D})$ *be* $\{\exists, \forall, \wedge, \vee\}$-PMC *templates. If* $(\mathbb{C}, \mathbb{D})$ *is a relaxation of* $(\mathbb{A}, \mathbb{B})$*, then* $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{C}, \mathbb{D}) \leq \{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{A}, \mathbb{B})$.

Proof. The trivial reduction, which does not change the input, works. Indeed, `Yes`-instances of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{C}, \mathbb{D})$ are `Yes`-instances of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{A}, \mathbb{B})$ since $(\mathbb{C}, \mathbb{A})$ is an $\{\exists, \forall, \wedge, \vee\}$-PMC template, and `No`-instances of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{C}, \mathbb{D})$ are `No`-instances of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{A}, \mathbb{B})$ since $(\mathbb{B}, \mathbb{D})$ is an $\{\exists, \forall, \wedge, \vee\}$-PMC template. ☐

Theorem 50. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \forall, \wedge, \vee\}$-PMC *template. Then the following holds.*

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

Proof. For the first item, let f be an A-smuhom from $\mathbb{A}$ to $\mathbb{B}$ with $f(a^*) = B$, and consider an input ϕ in the special form (2.3), 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 $\mathbb{A} \vDash \phi'(a^*, a^*, \dots, a^*, \mathbf{a})$; this can be clearly decided in NP. It is clear that the answer is `Yes` whenever ϕ is a `Yes`-instance of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{A}, \mathbb{B})$. On the other hand, if $\mathbb{A} \vDash \phi'(a^*, \dots, a^*, \mathbf{a})$, then any functions $\beta_1 : B \to B$, $\dots$, $\beta_m : B^m \to B$ 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 $\mathbb{B} \vDash \phi$ by Lemma 40. 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 [MM11]. For the general case, we will construct $\mathbb{C}$ such that there is an 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 47, and then membership of $\{\exists, \forall, \wedge, \vee\}\text{-PMC}(\mathbb{A}, \mathbb{B})$ in L will follow from Proposition 49 and the mentioned Corollary 9 in [MM11]. Let f be an 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 $R^{\mathbb{C}} = \cup_{\mathbf{a} \in R^{\mathbb{A}}} f'(\mathbf{a})$ 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 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 AE-smuhom. Indeed, if $\mathbf{c} \in R^{\mathbb{C}}$, then $\mathbf{c} \in f'(\mathbf{a})$ for some $\mathbf{a} \in R^{\mathbb{A}}$. 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 $f'(\mathbf{a}) \subseteq R^{\mathbb{C}}$ as $f' \in \mathrm{SMuHom}(\mathbb{A}, \mathbb{C})$, and we are done. ☐

These membership results together with the (more involved) hardness results were sufficient for the tetrachotomy in [MM11]. One problem with generalizing this tetrachotomy is that, unlike in the non-promise setting, an $\{\exists, \forall, \wedge, \vee\}$-PMC template can admit an A-smuhom and an E-smuhom, but no AE-smuhom. However, such a situation cannot happen for digraphs.

Proposition 51. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \forall, \wedge, \vee\}$-PMC *template such that* $\mathbb{A}$ *and* $\mathbb{B}$ *are digraphs. If* $(\mathbb{A}, \mathbb{B})$ *admits an* A-smuhom *and an* E-smuhom*, then it admits an* AE-smuhom*.*

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

The proof of Proposition 51 is concluded. $\square$

2.5.4 Hardness

As a consequence of Theorems 43 and 48, we obtain the following hardness result.

Theorem 52. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \forall, \wedge, \vee\}$-PMC *template.*

1. *If there is no* E-smuhom *from* $\mathbb{A}$ *to* $\mathbb{B}$*, then* $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is* NP-*hard.*

2. *If there is no* A-smuhom *from* $\mathbb{A}$ *to* $\mathbb{B}$*, then* $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is* coNP-*hard.*

Proof. If there exists no E-smuhom from $\mathbb{A}$ to $\mathbb{B}$, then $\mathrm{SMuHom}(\mathbb{A}, \mathbb{B})$ is contained in $\mathrm{SMuHom}((A; \mathrm{Rb}_A^{2|A|}), (B; \mathrm{NAE}_B^{2|A|}))$. Theorem 43 and Theorem 48 then imply the first item. The second item follows by the duality argument. $\square$

In the non-promise setting, the absence of A-smuhoms and E-smuhoms is sufficient for PSPACE-hardness [MM11, MM18]. 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 $\{\exists, \forall, \wedge, \vee\}$.

Proposition 53. *$\{\exists, \forall, \wedge, \vee\}$-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 $\{\exists, \forall, \wedge, \vee\}$-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 49, 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 $\{\exists, \forall, \wedge, \vee\}$-MC$(\mathbb{B})$, a PSPACE-hard problem by, e.g., [Mar08]. Consider an input ϕ to $\{\exists, \forall, \wedge, \vee\}$-MC$(\mathbb{B})$ in the special form (2.3), 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 $\{\exists, \forall, \wedge, \vee\}$-MC$(\mathbb{B})$ are transformed to `Yes`-instances of $\{\exists, \forall, \wedge, \vee\}$-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 $A \to B$. We set

$$\psi = \forall x_1 \forall x_2 \ \exists x_3 \exists x_4 \dots \exists x_k \ \ (x_1 = x_2) \ \vee \bigwedge_{f : A \to B} \rho_f, \quad \text{where} \tag{2.4}$$

$$\rho_f = (\forall y_1' \exists z_1 \dots \forall y_m' \exists z_m) \ \ (\exists y_1 \dots \exists y_m) \ \left(\bigwedge_{i=1}^{m} \sigma[f, y_i', y_i] \right) \wedge \phi'(\mathbf{y}, \mathbf{z}) \tag{2.5}$$

$$\sigma[f, y_i', y_i] = \bigvee_{a \in A} \Big((y_i' = x_a) \wedge (y_i = x_{f(a)}) \Big) \tag{2.6}$$

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 $\beta_1, \dots, \beta_m$ 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 (2.4) 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 (2.6), 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 (2.5) 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 $\beta_1, \dots, \beta_m$ are witnesses for $\mathbb{B} \models \phi$, as required.

We now suppose that $\beta_1, \ldots, \beta_m$ are witnesses for $\mathbb{B} \models \phi$, and aim to show that $\mathbb{A} \models \psi$. Because of the first disjunct of (2.4), 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 $\{\exists, \forall, \wedge, \vee\}$-formulas (in the sense of Lemma 45), 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 $\varepsilon_A(x_a) = a$, $a = 3, 4, \ldots, k$. We take a function $f : A \to B$ and argue that ρ_f is satisfied in $\mathbb{A}$. Given a selection of $\varepsilon_A(y'_i)$, we evaluate z_i as $\varepsilon_A(z_i) = \beta_i(f(\varepsilon_A(y'_1)), \ldots, f(\varepsilon_A(y'_i)))$, 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 (2.6). The second conjunct in (2.5), $\phi'(\mathbf{y}, \mathbf{z})$, is also satisfied: we know $\mathbb{B} \models \phi'(\mathbf{c}, \beta_1(c_1), \ldots, \beta_m(c_1, \ldots, c_m))$ in particular for $c_1 = f(\varepsilon_A(y'_1))$, $\ldots$, $c_m = f(\varepsilon_A(y'_m))$ and, with this $\mathbf{c}$, it is apparent from the choice of evaluations that $\mathbb{B} \models \phi'(\mathbf{c}, \beta_1(c_1), \ldots, \beta_m(c_1, \ldots, c_m))$ is equivalent to $\mathbb{A} \models \phi'(\varepsilon_A(y_1), \ldots, \varepsilon_A(y_m), \varepsilon_A(z_1), \ldots, \varepsilon_A(z_m))$. The proof of $\mathbb{A} \models \psi$ is concluded. $\square$

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 54. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \forall, \wedge, \vee\}$-PMC *template which is closed under complementation. Then* $\{\exists, \forall, \wedge, \vee\}$-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 $i \in [\mathrm{ar}(R)]$, and every $\mathbf{c}, \mathbf{c}' \in A^{\mathrm{ar}(R)}$, if $c_i = a$, $c'_i = a'$, $c_j = c'_j$ for all $j \in [\mathrm{ar}(R)] \setminus \{i\}$, 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 context.

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 $\mathbf{c}, \mathbf{c}' \in A^{\mathrm{ar}(R)}$ 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 48, $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{C}, \mathbb{D}) \leq \{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$. Since there exists a surjective multi-valued function from A to B that preserves $\sim$ (namely, any $f \in \mathrm{SMuHom}(\mathbb{A}, \mathbb{B})$), 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 49, $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{E}, \mathbb{F}) \leq \{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{C}, \mathbb{D})$; therefore, $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{E}, \mathbb{F}) \leq \{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$. The former $\{\exists, \forall, \wedge, \vee\}$-PMC is PSPACE-hard by Proposition 53, so $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ is PSPACE-hard, too. $\square$

2.5.5 Summary and examples

The claims stated in Figure 2.2 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 [Lyn77].

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 55. *Let* $(\mathbb{A}, \mathbb{B})$ *be an* $\{\exists, \forall, \wedge, \vee\}$-PMC *template.*

1. *If* $\mathbb{B}$ *is Boolean, then* $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is in* L*, or is* NP*-complete, or* PSPACE*-complete.*

2. *If* $\mathbb{A}$ *is Boolean, then* $\{\exists, \forall, \wedge, \vee\}$-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* $\{\exists, \forall, \wedge, \vee\}$-PMC$(\mathbb{A}, \mathbb{B})$ *is in* L*, or is* PSPACE*-complete.*

Proof. If $\mathbb{B}$ is Boolean, then every E-smuhom (from $\mathbb{A}$ to $\mathbb{B}$) is an AE-smuhom. Moreover, if there is no 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 53 and Theorem 48. The other items are easy as well. $\square$

There are two wide gaps left for further investigation. First, it is unclear what the complexity is for the $\{\exists, \forall, \wedge, \vee\}$-PMC over templates that admit both an A-smuhom and an E-smuhom, but no 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).

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

The binary example above is pictured in Figure 2.3. The first relation is blue and the second black.

The second gap is between simultaneous NP- and coNP-hardness, and PSPACE-hardness, when the template admits neither an A-smuhom nor an

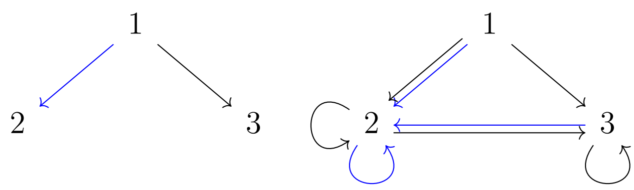

Figure 2.3: An example of a template having an A-smuhom and an E-smuhom, but no AE-smuhom.

E-smuhom. Examples with unknown complexity include the following.

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

The binary example above is pictured in Figure 2.4.

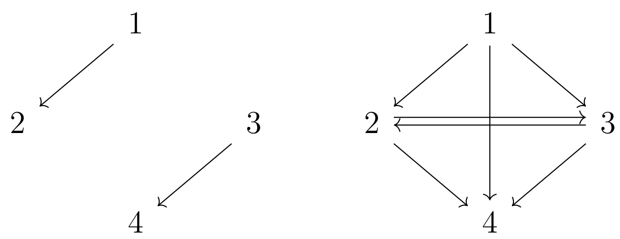

Figure 2.4: An example of a template that admits neither an A-smuhom nor an E-smuhom.

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.

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

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 [BKW17b, BBKO21]. It may be also helpful to characterize and study the sets of surjective multi-homomorphisms in the spirit of [Mar10, CM21].

3. PCSP seen from the other side

In the standard Promise CSP (PCSP), a pair of relational structures $(\mathbb{A}, \mathbb{B})$ (such that there is a homomorphism from $\mathbb{A}$ to $\mathbb{B}$) is fixed and PCSP$(\mathbb{A}, \mathbb{B})$ is defined as the problem of deciding whether an input structure admits a homomorphism to $\mathbb{A}$ or not even to $\mathbb{B}$. In this chapter we introduce a similar problem, where we restrict the left-hand side instead of the right-hand side, motivated by the so-called *left-hand side restricted CSP*, also called *CSP seen from the other side*. Namely, we fix a collection of pairs of relational structures Γ (such that for every pair there is a homomorphism from the first structure to the second one) and ask the following: for an input pair $(\mathbb{A}, \mathbb{B})$ from Γ and an input structure $\mathbb{X}$, decide whether there is a homomorphism from $\mathbb{B}$ to $\mathbb{X}$ or not even from $\mathbb{A}$ to $\mathbb{X}$.

The first two sections are devoted to the left-hand side resrticted CSP. The presentation largely follows Grohe's paper *The complexity of homomorphism and constraint satisfaction problems seen from the other side* [Gro07]. The last section is devoted to the left-hand side restricted PCSP.

3.1 Preliminaries

3.1.1 Relational structures and homomorphisms

A *signature* is a finite collection of relation symbols each with an associated *arity*, denoted $\text{ar}(R)$ for a relation symbol R. The *arity* of a signature is the maximum of the arities of all relations symbols it contains. A *relational structure* $\mathbb{A}$ in the signature σ, or a σ-*structure*, consists of a finite set A, called the *universe* of $\mathbb{A}$, and a relation $R^{\mathbb{A}} \subseteq A^{\text{ar}(R)}$ 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. We say that a class $\mathcal{C}$ of structures is of *bounded arity* if there is an r such that arity of the signature of every structure in $\mathcal{C}$ is at most r.

A structure over a signature containing a single binary relation symbol is called a *directed graph*, or *digraph*. If this relation is symmetric and loop free (i.e., it contains no pairs of the form (a, a)), we call the structure *undirected graph*. If the relation of a graph is the disequality relation on the universe, we call the graph a *complete graph* or *clique*.

A σ-structure $\mathbb{A}$ is a *substructure* of a σ-structure $\mathbb{B}$, denoted by $\mathbb{A} \subseteq \mathbb{B}$, if $A \subseteq B$ and $R^{\mathbb{A}} \subseteq R^{\mathbb{B}}$ for all $R \in \sigma$. A structure $\mathbb{A}$ is a *proper* substructure of $\mathbb{B}$, denoted by $\mathbb{A} \subset \mathbb{B}$, if $\mathbb{A} \subseteq \mathbb{B}$ and $\mathbb{A} \neq \mathbb{B}$.

We define the size of a σ-structure $\mathbb{A}$ to be

$$\|\mathbb{A}\| = |\sigma| + |A| + \Sigma_{R\in\sigma} |R^A| ar(R).$$

$\|\mathbb{A}\|$ is roughly the size of a reasonable encoding of $\mathbb{A}$. When taking structures A as inputs for algorithms, we measure the running time of the algorithm in terms of $\|\mathbb{A}\|$.

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. If there exists a homomorphism from $\mathbb{A}$ to $\mathbb{B}$, we write $\mathbb{A} \to \mathbb{B}$, and if there is none, we write $\mathbb{A} \not\to \mathbb{B}$. The composition of homomorphisms is a homomorphism.

Two structures $\mathbb{A}$ and $\mathbb{B}$ are *homomorphically equivalent* if $\mathbb{A} \to \mathbb{B}$ and $\mathbb{B} \to \mathbb{A}$.

A relational structure $\mathbb{A}$ is a *core* if there is no homomorphism from $\mathbb{A}$ to a proper substructure of $\mathbb{A}$. A core *of* a structure $\mathbb{A}$ is a substructure $\mathbb{A}'$ of $\mathbb{A}$ such that $\mathbb{A} \to \mathbb{A}'$ and $\mathbb{A}'$ is a core. Obviously, every core of a structure is homomorphically equivalent to the structure. It can be shown that all cores of a structure $\mathbb{A}$ are isomorphic. So, we often speak of *the* core of $\mathbb{A}$.

3.1.2 Homomorphism problem

General *homomorphism problem* (or *constraint satisfaction problem*) asks whether there is a homomorphism from one structure to another. We are interested in restrictions of this problem. For two classes $\mathcal{C}$ and $\mathcal{D}$ of structures, $\mathrm{Hom}(\mathcal{C}, \mathcal{D})$ is the following problem.

Input: Similar structures $\mathbb{A} \in \mathcal{C}$, $\mathbb{B} \in \mathcal{D}$;
Output: `Yes` if $\mathbb{A} \to \mathbb{B}$; `No` if $\mathbb{A} \not\to \mathbb{B}$.

If $\mathcal{C}$ is the class of all finite structures, we write $\mathrm{Hom}(-, \mathcal{D})$ instead of $\mathrm{Hom}(\mathcal{C}, \mathcal{D})$. The problem $\mathrm{Hom}(-, \{\mathbb{A}\})$ is also known as $\mathrm{CSP}(\mathbb{A})$.

Similarly, if $\mathcal{D}$ is the class of all finite structures, we write $\mathrm{Hom}(\mathcal{C}, -)$ instead of $\mathrm{Hom}(\mathcal{C}, \mathcal{D})$, and we call such a problem the *left-hand side restricted CSP*.If $\mathcal{C}$ is finite, then $\mathrm{Hom}(\mathcal{C}, -)$ is solvable in polynomial time, so usually we are interested in the case when $\mathcal{C}$ is an infinite collection of structures.

If $\mathcal{C}$ is the class of all cliques, the problem $\mathrm{Hom}(\mathcal{C}, -)$ is called the `Clique` *problem*. In [Kar72] Karp proved that `Clique` is NP-complete.

3.1.3 Graph minors and tree width

We will denote the vertex set of a graph $\mathbb{G}$ by G and its relation (or the set of edges) by $E^{\mathbb{G}}$. Since we are considering undirected graphs, we will view its edges as sets (unordered pairs) $e = \{v, w\}$, and we will use notations like $v \in e$ or $\{v, w\} \in E^{\mathbb{G}}$.

A graph $\mathbb{H}$ is a *minor* of a graph $\mathbb{G}$ if $\mathbb{H}$ is isomorphic to a graph that can be obtained from a subgraph of $\mathbb{G}$ by contracting edges. For example, in Figure 3.1, it is easy to see that $\mathbb{H}$ is a minor of $\mathbb{G}$.

A *minor map* from $\mathbb{H}$ to $\mathbb{G}$ is a mapping $\mu : H \to 2^G$ with the following properties.

- For all $v \in H$, the set $\mu(v)$ is nonempty and connected in $\mathbb{G}$.
- For all $v, w \in H$, with $v \neq w$, the sets $\mu(v)$ and $\mu(w)$ are disjoint.
- For all edges $\{v, w\} \in E^{\mathbb{H}}$, there are $v' \in \mu(v)$ and $w' \in \mu(w)$ such that $\{v', w'\} \in E^{\mathbb{G}}$.

For any two graphs $\mathbb{H}$ and $\mathbb{G}$, there is a minor map from $\mathbb{H}$ to $\mathbb{G}$ if and only if $\mathbb{H}$ is a minor of $\mathbb{G}$. Moreover, if $\mathbb{H}$ is a minor of a connected graph $\mathbb{G}$, then we

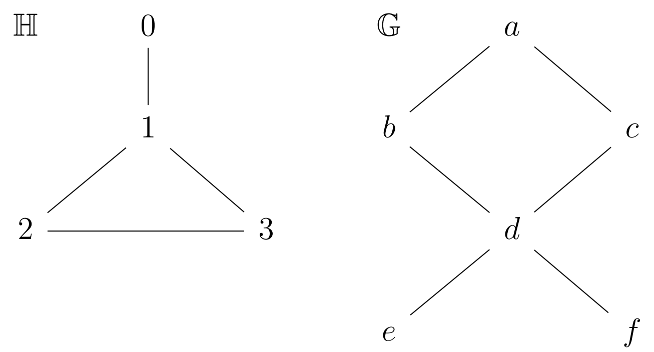

Figure 3.1: An example of a minor of a graph.

can always find a minor map from $\mathbb{H}$ onto $\mathbb{G}$, where by *onto* we mean

$$\bigcup_{v \in H} \mu(v) = G.$$

Going back to Figure 3.1, an example of a minor map from $\mathbb{H}$ to $\mathbb{G}$ is $\mu(0) = \{e\}$, $\mu(1) = \{d\}$, $\mu(2) = \{b\}$, $\mu(3) = \{a, c\}$.

Trees are connected acyclic graphs. A *tree-decomposition* of a graph $\mathbb{G}$ is a pair $(\mathbb{T}, \beta)$, where $\mathbb{T}$ is a tree and $\beta : T \to 2^G$ such that the following conditions are satisfied:

- For every $v \in G$ the set $\{t \in T | v \in \beta(t)\}$ is non-empty and connected in $\mathbb{T}$.
- For every $e \in E^{\mathbb{G}}$ there is a $t \in T$ such that $e \subseteq \beta(t)$.

For example, for every graph $\mathbb{G}$ there is a tree-decomposition where the tree is one vertex mapping to the whole set G.

The *width* of a tree-decomposition $(\mathbb{T}, \beta)$ is $\max\{|\beta(t)||t \in T\} - 1$, and the *tree width* of a graph $\mathbb{G}$, denoted by $tw(\mathbb{G})$, is the minimum w such that $\mathbb{G}$ has a tree-decomposition of width w.

For $k, l \geq 1$, the $(k \times l)$*-grid* is the graph with vertex set $[k] \times [l]$ and an edge between (i, j) and (i', j') if and only if $|i - i'| + |j - j'| = 1$. It can be shown that the $(k \times k)$-grid has tree width k. Figure 3.2 shows a tree-decomposition of width 3 for (3×3)-grid. In [RS86] Robertson and Seymour proved the following "converse", which is known as the Excluded Grid Theorem.

Theorem 56. *For every k there exists a $w(k)$ such that the $(k \times k)$-grid is a minor of every graph of tree width at least $w(k)$.*

We need to transfer some of the graph theoretic notions to arbitrary relational structures. The *Gaifman graph* (also known as *primal graph*) of a σ-structure $\mathbb{A}$ is the graph $\mathcal{G}(\mathbb{A})$ with vertex set A and an edge between vertices a and b if $a \neq b$ and there is a relation symbol $R \in \sigma$, say, of arity r, and a tuple $(a_1, a_2, \ldots, a_r) \in R^{\mathbb{A}}$ such that $a, b \in \{a_1, a_2, \ldots, a_r\}$. We can now transfer the notions of graph minor theory to relational structures. In particular, a subset $B \subseteq A$ is *connected* in a relational structure $\mathbb{A}$ if it is connected in $\mathcal{G}(\mathbb{A})$. A *minor map* from a relational structure $\mathbb{A}$ to a relational structure $\mathbb{B}$ is a mapping $\mu : A \to 2^B$ that is a minor

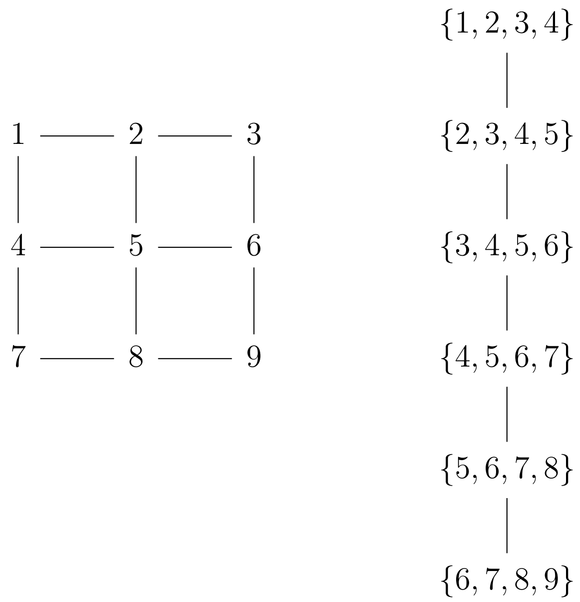

Figure 3.2: A tree-decomposition of a (3×3)-grid.

map from $\mathcal{G}(\mathbb{A})$ to $\mathcal{G}(\mathbb{B})$. A *tree decomposition* of a relational structure $\mathbb{A}$ can simply be defined to be a tree-decomposition of $\mathcal{G}(\mathbb{A})$.

A class $\mathcal{C}$ of structures has *bounded tree width* if there exists w such that $tw(\mathbb{A}) \leq w$ for all $\mathbb{A} \in \mathcal{C}$. A class $\mathcal{C}$ of structures has *bounded tree width modulo homomorphic equivalence* if there is w such that every $\mathbb{A} \in \mathcal{C}$ is homomorphically equivalent to a structure of tree width at most w.

3.1.4 Parameterized complexity theory

Parameterized complexity theory studies the complexity of decision problems with respect to both the size of the input and additional parameter.

Formally, a *parameterization* of a decision problem $P \subseteq \Sigma^*$, where Σ is an alphabet, is a polynomial time computable mapping $\kappa : \Sigma^* \to \mathbb{N}$, and a *parameterized problem* over Σ is a pair (P, κ) consisting of a problem $P \subseteq \Sigma^*$ and a parameterization κ of P. For example, the *parameterized clique problem* p-`Clique` is the problem (P, κ), where P is the set of all pairs $(\mathbb{G}, k)$ (suitably encoded over some finite alphabet) such that $\mathbb{G}$ contains a k-clique and the parameter κ is defined by $\kappa(\mathbb{G}, k) := k$. We present parameterized problems in the following form.

p-`Clique`:
Input: Graph $\mathbb{G}$, $k \in \mathbb{N}$;
Parameter: k;
Output: `Yes` if $\mathbb{G}$ has a clique of size k; `No` otherwise.

We will parameterize the homomorphism problem in the following way.

p-Hom$(\mathcal{C}, \mathcal{D})$:
Input: Similar structures $\mathbb{A} \in \mathcal{C}$, $\mathbb{B} \in \mathcal{D}$;
Parameter: $\|\mathbb{A}\|$;

Output: `Yes` if $\mathbb{A} \to \mathbb{B}$; `No` otherwise.

As before, if $\mathcal{D}$ is the set of all finite relational structures, we write p-Hom$(\mathcal{C}, -)$.

A parameterized problem (P, κ) over Σ is *fixed-parameter tractable* if there is a computable function $f : \mathbb{N} \to \mathbb{N}$ and an algorithm that decides if a given instance $x \in \Sigma^*$ belongs to the problem P in time

$$f(\kappa(x))|x|^{O(1)}.$$

The class of all fixed-parameter tractable parameterized problems is denoted by FPT.

While in the traditional theory of computational complexity we use polynomial-time reductions, in the theory of parameterized complexity we require an analogous notion of reduction that preserves fixed-parameter tractability, called an *fpt-reduction*.

An *fpt-reduction* from a parameterized problem (P, κ) over Σ to a parameterized problem (P', κ') over Σ' is a mapping $R : \Sigma^* \to (\Sigma')^*$ such that:

- For all $x \in \Sigma^*$ we have $x \in P$ if and only if $R(x) \in P'$.
- There is a computable function $f : \mathbb{N} \to \mathbb{N}$ and an algorithm that, given $x \in \Sigma^*$, computes $R(x)$ in time $f(\kappa(x))|x|^{O(1)}$.
- There is a computable function $g : \mathbb{N} \to \mathbb{N}$ such that for all instances $x \in \Sigma^*$ we have $\kappa'(R(x)) \leq g(\kappa(x))$.

.

A specific example of an fpt-reduction is the following reduction of p-`Clique` to p-Hom$(\mathcal{C}, -)$, where $\mathcal{C}$ is a class of structures that contains all cliques: an instance $(\mathbb{G}, k)$ of p-`Clique` is mapped to the instance $(\mathbb{K}_k, \mathbb{G})$ of p-Hom$(\mathcal{C}, -)$.

Hardness and *completeness* of parameterized problems for a parameterized complexity class are defined in the usual way: a parameterized problem is said to be hard for a parameterized complexity class if every problem in that class can be reduced to it using an fpt-reduction. A parameterized problem is said to be complete for a parameterized complexity class if it is hard for that class and it belongs to that class.

An analogue of NP in parameterized complexity theory is the class W[1]. It is a widely believed standard assumption that FPT $\neq$ W[1]. For the technical definitions of this and other parameterized complexity classes see [DF12] or [FG06].

It was shown that p-`Clique` is W[1]-complete [DF95]. The significance of this problem in the area of parameterized complexity is fundamental, as it has been used as a starting point for numerous reductions. We will see one of them in the next section.

3.2 Complexity of the left-hand side restricted CSP

In this section we present a dichotomy theorem [Gro07] for decidable classes of structures of bounded arity: for each such a class $\mathcal{C}$, p-Hom$(\mathcal{C}, -)$ is either in

FPT (even solvable in polynomial time) or W[1]-complete.

Marx's deep paper [Mar13] investigates the complexity for the case of unbounded arity. He also argues that investigating the fixed-parameter tractability of left-hand side restrictions is at least as interesting as investigating polynomial-time solvability. One reason is that FPT seems to be a more robust class in this context, e.g., these problems are unlikely to exhibit a standard, non-parameterized complexity dichotomy [Gro07].

The tractability part of the bounded arity dichotomy has been proved by Dalmau, Kolaitis, and Vardi in [DKV02]. We state a slightly stronger version as stated in [Gro07].

Theorem 57. *Let $\mathcal{C}$ be a class of structures of bounded tree width modulo homomorphic equivalence. Then* $\mathrm{Hom}(\mathcal{C}, -)$ *is in polynomial time.*

In [Gro07] Grohe proved the following hardness theorem.

Theorem 58. *Let $\mathcal{C}$ be a recursively enumerable class of structures of bounded arity that does not have bounded tree width modulo homomorphic equivalence. Then* $\text{p-Hom}(\mathcal{C}, -)$ *is* $\mathrm{W}[1]$*-hard under fpt-reductions.*

We will sketch the proof, but we need some preparation first.

Let $k \geq 2$ and $K = \binom{k}{2}$ and let $\mathbb{A}$ be a connected σ-structure with a $(k \times K)$-grid as a minor in its Gaifman graph. Since A is connected, there is a minor map from the $(k \times K)$-grid onto $\mathbb{A}$. We fix such a map μ. We also fix some bijection ρ between $[K]$ and the set of all unordered pairs of elements of $[k]$. For $p \in [K]$ we will write $i \in p$ instead of $i \in \rho(p)$. It will be convenient to switch between viewing the columns of the $(k \times K)$-grid as being indexed by elements of $[K]$ and unordered pairs of elements of $[k]$.

For a graph $\mathbb{G} = (G, E^{\mathbb{G}})$ we will define a σ-structure $\mathbb{M} = \mathbb{M}(\mathbb{A}, \mu, \mathbb{G})$ such that there exists a homomorphism from $\mathbb{A}$ to $\mathbb{M}$ if and only if $\mathbb{G}$ contains a k-clique. We define the universe of $\mathbb{M}$ to be

$$
\begin{aligned}
M = \{(v, e, i, p, a) | v \in G, e \in E^{\mathbb{G}},& \\
& i \in [k], p \in [K] \text{ such that } (v \in e \iff i \in p), \\
& a \in \mu(i, p)\}.
\end{aligned}
$$

We define the *projection* $\Pi : M \to A$ by letting

$$\Pi(v, e, i, p, a) = a$$

for all $(v, e, i, p, a) \in M$.

We shall define the relations of $\mathbb{M}$ in such a way that Π is a homomorphism from $\mathbb{M}$ to $\mathbb{A}$. For every $R \in \sigma$, say, of arity r, and for all tuples $\mathbf{a} = (a_1, a_2, \ldots, a_r) \in R^{\mathbb{A}}$ we add to $R^{\mathbb{M}}$ all tuples $\mathbf{m} = (m_1, m_2, \ldots, m_r) \in \Pi^{-1}(\mathbf{a})$ satisfying the following two constraints for all $m, m' \in \{m_1, m_2, \ldots, m_r\}$:

- If $m = (v, e, i, p, a)$ and $m' = (v', e', i, p', a')$, then $v = v'$.
- If $m = (v, e, i, p, a)$ and $m' = (v', e', i', p, a')$, then $e = e'$.

The following lemmas will guarantee correctness of the reduction in the proof of Theorem 58.

Lemma 59. *If $\mathbb{G}$ contains a k-clique, then $\mathbb{A} \to \mathbb{M}$.*

Lemma 60. *Suppose that $\mathbb{A}$ is a core. If $\mathbb{A} \to \mathbb{M}$, then $\mathbb{G}$ contains a k-clique.*

Sketch of the proof of Theorem 58. We will give an fpt-reduction from p-`Clique` to p-Hom$(\mathcal{C}, -)$. Let $\mathbb{G}$ be a graph and let $k \geq 1$. Let $K = \binom{k}{2}$. By the Theorem 56, there exists a structure $\mathbb{A} \in \mathcal{C}$ such that the Gaifman graph of the core of $\mathbb{A}$ has the $(k \times K)$-grid as a minor.

We find such an $\mathbb{A}$, compute the core $\mathbb{A}'$ of $\mathbb{A}$ and a minor map μ from the $(k \times K)$-grid to $\mathbb{A}'$. We let $\mathbb{A}''$ be the connected component of $\mathbb{A}'$ that contains the image of μ. $\mathbb{A}''$ is also a core. We can assume, without loss of generality, that μ is a minor map from the $(k \times K)$-grid onto $\mathbb{A}''$. We let $\mathbb{M}' = \mathbb{M}(\mathbb{A}'', \mu, \mathbb{G})$. By Lemma 59 and Lemma 60, $\mathbb{A}'' \to \mathbb{M}'$ if and only if $\mathbb{G}$ contains a k-clique. Let $\mathbb{M}$ be a disjoint union of $\mathbb{M}'$ and $\mathbb{A}' \backslash \mathbb{A}''$. Since $\mathbb{A}'$ is a core, every homomorphism from $\mathbb{A}'$ to $\mathbb{M}$ maps $\mathbb{A}''$ to $\mathbb{M}'$. So, $\mathbb{A}' \to \mathbb{M}$ if and only if $\mathbb{G}$ contains a k-clique. Since $\mathbb{A}'$ is the core of $\mathbb{A}$, it means that $\mathbb{A} \to \mathbb{M}$ if and only if $\mathbb{G}$ has a k-clique. $\square$

The main result of [Gro07] is the following theorem, which combines the previous two (Theorem 57 and Theorem 58). Note that for a decidable class $\mathcal{C}$, p-Hom$(\mathcal{C}, -)$ is in W[1], so W[1]-hardness becomes W[1]-completeness.

Theorem 61. *Assume that* FPT $\neq$ W[1]. *Then, for every recursively enumerable class $\mathcal{C}$ of structures of bounded arity the following statements are equivalent:*

- Hom$(\mathcal{C}, -)$ *is in polynomial time.*
- p-Hom$(\mathcal{C}, -)$ *is fixed-parameter tractable.*
- *$\mathcal{C}$ has bounded tree width modulo homomorphic equivalence.*

If either statement is false, then p-Hom$(\mathcal{C}, -)$ *is* W[1]*-hard.*

3.3 Left-hand side restricted PCSP

General *promise homomorphism problem* is the following parameterized problem:

Input: Similar structures $\mathbb{A}, \mathbb{B}, \mathbb{X}$ such that $\mathbb{A} \to \mathbb{B}$;
Parameter: $\|A\| + \|B\|$;
Output: `Yes` if $\mathbb{B} \to \mathbb{X}$; `No` if $\mathbb{A} \not\to \mathbb{X}$.

It is a *promise problem* in that the sets of `Yes` and `No` instances do not cover the set of all inputs and there are no requirements on algorithms if the input does not fall into `Yes` or `No` (or, put differently, the computer is promised that the input is `Yes` or `No` and the task is to decide which of the options take place)[1]. On the other hand, we do require that the set of `Yes` instances is disjoint from the set of `No` instances. For the general promise homomorphism problem, this property is guaranteed by the requirement $\mathbb{A} \to \mathbb{B}$.

[1]In fact, the general homomorphism problem and its restrictions should be also regarded as promise problems – we are promised that the input is in the expected form; see [Gro07] for a discussion about this issue.

The definition of an fpt-reduction naturally extends to promise problems: Instead of the first condition, we require that `Yes`-instances are mapped to `Yes`-instances (*completeness*) and that `No`-instances are mapped to `No`-instances (*soundness*). Since the definition of `No`-instances often involves negation (e.g. in the general homomorphism problem), soundness is often shown by proving the contrapositive: if the image is not a `No`-instance, then neither is the original instance.

We define the left-hand side restricted PCSP as the general promise homomorphism problem restricted to a class of pair of structures $(\mathbb{A}, \mathbb{B})$. We will be only concerned with recursively enumerable classes of bounded arity and so we include this requirement.

Definition 31. *A collection of pairs of similar structures* $(\mathbb{A}, \mathbb{B})$ *such that* $\mathbb{A} \to \mathbb{B}$ *is called a* template *if it is recursively enumerable and of bounded arity.*

For a template Γ*, the* left-hand side restricted PCSP over Γ*, denoted* $\mathrm{PHom}(\Gamma, -)$*, is the following problem.*

Input: Similar structures $\mathbb{A}, \mathbb{B}, \mathbb{X}$ *where* $(\mathbb{A}, \mathbb{B}) \in \Gamma$*;*
Parameter: $\|A\| + \|B\|$*;*
Output: `Yes` *if* $\mathbb{B} \to \mathbb{X}$*;* `No` *if* $\mathbb{A} \not\to \mathbb{X}$*.*

It is clear that the left-hand side restricted PCSP is a generalization of the (bounded arity, recursively enumerable) left-hand side restricted CSP: p-$\mathrm{Hom}(\mathcal{C}, -)$ is equivalent to $\mathrm{PHom}(\Gamma, -)$ for $\Gamma = \{(\mathbb{A}, \mathbb{A}) \mid \mathbb{A} \in \mathcal{C}\}$. We do not distinguish between $\mathcal{C}$ and the template Γ in this situation and call this template a *CSP template*.

Important examples of left-hand side restricted PCSP are approximation versions of the `Clique` problems. For a given computable function $f : \mathbb{N} \to \mathbb{N}$ with $f(n) < n$, $n \in \mathbb{N}$, the f-`Gap-Clique` problem is: given a graph $\mathbb{G}$ and $k \in \mathbb{N}$ decide whether $\mathbb{G}$ has a k-clique or not even an $f(k)$-clique. Clearly, f-`Gap-Clique` is equivalent to the $\mathrm{PHom}(\Gamma, -)$ for $\Gamma = \{(\mathbb{K}_{f(k)}, \mathbb{K}_k) \mid k \in \mathbb{N}\}$. We discuss this class of problems in the last subsection.

3.3.1 Homomorphic relaxations

A simple, but important reduction for the right-hand side restricted PCSP is by means of homomorphic relaxation (see [BBKO21]). A natural left-hand side version of this concept is as follows.

Definition 32. *Let* Γ *and* Δ *be templates. We say that* Γ *is a* (left-hand side) homomorphic relaxation *of* Δ *if for every* $(\mathbb{A}, \mathbb{B}) \in \Gamma$ *there exists* $(\mathbb{C}, \mathbb{D}) \in \Delta$ *such that all four structures are similar and* $\mathbb{A} \to \mathbb{C}$ *and* $\mathbb{D} \to \mathbb{B}$*.*

Proposition 62. *Let* Γ *and* Δ *be templates. If* Γ *is a homomorphic relaxation of* Δ*, then* $\mathrm{PHom}(\Gamma)$ *is fpt-reducible to* $\mathrm{PHom}(\Delta)$*.*

Proof. We map an instance $\mathbb{A}, \mathbb{B}, \mathbb{X}$ of $\mathrm{PHom}(\Gamma, -)$ to the instance $\mathbb{C}, \mathbb{D}, \mathbb{X}$ of $\mathrm{PHom}(\Delta, -)$, where $\mathbb{C}, \mathbb{D}$ are chosen so that $\mathbb{A} \to \mathbb{C}$ and $\mathbb{D} \to \mathbb{B}$ (in order to algorithmically find such a $\mathbb{C}, \mathbb{D}$ we simply enumerate the recursively enumerable class Δ.)

If $\mathbb{A}, \mathbb{B}, \mathbb{X}$ is a `Yes` instance of $\mathrm{PHom}(\Gamma)$, then $\mathbb{B} \to \mathbb{X}$. Therefore, $\mathbb{D} \to \mathbb{X}$ as $\mathbb{D} \to \mathbb{B}$ and homomorphisms compose, so $\mathbb{C}, \mathbb{D}, \mathbb{X}$ is a `Yes` instance of $\mathrm{PHom}(\Delta)$. This shows completeness of the reduction. Similarly, if $\mathbb{C}, \mathbb{D}, \mathbb{X}$ is not a `No` instance of $\mathrm{PHom}(\Delta)$, then $\mathbb{C} \to \mathbb{X}$, so $\mathbb{A} \to \mathbb{X}$ (as $\mathbb{A} \to \mathbb{C}$), thus $\mathbb{A}, \mathbb{B}, \mathbb{X}$ is not a `No` instance of $\mathrm{PHom}(\Gamma)$, showing soundness. The remaining requirements on fpt-reduction are clear. $\square$

By Theorem 57, $\mathrm{Hom}(\mathcal{C}, -)$ is in polynomial time whenever $\mathcal{C}$ is of bounded tree width. We immediately obtain the following corollary.

Corollary 63. *If a template Γ is a homomorphic relaxation of a CSP template of bounded tree width, then $\mathrm{PHom}(\Gamma, -)$ is fixed-parameter tractable.*

Note that a CSP template Γ is a homomorphic relaxation of a CSP template of bounded tree width if and only if Γ has bounded tree width modulo homomorphic equivalence. The tractability condition from Corollary 63 is by Theorem 58 the only source of fixed-point tractability (assuming $\mathrm{FPT} \neq \mathrm{W}[1]$) for CSP templates. We do not have a good reason to believe that this result generalizes to general templates, but neither do we have a counter-example, and the following question thus arises.

Question 64. *Let Γ be a template which is not a homomorphic relaxation of a CSP template of bounded tree width. Is then $\mathrm{PHom}(\Gamma, -)$ necessarily W[1]-hard?*

3.3.2 Sufficient condition for hardness

In this subsection we improve the sufficient condition for W[1]-hardness from Theorem 58 and give some corollaries.

The construction in the proof is largely inspired by Grohe's construction presented in the last section, but there are several differences. First, we remove some unnecessary components, namely i, p in the definition of M. On the other hand, we allow constant number of components of type v, e to add more flexibility. Third, we formulate the hardness criterion so that it can be directly applied e.g. to all the left-hand side restricted CSPs, not just connected cores.

The fourth difference is in that we use $(k \times k)$-grids instead of $(k \times \binom{k}{2})$-grids, which makes the construction somewhat more natural. In fact, instead of minor maps from grid, we use a more general concept of grid-like mappings that we now introduce. We use the following convention. If ρ is a mapping from C to a product $D \times E$, we denote by $\rho^{\leftarrow}$ and $\rho^{\rightarrow}$ its left and right components, respectively. That is, $\rho^{\leftarrow} : C \to D$ and $\rho^{\rightarrow} : C \to E$ are such that $\rho(c) = (\rho^{\leftarrow}(c), \rho^{\rightarrow}(c))$.

Definition 33. *Let $\mathbb{C}$ be a structure. A mapping $\rho : C \to [k] \times [k]$ is called a* grid-like mapping *from $\mathbb{C}$ onto $[k] \times [k]$ if it is surjective and, for each $i \in [k]$, both $(\rho^{\leftarrow})^{-1}(\{i\})$ and $(\rho^{\rightarrow})^{-1}(\{i\})$ are connected subsets of the Gaifman graph of $\mathbb{C}$.*

Note that a minor map μ from a $(k \times k)$-grid onto a structure $\mathbb{C}$ gives rise to a grid-like mapping ρ from $\mathbb{C}$ onto $[k] \times [k]$ by defining $\rho(c)$ as the unique pair (i, j) such that $c \in \mu(i, j)$.

Theorem 65. *Let Γ be a template. Suppose there exists $L \in \mathbb{N}$ such that for every $k \in \mathbb{N}$ the following condition is satisfied:*

(*) *There exists* $(\mathbb{A}, \mathbb{B}) \in \Gamma$ *and mappings* $\rho_1, \rho_2, \ldots, \rho_L : B \to [k] \times [k]$ *such that for each homomorphism* $g : \mathbb{A} \to \mathbb{B}$ *there exists a structure* $\mathbb{C}$*, a homomorphism* $h : \mathbb{C} \to \mathbb{A}$*, and* $l \in [L]$ *such that* $\rho_l gh$ *is a grid-like mapping from* $\mathbb{C}$ *onto* $[k] \times [k]$*.*

Then $\mathrm{PHom}(\Gamma, -)$ *is* $W[1]$*-hard.*

Proof. Let L be as in the statement. We will give an fpt-reduction from p-`Clique` to $\mathrm{PHom}(\Gamma, -)$. Let $\mathbb{G}$ be a graph and let $k \geq 1$. Let σ-structures $\mathbb{A}$ and $\mathbb{B}$ and mappings ρ_i be as in (*). We map the instance $(\mathbb{G}, k)$ of p-`Clique` to the instance $(\mathbb{A}, \mathbb{B}, \mathbb{X})$, where $\mathbb{X}$ is the σ-structure constructed as follows.

We define the universe of $\mathbb{X}$ to be

$$\begin{aligned} X = \{(b, (u_i, v_i)_{i=1}^L) \mid b \in B, \text{ for all } i \in [L]\ (u_i, v_i) \in G \times G \text{ such that} \\ \rho_i^{\leftarrow}(b) = \rho_i^{\rightarrow}(b) \Rightarrow u_i = v_i \text{ and} \\ \rho_i^{\leftarrow}(b) \neq \rho_i^{\rightarrow}(b) \Rightarrow \{u_i, v_i\} \in E^{\mathbb{G}}\} \subseteq B \times (G \times G)^L. \end{aligned}$$

We define the projection $\Pi : X \to B$ by

$$\Pi(b, (u_i, v_i)_{i=1}^L) = b$$

for all $(b, (u_i, v_i)_{i=1}^L) \in X$. We define the relations of $\mathbb{X}$ in such a way that Π is a homomorphism from $\mathbb{X}$ to $\mathbb{B}$. For every symbol $R \in \sigma$, say, of arity r, and for all tuples $\mathbf{b} = (b_1, b_2, \ldots, b_r) \in R^{\mathbb{B}}$ we add to $R^{\mathbb{X}}$ all tuples $\mathbf{x} = (x_1, x_2, \ldots, x_r) \in \Pi^{-1}(\mathbf{b})$ satisfying the following two constraints for all $(b, (u_i, v_i)_{i=1}^L), (b', (u_i', v_i')_{i=1}^L) \in \{x_1, x_2, \ldots, x_r\}$ and all $i \in [L]$:

(Cl) If $\rho_i^{\leftarrow}(b) = \rho_i^{\leftarrow}(b')$, then $u_i = u_i'$.

(Cr) If $\rho_i^{\rightarrow}(b) = \rho_i^{\rightarrow}(b')$, then $v_i = v_i'$.

Note that Π is indeed a homomorphism from $\mathbb{X}$ to $\mathbb{B}$ (even without imposing the constraints (Cl) and (Cr)).

The completeness of the reduction is guaranteed by the following claim.

Claim. *If* $\mathbb{G}$ *contains a* k*-clique, then* $\mathbb{B} \to \mathbb{X}$*.*

Proof. Let $v\langle 1\rangle, v\langle 2\rangle, \ldots, v\langle k\rangle \in G$ be vertices of a k-clique in $\mathbb{G}$. We define $h : B \to X$ by

$$h(b) = (b, (v\langle \rho_i^{\leftarrow}(b)\rangle, v\langle \rho_i^{\rightarrow}(b)\rangle)_{i=1}^L).$$

We need to verify that $h(b)$ is indeed in X, i.e., that $\rho_i^{\leftarrow}(b) = \rho_i^{\rightarrow}(b)$ implies $v\langle \rho_i^{\leftarrow}(b)\rangle = v\langle \rho_i^{\rightarrow}(b)\rangle$, and that $\rho_i^{\leftarrow}(b) \neq \rho_i^{\rightarrow}(b)$ implies $\{v\langle \rho_i^{\leftarrow}(b)\rangle, v\langle \rho_i^{\rightarrow}(b)\rangle\} \in E^{\mathbb{G}}$. Both implications are immediate.

In order to check that h is a homomorphism, take arbitary $R \in \sigma$ and $\mathbf{b} = (b_1, b_2, \ldots, b_r) \in R^{\mathbb{B}}$. Clearly, $h(\mathbf{b}) \in \Pi^{-1}(\mathbf{b})$. Moreover, for any $j, j' \in [r]$, if $\rho_i^{\leftarrow}(b_j) = \rho_i^{\leftarrow}(b_{j'})$, then trivially $v\langle \rho_i^{\leftarrow}(b_j)\rangle = v\langle \rho_i^{\leftarrow}(b_{j'})\rangle$, so (Cl) is satisfied. Similarly, (Cr) is satisfied as well, therefore $h(\mathbf{b}) \in R^{\mathbb{X}}$. This finishes the proof of the claim.

In order to prove the soundness of the reduction, assume that there exists a homomorphism from $\mathbb{A}$ to $\mathbb{X}$, say $\alpha : \mathbb{A} \to \mathbb{X}$. We need to find a k-clique in $\mathbb{G}$. Let $g = \Pi\alpha$ and let $\mathbb{C}$, h, and $l \in [L]$ be as in (*), i.e., ξ defined by

$$\xi = \rho_l \Pi \alpha h$$

is a grid-like mapping from $\mathbb{C}$ to $[k] \times [k]$.

For each $c \in C$, the element $\alpha h(c)$ is of the form $\alpha h(c) = (b, (u_i, v_i)_{i=1}^L) \in X$. We set $u\langle c\rangle = u_l$ and $v\langle c\rangle = v_l$. The definition of X implies the following claim.

Claim. *If $\xi^{\leftarrow}(c) = \xi^{\rightarrow}(c)$, then $u\langle c\rangle = v\langle c\rangle$. If $\xi^{\leftarrow}(c) \neq \xi^{\rightarrow}(c)$, then $\{u\langle c\rangle, v\langle c\rangle\} \in E^{\mathbb{G}}$.*

Proof. It is enough to notice that $\xi^{\leftarrow}(c) = \rho_l^{\leftarrow}\Pi\alpha h(c) = \rho_l^{\leftarrow}(b)$ and $\xi^{\rightarrow}(c) = \rho_l^{\rightarrow}(b)$, so the conclusion indeed follows form the definition of X.

The next claim follows from the constraints (Cl) and (Cr).

Claim. *Suppose that $\{c, c'\}$ is an edge in the Gaifman graph of $\mathbb{C}$. If $\xi^{\leftarrow}(c) = \xi^{\leftarrow}(c')$, then $u\langle c\rangle = u\langle c'\rangle$. If $\xi^{\rightarrow}(c) = \xi^{\rightarrow}(c')$, then $v\langle c\rangle = v\langle c'\rangle$.*

Proof. Since $\{c, c'\}$ is an edge in the Gaifman graph of $\mathbb{C}$, there exists $R \in \sigma$ and a tuple of the form $(\dots, c, \dots, c', \dots)$ in $R^{\mathbb{C}}$. The (αh)-image of this tuple, which is of the form

$$(\dots, (b, (u_i, v_i)_{i=1}^L), \dots, (b', (u_i', v_i')_{i=1}^L), \dots)$$

is in $R^{\mathbb{X}}$ as αh is a homomorphism from $\mathbb{C}$ to $\mathbb{X}$.

If $\xi^{\leftarrow}(c) = \xi^{\leftarrow}(c')$, then $\rho_l^{\leftarrow}(b) = \rho_l^{\leftarrow}(b')$ (see the proof of the previous claim), and thus $u_l = u_l'$ by (Cl). But then $u\langle c\rangle = u\langle c'\rangle$ by the definition of $u\langle c\rangle$ and $u'\langle c\rangle$.

The second part is proved analogously using (Cr) instead of (Cl) and the proof of the claim is concluded.

The proof of soundness is now concluded as follows. Since ξ is grid-like, we know that both $(\xi^{\leftarrow})^{-1}(\{i\})$ and $(\xi^{\rightarrow})^{-1}(\{i\})$ are connected subsets of the Gaifman graph of $\mathbb{C}$ for each $i \in [k]$. It follows from the last claim that $u\langle -\rangle$ is constant on $(\xi^{\leftarrow})^{-1}(\{i\})$ and $v\langle -\rangle$ is constant on $(\xi^{\rightarrow})^{-1}(\{i\})$. In other words, there exist $U\langle 1\rangle$, $U\langle 2\rangle$, …, $U\langle k\rangle \in G$ and $V\langle 1\rangle$, $V\langle 2\rangle$, …, $V\langle k\rangle \in G$ such that

$$u\langle c\rangle = U\langle \xi^{\leftarrow}(c)\rangle, \quad v\langle c\rangle = V\langle \xi^{\rightarrow}(c)\rangle.$$

Since ξ is onto $[k] \times [k]$, the preceding claim then gives us $U\langle i\rangle = V\langle i\rangle$ for each $i \in [k]$ and $\{U\langle i\rangle, V\langle j\rangle\} \in E^{\mathbb{G}}$ for each $i, j \in [k]$ with $i \neq j$, therefore $V\langle 1\rangle, \dots, V\langle n\rangle$ is a k-clique in $\mathbb{G}$.

Finally, notice that $|X|$ is at most $|B| \cdot |G|^{2L}$, which is polynomial in $|G|$, and it then easily follows that the reduction is an fpt-reduction. $\square$

Our first corollary says that Theorem 65 indeed covers Grohe's hardness result, Theorem 58.

Corollary 66. *Let $\mathcal{C}$ be a recursively enumerable class of structures of bounded arity that does not have bounded tree width modulo homomorphic equivalence. Then $\Gamma = \{(\mathbb{A}, \mathbb{A}) \mid \mathbb{A} \in \mathcal{C}\}$ satisfies the assumptions of Theorem 65 with $L = 1$.*

Proof. Let $k \in \mathbb{N}$. By Theorem 56, there exists $(\mathbb{A}, \mathbb{A}) \in \Gamma$ such that the Gaifman graph of the core of $\mathbb{A}$ has the $(k \times k)$-grid as a minor. Let $\mathbb{A}'$ be the core of $\mathbb{A}$, α a homomorphism from $\mathbb{A}$ to $\mathbb{A}'$, β a homomorphism from $\mathbb{A}'$ to $\mathbb{A}$, and μ a minor map from the $(k \times k)$-grid to $\mathbb{A}'$. Let $\mathbb{A}''$ be the connected component of $\mathbb{A}'$ that contains the image of μ and assume, without loss of generality, that μ is onto $\mathbb{A}''$. Let ν be the induced grid-like mapping from $\mathbb{A}''$ onto $[k] \times [k]$, that is, $\nu(a)$ is the unique pair (i, j) such that $a \in \mu(i, j)$. We extend ν arbitrarily to A' and define $\rho_1 = \nu\alpha$.

Let g be a homomorphism from $\mathbb{A}$ to $\mathbb{A}$. Since $\mathbb{A}'$ is a core, the homomorphism $\alpha g \beta$ has a right inverse $\gamma : \mathbb{A}' \to \mathbb{A}'$, i.e., $\alpha g \beta \gamma$ is the identity on A'. We define $\mathbb{C} = \mathbb{A}''$ and $h = \beta\gamma\iota$, where $\iota : \mathbb{A}'' \to \mathbb{A}'$ is the inclusion map. Now $\rho_1 g h = \nu\alpha g \beta\gamma\iota = \nu\iota$, which is grid-like by the choice of ν, and condition (*) is verified. □

The next corollary shows that Theorem 65 goes beyond the left-hand side restricted CSPs. The assumptions could be made slightly weaker, but they are, in any case, rather restrictive.

Corollary 67. *Let Γ be a template and $L \in \mathbb{N}$ such that*

- $\{\mathbb{A} : (\mathbb{A}, \mathbb{B}) \in \Gamma,\ \mathbb{A}$ *is connected*$\}$ *does not have bounded tree with modulo homomorphic equivalence and*
- *for every $(\mathbb{A}, \mathbb{B})$ there exist injective homomorphisms $g_1, g_2, \dots, g_L : \mathbb{A} \to \mathbb{B}$ such that, for every homomorphism $g : \mathbb{A} \to \mathbb{B}$, we have $gh = g_l$ for some homomorphism $h : \mathbb{A} \to \mathbb{A}$ and some $l \in [L]$.*

Then Γ satisfies the assumptions of Theorem 65.

Proof. For each k we use Theorem 56 to find $(\mathbb{A}, \mathbb{B}) \in \Gamma$ such that $\mathbb{A}$ is connected and has a $(k \times k)$-grid as a minor. This gives us a grid-like mapping ν from $\mathbb{A}$ onto the $(k \times k)$-grid. For each $l \in [L]$ we take any ρ_l such that $\rho_l g_l = \nu$, which is possible as g_l is injective. For each homomorphism $g : \mathbb{A} \to \mathbb{B}$, we have $gh = g_l$ for some h and l. Since $\rho_l g h = \rho_l g_l = \nu$, the condition (*) is satisfied with $\mathbb{C} = \mathbb{A}$. □

The assumptions of the last corollary in particular require "small number" of homomorphisms from $\mathbb{A}$ to $\mathbb{B}$. Our final observation is that Theorem 65 can be sometimes applied when the "number" of homomorphisms is not bounded by a constant.

Example. In this example it will be convenient to shift the vertex set of a $(k \times k)$-grid to $\{0, 1, \dots, k-1\} \times \{0, 1, \dots, k-1\}$ (and change the definition of grid-like mapping accordingly).

Let σ be a signature containing a binary symbol R and let $\Gamma = \{(\mathbb{A}_k, \mathbb{B}_k) | k \in \mathbb{N}\}$, where A_k with $R^{\mathbb{A}_k}$ is a $(k \times k)$-grid (in particular, $A_k = \{0, 1, \dots, k-1\}^2$), B_k with $R^{\mathbb{B}_k}$ is an $(f(k) \times f(k))$-grid for some $f(k) \geq k$, and every homomorphism from $\mathbb{A}_k$ to $\mathbb{B}_k$ is injective.

We show that the assumptions of Theorem 65 are satisfied with $L = 1$. For each k we take $(\mathbb{A}, \mathbb{B}) = (\mathbb{A}_k, \mathbb{B}_k)$ and set $\rho_1(i, j) = (i \mod k, j \mod k)$ where $(i, j) \in B_k$. Because of R, every (injective) homomorphism $g : \mathbb{A} \to \mathbb{B}$ is of the form $g(i, j) = (l + si, l' + s'j)$ for some $l, l' \in \{0, 1, \dots\}$ and $s, s' \in \{-1, 1\}$, or of the form $g(i, j) = (l + sj, l' + s'i)$. In all the cases, $\rho_1 g$ is a grid-like mapping from $\mathbb{A}$ onto $\{0, 1, \dots, k-1\} \times \{0, 1, \dots, k-1\}$, so (*) is satisfied with $\mathbb{C} = \mathbb{A}$ and the identical h.

3.3.3 Approximating clique

In this final subsection we briefly discuss the f-`Gap-Clique` problems. Recall that $f : \mathbb{N} \to \mathbb{N}$ is a function such that $f(n) \leq n$ for each $n \in \mathbb{N}$ and that f-`Gap-Clique` is equivalent to $\mathrm{PHom}(\{(\mathbb{K}_{f(k)}, \mathbb{K}_k) \mid k \in \mathbb{N}\})$. In this subsection, we implicitly assume that all functions and sets are computable.

For the identity function f, f-`Gap-Clique` is p-`Clique`, it is therefore a $W[1]$-hard problem. A well known open question is how small can f be made.

Question 68. *For what functions f is f-`Gap-Clique` W[1]-hard? Is it W[1]-hard for any unbounded function f?*

Note that f-`Gap-Clique` can be used, instead of `Clique`, as a starting point in the proof of Theorem 65. It follows from the proof that, whenever f-`Gap-Clique` is W[1]-hard, (*) can be weakened to "$\dots \rho_l gh$ is a grid-like mapping from $\mathbb{C}$ onto $K \times K$ with $K \geq f(k)$" (where the definition of a grid-like mapping is naturally extended).

A recent breakthrough toward answering Question 68 is the following result of Lin [Lin21].

Theorem 69. *For any $0 < c \leq 1$, the problem f-`Gap-Clique` is W[1]-hard whenever $f(n) \geq cn$ for all $n \in \mathbb{N}$.*

The result was further improved in [SK22].

Another natural question is what happens if we consider templates $\{(\mathbb{K}_{f(k)}, \mathbb{K}_k)\}$ where k runs through some infinite set instead of the whole $\mathbb{N}$. Here is a simple observation in this direction.

Proposition 70. *Let $f, g : \mathbb{N} \to \mathbb{N}$ be functions such that $f(n), g(n) < n$ for each $n \in \mathbb{N}$, and let $L \subseteq \mathbb{N}$. Suppose that for every $k \in \mathbb{N}$ there exists $l \in L$ such that $g(l) \geq (l/k + 1)f(k)$. Then f-`Gap-Clique` is fpt-reducible to $\mathrm{PHom}(\{(\mathbb{K}_{g(l)}, \mathbb{K}_l) \mid l \in L\})$.*

Proof. Given an instance $(\mathbb{K}_{f(k)}, \mathbb{K}_k, \mathbb{G})$ of f-`Gap-Clique` we find $l \in L$ such that $g(l) \geq (l/k+1)f(k)$ and take the smallest integer m such that $mk \geq l$. Note that $m \leq l/k + 1$. We map the given instance to the instance $(\mathbb{K}_{g(l)}, \mathbb{K}_l, \mathbb{H})$ where $\mathbb{H}$ is obtained by taking m disjoint copies of $\mathbb{G}$ and making all pairs of vertices in different copies adjacent.

If $\mathbb{G}$ contains a k-clique, then $\mathbb{H}$ contains an mk-clique for we can take the same clique in every copy of $\mathbb{G}$ in $\mathbb{H}$ to get such a clique. Since $l \leq mk$, the graph $\mathbb{H}$ contains an l-clique. This proves the completeness of the reduction.

Assume now that $\mathbb{H}$ contains a $g(l)$-clique. By taking the largest intersection of that clique with a copy of $\mathbb{G}$ in $\mathbb{H}$, we obtain a clique in $\mathbb{G}$ of size at least $g(l)/m$. Since $g(l)/m \geq g(l)/(l/k + 1) \geq f(k)$, the graph $\mathbb{G}$ contains an $f(k)$-clique, proving soundness of the reduction. $\square$

The proposition allows us to slightly refine Theorem 69.

Corollary 71. *For any $0 < c \leq 1$ and any infinite $L \subseteq \mathbb{N}$, the problem* $\mathrm{PHom}(\{(\mathbb{K}_{g(l)}, \mathbb{K}_l) \mid l \in L\})$ *is W[1]-hard whenever $g(l) \geq cl$ for all $l \in L$.*

Proof. We define $f(n) = (c/2)n$ and for each k we take any $l \in L$ with $l \geq k$. Since $g(l) \geq cl \geq c(l/k + 1)k/2 = (l/k + 1)f(k)$, Proposition 70 gives us a reduction from f-`Gap-Clique`, which is W[1]-hard by Theorem 69. □

3.4 Conclusion

We introduced the framework of left-hand side restricted PCSPs, which simultaneously generalizes left-hand side restricted CSPs and approximation versions of the k-clique problem, and we provided some initial results. The main technical contribution is the sufficient condition for W[1]-hardness in Theorem 65 which, in particular, covers left-hand side restricted bounded arity CSPs. However, it remains to be seen whether this general framework for left-hand side restriction can be as fruitful as it is for the right-hand side restrictions (see [BBKO21]). A challenging problem in this direction is to improve the sufficient condition so that it not only covers CSPs but also the constant factor approximation of k-clique stated in Theorem 69. Such a result seems to require a significantly different construction.

Bibliography

[AB21] Kristina Asimi and Libor Barto. Finitely Tractable Promise Constraint Satisfaction Problems. In Filippo Bonchi and Simon J. Puglisi, editors, *46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021)*, volume 202 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 11:1–11:16, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[ABB22] Kristina Asimi, Libor Barto, and Silvia Butti. Fixed-Template Promise Model Checking Problems. In Christine Solnon, editor, *28th International Conference on Principles and Practice of Constraint Programming (CP 2022)*, volume 235 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 2:1–2:17, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[AGH17] Per Austrin, Venkatesan Guruswami, and Johan Håstad. $(2+\epsilon)$-Sat is NP-hard. *SIAM J. Comput.*, 46(5):1554–1573, 2017.

[Bar19] L. Barto. Promises make finite (constraint satisfaction) problems infinitary. In *2019 34th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–8, 2019.

[BBKO18] Libor Barto, Jakub Bulín, Andrei A. Krokhin, and Jakub Oprsal. Algebraic approach to promise constraint satisfaction. *CoRR*, abs/1811.00970, 2018.

[BBKO21] 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.

[BG17] Joshua Brakensiek and Venkatesan Guruswami. Promise constraint satisfaction: Algebraic structure and a symmetric boolean dichotomy. *CoRR*, abs/1704.01937, 2017.

[BG18] 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.

[BJK05] Andrei Bulatov, Peter Jeavons, and Andrei Krokhin. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, March 2005.

[BK12] Libor Barto and Marcin Kozik. Absorbing subalgebras, cyclic terms, and the constraint satisfaction problem. *Log. Methods Comput. Sci.*, 8(1:07):1–26, 2012. Special issue: Selected papers of the Conference "Logic in Computer Science (LICS) 2010".

[BKO19] Jakub Bulín, Andrei Krokhin, and Jakub Opršal. Algebraic approach to promise constraint satisfaction. In *Proceedings of the 51st Annual ACM SIGACT Symposium on the Theory of Computing (STOC '19)*, New York, NY, USA, 2019. ACM.

[BKW17a] 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.

[BKW17b] 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.

[Bod08] Manuel Bodirsky. Constraint satisfaction problems with infinite templates. In Nadia Creignou, Phokion G. Kolaitis, and Heribert Vollmer, editors, *Complexity of Constraints*, volume 5250 of *Lecture Notes in Computer Science*, pages 196–228. Springer, 2008.

[BOP18] Libor Barto, Jakub Opršal, and Michael Pinsker. The wonderland of reflections. *Israel Journal of Mathematics*, 223(1):363–398, Feb 2018.

[Bör08] 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.

[Bul17] Andrei 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.

[Che12] 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.

[CM21] Catarina Carvalho and Barnaby Martin. The lattice and semigroup structure of multipermutations. *International Journal of Algebra and Computation*, 0(0):1–25, 2021.

[DF95] Rod G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness ii: On completeness for w[1]. *Theoretical Computer Science*, 141(1):109–131, 1995.

[DF12] Rodney G. Downey and Michael Ralph Fellows. *Parameterized complexity*. Monographs in Computer Science. Springer New York, NY, 2012.

[DKV02] Víctor Dalmau, Phokion G. Kolaitis, and Moshe Y. Vardi. Constraint satisfaction, bounded treewidth, and finite-variable logics. In Pascal Van Hentenryck, editor, *Principles and Practice of Constraint Programming - CP 2002*, pages 310–326, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[DSM+20] Guofeng Deng, Ezzeddine El Sai, Trevor Manders, Peter Mayr, Poramate Nakkirt, and Athena Sparks. Sandwiches for promise constraint satisfaction, 2020.

[FG06] Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Texts in Theoretical Computer Science. An EATCS Series. Springer Berlin, Heidelberg, 2006.

[FKOS19] Miron Ficak, Marcin Kozik, Miroslav Olsák, and Szymon Stankiewicz. Dichotomy for Symmetric Boolean PCSPs. In Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi, editors, *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 57:1–57:12, Dagstuhl, Germany, 2019. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[FV98] Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1):57–104, February 1998.

[GL18] Venkatesan Guruswami and Euiwoong Lee. Strong inapproximability results on balanced rainbow-colorable hypergraphs. *Comb.*, 38(3):547–599, 2018.

[Gro07] Martin Grohe. The complexity of homomorphism and constraint satisfaction problems seen from the other side. *J. ACM*, 54(1), mar 2007.

[HN90] Pavol Hell and Jaroslav Nešetřil. On the complexity of H-coloring. *J. Combin. Theory Ser. B*, 48(1):92–110, 1990.

[Jea98] Peter Jeavons. On the algebraic structure of combinatorial problems. *Theor. Comput. Sci.*, 200(1-2):185–204, 1998.

[Kar72] Richard M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972.

[KKR17] Vladimir Kolmogorov, Andrei Krokhin, and Michal Rolínek. The complexity of general-valued CSPs. *SIAM Journal on Computing*, 46(3):1087–1110, 2017.

[Lin21] Bingkai Lin. Constant approximating k-clique is w[1]-hard. *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, 2021.

[Lyn77] Nancy Lynch. Log space recognition and translation of parenthesis languages. *J. ACM*, 24(4):583–590, oct 1977.

[Mar08] 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.

[Mar10] 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.

[Mar13] Dániel Marx. Tractable hypergraph properties for constraint satisfaction and conjunctive queries. *J. ACM*, 60(6), nov 2013.

[Mar17] 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.

[MM10] 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.

[MM11] 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.

[MM12] Florent Madelaine and Barnaby Martin. The complexity of positive first-order logic without equality. *ACM Trans. Comput. Logic*, 13(1), January 2012.

[MM18] Florent R. Madelaine and Barnaby Martin. On the complexity of the model checking problem. *SIAM J. Comput.*, 47(3):769–797, 2018.

[RS86] Neil Robertson and P.D Seymour. Graph minors. v. excluding a planar graph. *Journal of Combinatorial Theory, Series B*, 41(1):92–114, 1986.

[Sch78] Thomas J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, pages 216–226, New York, NY, USA, 1978. ACM.

[SK22] Karthik C. S. and Subhash Khot. Almost Polynomial Factor Inapproximability for Parameterized k-Clique. In Shachar Lovett, editor, *37th Computational Complexity Conference (CCC 2022)*, volume 234 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 6:1–6:21, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[Zhu17] Dmitriy Zhuk. A proof of CSP dichotomy conjecture. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 331–342, Oct 2017.

[Zhu20] Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM*, 67(5), August 2020.

[ZM20] 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, *Proccedings 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.

List of Figures

List of publications

- Kristina Asimi and Libor Barto. *Finitely Tractable Promise Constraint Satisfaction Problems.* In Filippo Bonchi and Simon J. Puglisi, editors, 46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021), volume 202 of Leibniz International Proceedings in Informatics (LIPIcs), pages 11:1–11:16, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik

- Kristina Asimi, Libor Barto, and Silvia Butti. *Fixed-Template Promise Model Checking Problems.* In Christine Solnon, editor, 28th International Conference on Principles and Practice of Constraint Programming (CP 2022), volume 235 of Leibniz International Proceedings in Informatics (LIPIcs), pages 2:1–2:17, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik