Classification of the existence of gadget reductions between some Promise CSP's

Adam Steif

2024

Abstract

A fundamental goal of computer science is to understand the complexity of computational problems. One class of problems are promise constraint satisfaction problems (PCSP's). We study PCSP's related to graph coloring, linearly ordered coloring (LO-coloring) and rainbow coloring. We first dive into graph coloring PCSP's and classify the non-existence of certain gadget reductions between such problems. Next we systematically classify the existence gadget reductions between PCSP's related to graph coloring and LO-coloring and between PCSP's related to graph coloring and rainbow coloring that could yield new complexity results. For the most part these gadget reductions are shown to be non-existent. We do however prove the existence of a reduction from PCSP($\mathbf{K}_l, \mathbf{K}_k$) to PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) for $k \geq l \geq 3$ and $r \geq 4$ yielding a substantial number of new NP-hardness results for PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$).

Acknowledgements

I would like to take this opportunity to say a big thank you to my advisor Per Austrin for his support and guidance throughout this project providing important ideas and insights. I would also like to say thanks to my examiner Johan Håstad for providing many useful comments that helped elevate the quality of the report.

Contents

1 Introduction

One of the fundamental goals of computer science is to classify the complexity of computational problems. That is determine whether specific computational problems are polynomial time solvable, NP-hard or of another complexity. One class of computational problems are constraint satisfaction problems (CSP's), in short the aim of a CSP is to given a set of elements with relations between the elements, can we assign a value to each element in a way that satisfies certain constraints. This class consists of a large variety of problems with a rich mathematical structure providing a good place to experiment with complexity classifications and algorithmic techniques.

Problems that can be formulated as CSP's have been around for a long time, a prime example is graph coloring. The more formal framework however originated apporximately 40 years ago independently in three different fields, computer science, artificial intelligence and database theory. It showed up in different ways in the three fields and it was not until the late 1990s that it was realised that all three fields were in fact studying the same concept. As mentioned above these problems are extensively studied in theoretical computer science. In AI and applied computer science they can be used to solve various real-world problems such as scheduling. [17]

An extensively studied class of CSP's are the ones with a restricted set of types of constraints, or a fixed constraint language. The CSP's we study only have one type of constraints. Graph coloring is again a prime example where the only constraint is that two adjacent edges cannot receive the same color.

An important question regarding CSP's with a fixed constraint language was whether all CSP's with a fixed constraint language are either in polynomial time solvable or NP-complete. This was conjectured by Feder and Vardi [12], the conjecture had support in previous works such as Schaefer [20] who showed that the conjecture holds for Boolean CSP's. This conjecture distinguishes CSP's from the more general class of problems NP since assuming P≠NP there are infinitely many so called NP-intermediate problems that are not in P but neither NP-complete [18]. This conjecture, known as the Dichotomy Conjecture, lead to a great deal of research in the area for about two decades which resulted in two independent proofs by Bulatov [11] and by Zhuk [22, 23]. The algebraic approach to the CSP can be used as a tool to determine the complexity of CSP's and was in particular an important tool for both proofs. This uses so called polymorphisms that are multivariate mappings between relations on sets where relations are preserved under the mapping. This approach started with the importance of polymorphisms being established in a couple of papers by Jeavons et al. [16] and Jeavons [15]. Next this was brought to a more abstract level by Bulatov et al. [9, 10] where the CSP's could be translated into an algebraic problem. This gave a more powerful approach to the study of CSP's.

The problems studied in this thesis are an extended version of the CSP, namely the promise constraint satisfaction problem. This is denoted by PCSP. The aim of these problems is to find an approximately good solution to a problem given that a good solution is known to exist. By an approximate solution we mean a solution in which all constraints are satisfied, but where each constraint is relaxed. An example of this is given a 3-colorable graph, find a valid 8-coloring [1].

We study the decision variant of this problem which asks whether a good solution exists or if not even an approximately good solution exists for a given input. Open problems of this form have existed and been studied before the notion of PCSP was introduced. The PCSP was introduced quite recently [2, 7, 8] formalizing a new set of computational problems containing these open inapproximability problems together with infinitely more problems yet to be studied. Note that PCSP's contain all CSP's as these problems simply ask whether an input has a good solution or does not even have a good solution, i.e we can let the approximate solution be the good solution.

One of the open problems contained in PCSP's that is studied in this thesis is determining the complexity of the problem of distinguishing graphs that are k-colorable or not even q-colorable for

[1] Note that we are not referring to the most common notion of an approximately good solution which is a solution in which almost all constraints are satisfied. An example of this is a 3-coloring of a graph where almost no two adjacent vertices are colored the same.

$q \geq k$; PCSP($\mathbf{K}_k, \mathbf{K}_q$). Two substantial results for the NP-hardness of these results were found recently; distinguishing k-colorable from not even $(2k-1)$-colorable graphs is NP-hard for $k \geq 3$ [5]. Also, for $k \geq 4$ distingushing k-colorable graphs from not even $(\binom{k}{\lfloor k/2 \rfloor} - 1)$-colorable graphs is NP-hard[21].

We also look at PCSP's related to Linearly ordered coloring (LO-coloring) and rainbow coloring of hypergraphs. LO-coloring is a natural variant of normal hypergraph colorings studied by Barto, Baristelli and Berg [3], they studied 3-uniform hypergraphs where they called an admissible k-coloring one where if two vertices of an edge have the same color then the remaining one is given a higher color, the colors are $[k] = \{1, 2, ..., k\}$. They posed the question if it is NP-hard to find an LO-k-coloring of a 3-uniform hypergraph with the promise that the input graph is LO 2-colorable. This is still an open problem. Nakajima and Živný [19] studied more general cases of LO-coloring where not only LO-colorings of 3-uniform, but r-uniform hypergraphs for $r \geq 4$ were studied. They did not answer the question posed by Barto et al. [3], but they did however prove various NP-hardness results for r-unifrom hypergraphs for $r \geq 5$.

A c-rainbow coloring of a k-uniform hypergraph H is a coloring of H such that all c colors are represented in each edge. PCSP's related to rainbow coloring have been studied in the last few years [1, 6, 13, 14] and still has open questions regarding its complexity. The specific PCSP related to rainbow coloring we study in this thesis is to distinguish k-rainbow colorable hypergraph from those that are not even 2-colorable. The definitions of these colorings are presented in Section 2.2. There exists specific complexity results for this PCSP [1, 13, 14] that that we present and use in this thesis.

The aim of this thesis is to present a more comprehensive classification for when so called gadget reductions exist or do not exist between PCSP's of the mentioned types of coloring. A more comprehensive classification of this is of interest as it could directly yield new complexity results. A problem PCSP($\mathbf{A}, \mathbf{B}$) can be concluded to be NP-hard if an NP-hard PCSP is found to have a reduction to PCSP($\mathbf{A}, \mathbf{B}$). The problem PCSP($\mathbf{A}, \mathbf{B}$) could also be found to be in P if it can be reduced to a PCSP in P. Furthermore it is of interest to find reductions between PCSP's of unknown complexity, because then a future complexity result of one PCSP could by this more comprehensive classification immediately yield information about the complexity of other problems.

2 Background

We closely follow the notation and definitions presented by Barto et al. [5]. We however talk about relations on sets instead of relational structures as the objects we study are relational structures containing only one relation. We use notation $[n] = \{1, 2, ..., n\}$.

Definition 2.1. *A* ***relation*** *on a set A is a tuple $\boldsymbol{A} = (A; R^A)$ where A is a set and each $R^A \subseteq A^{ar(R)}$ is a relation on A of well defined arity $ar(R^A) \geq 1$.*

Throughout this thesis we refer to tuples existing in a relation on a set $\mathbf{A} = (A; R^A)$, when this is done we are formally referring to tuples existing in R^A.

A simple example of a relation is on a digraph $\mathbf{D} = (V; E)$ where V is the set of vertices and E is a relation of arity 2. We have that $(x, y) \in E \subset V^2$ for whenever x has an edge to y. In this thesis we study undirected graphs, $\mathbf{G} = (V; E)$ with the requirement that if an edge exists in one direction it also must exist in the other, that is if $(x, y) \in E$ then $(y, x) \in E$. Examples of specific graphs is the complete graph $\mathbf{K}_k = ([k]; \{(x, y) \in [k]^2 | x \neq y\})$ and a 5-cycle

$$\mathbf{C}_5 = ([5]; \{(1,2), (2,1), (2,3), (3,2), (3,4), (4,3), (5,4), (4,5), (5,1), (1,5)\}).$$

Other important relations in this thesis are on k-uniform hypergraphs. These are generalised graphs where instead of each edge consisting of two vertices they consist of k vertices for some $k \geq 2$. Note that a normal graph is a 2-uniform hypergraph. An example of this is $\mathbf{H}_{3,3} = ([3]; [3]^3 \backslash \{(1,1,1), (2,2,2), (3,3,3)\})$. Note that there exists plenty of edges containing a vertex more then once, two examples of this are the edges $(1, 1, 2)$ and $(3, 2, 3)$.

Definition 2.2. *Two relations on sets* $\boldsymbol{A} = (A; R^A)$ *and* $\boldsymbol{B} = (B; R^B)$ *are said to be* ***similar*** *if* $ar(R^A) = ar(R^B)$.

Note that this means that all graphs are similar.

Definition 2.3. *For two similar relations* $\boldsymbol{A}$ *and* $\boldsymbol{B}$ *a* ***homomorphism*** *from* $\boldsymbol{A}$ *to* $\boldsymbol{B}$ *is a map* $H : A \to B$ *such that* $(a_1, ..., a_{ar(R^A)}) \in R^A \implies (h(a_1), ..., h(a_{ar(R^B))}) \in R^B$

We write $\mathbf{A} \to \mathbf{B}$ to denote that a homomorphism exists from $\mathbf{A}$ to $\mathbf{B}$.

2.1 CSP's and PCSP's

Now we are ready to introduce a formal definition of a CSP problem.

Definition 2.4. *For a fixed relation* $\boldsymbol{B}$*,* ***CSP(B)*** *is the problem of deciding whether a given input relation* $\boldsymbol{I}$*, similar to* $\boldsymbol{B}$*, admits a homomorphism to* $\boldsymbol{B}$*. Here* $\boldsymbol{B}$ *is called the* ***template*** *for CSP(*$\boldsymbol{B}$*).*

An example of a CSP problem is 1-in-3-SAT, CSP($\mathbf{T}$) which uses the following relation on $\{0, 1\}$; $\mathbf{T} = (\{0, 1\}; \{(1, 0, 0), (0, 1, 0), (0, 0, 1)\})$. We can describe the problem from two equivalent perspectives given an input $\mathbf{I} = (\{x_1, ..., x_n\}; \{(x_{1,1}, x_{2,1}, x_{3,1}), (x_{1,2}, x_{2,2}, x_{3,2}), ..., (x_{1,m}, x_{2,m}, x_{3,m}) : x_{i,j} = x_l \text{ for some } l \in [n] \, \forall i, j \in [3] \times [m]\})$. The first perspective is just following the definition, does there exist a homomorphism $\mathbf{I} \to \mathbf{T}$? The second equivalent perspective is determining if we can assign values 0 or 1 to variables $x_1, ..., x_n$ so that for each of the clauses $(x_{1,1}, x_{2,1}, x_{3,1}), (x_{1,2}, x_{2,2}, x_{3,2}), ..., (x_{1,m}, x_{2,m}, x_{3,m}) : x_{i,j} = x_l$ for some $l \in [n] \, \forall i, j \in [3] \times [m]$ one element is given the value 1 and the other two are given the value 0. We think of 1 as True and of 0 as False. In this thesis we generally stick to the first perspective, especially in various proofs since the theorems we introduce use the first perspective. In the background we continue using both perspectives in order to better relate to the various examples that are presented.

Now let us introduce the main topic of this thesis; promise constraint satisfaction problems which we denote by PCSP. We begin with a formal definition.

Definition 2.5. *A* ***PCSP template*** *is a pair of similar relations* $\boldsymbol{A}$ *and* $\boldsymbol{B}$ *such that* $\boldsymbol{A} \to \boldsymbol{B}$*. The problem* ***PCSP(A, B)*** *is, given an input* $\boldsymbol{I}$ *similar to* $\boldsymbol{A}$ *and* $\boldsymbol{B}$*, output YES if* $\boldsymbol{I} \to A$ *and NO if* $\boldsymbol{I} \not\to \boldsymbol{B}$

The question the PCSP problem asks is whether an input $\mathbf{I}$ satisfies $\mathbf{A}$ or does not even satisfy $\mathbf{B}$. The promise in these problems is that it is never the case that $\mathbf{I} \to \mathbf{B}$ but $\mathbf{I} \not\to \mathbf{A}$. This would be a problem since neither YES nor NO could be the output of such an input. Also $\mathbf{A} \to \mathbf{B}$ makes sure that that for no input, the output could be both YES and NO. Because then we could have the case that $\mathbf{I} \to \mathbf{A}$ but $\mathbf{I} \not\to \mathbf{B}$. Note also that PCSP($\mathbf{A}, \mathbf{A}$) is the same problem as CSP($\mathbf{A}$).

The search version of the problem is given an input $\mathbf{I}$ such that $\mathbf{I} \to \mathbf{A}$, find a homomorphism $h : \mathbf{I} \to \mathbf{B}$. An example of this would be given that a graph G is 3-colorable, find a 6-coloring of G.

2.2 Relevant CSP's for this thesis

Here we introduce the CSP's that are relevant for this thesis.

2.2.1 Graph coloring

Recall that $\mathbf{K}_k = ([k]; \{(x, y) \in [k]^2 : x \neq y\})$ is the complete graph of k vertices.

Definition 2.6. *A graph* $\boldsymbol{G} = ([l]; \{$ *some subset of* $[l]^2\})$ *is* k***-colorable*** *if and only if there exists a homomorphism* $\boldsymbol{G} \to \boldsymbol{K}_k$.

Note then that the problem CSP($\mathbf{K}_k$) is, given an input $\mathbf{G}$, determine if the graph $\mathbf{G}$ is k-colorable. Note that this coincides with the regular definition of k-coloring. The homorphism existing simply means that we can assign one of k colors to each vertex in $\mathbf{G}$ such that the relation is preserved, that is if two vertices share an edge in $\mathbf{G}$ there images must share an edge in $\mathbf{K}_k$ meaning that they receive different colors. Let us study a graph

$$\mathbf{G} = (\{A, B, C, D\}; \{(A, B), (A, C), (B, A), (B, C), (B, D), (C, A), (C, B), (D, B)\})$$

And see how we could find a valid 3-coloring, or homorphism $\mathbf{G} \to \mathbf{K}_3$.

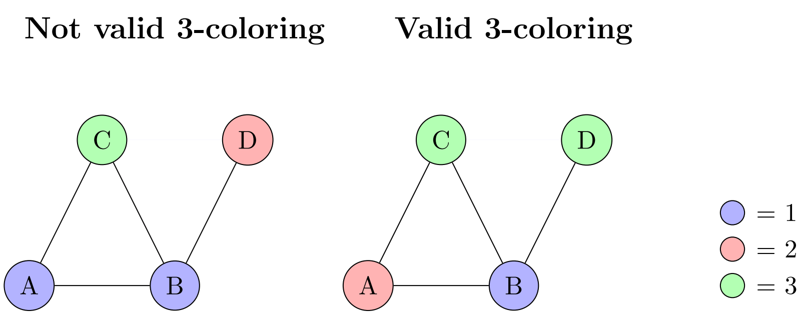

Figure 1: The left coloring is not valid since the edge (A, B) is monochromatic.

In Figure 1 have that a valid homomorphism $\mathbf{G} \to \mathbf{K}_3$ is sending $A \to 2$, $B \to 1$, $C \to 3$ and $D \to 3$.

2.2.2 Normal coloring of k-uniform hypergraph

We use the relation on a set $[c]$ $\mathbf{H}_{k,c} = ([c]; \{(a_1, ..., a_k) \in [c]^k : a_i \neq a_j \text{ for some } i, j \in [k]\})$. Note that we in this thesis study undirected hypergraphs. This means that if a hypergraph has the edge $(a_1, ..., a_k)$ then it also has the edge $(a_{\pi(1)}, ..., a_{\pi(k)})$ for all permutations $\pi : [k] \to [k]$. This is also the case for the templates for rainbow coloring and LO-coloring presented later.

Definition 2.7. *A k-uniform hypergraph $\boldsymbol{I}$ is c-**colorable** if and only if there exists a homomorphism $\boldsymbol{I} \to \boldsymbol{H}_{k,c}$*

Note then that the problem CSP($\mathbf{H}_{k,c}$) is, given an input $\mathbf{I}$, determine if the hypergraph $\mathbf{I}$ is c-colorable. This simply means that a k-uniform hypergraph $\mathbf{I}$ is c-colorable if and only if each vertex can be assigned a color such that no edge is monochromatic, i.e all vertices of the edge do not have the same color.

In Figure 2 we have that a valid homomorphism $\mathbf{I} \to \mathbf{H}_{4,2}$ is sending $A, B, C, D, E \to 1$ and $F \to 2$.

2.2.3 Rainbow coloring of k-unifrom hypergraph

Here we use a relation on the set $[c]$ $\mathbf{R}_{k,c} = ([c]; \{(a_1, a_2, ..., a_k) \in [c]^k : \forall a \in [c] \, a = a_i \text{ for some } i \in [k]$. This means that $(a_1, a_2, ..., a_k) \in [c]^k$ is a tuple in the relation if it contains all values of $[c]$.

Definition 2.8. *A k-uniform hypergraph $\boldsymbol{I}$ is c-**rainbow colorable** if and only if there exists a homomorphism $\boldsymbol{I} \to \boldsymbol{R}_{k,c}$.*

Note then that the problem CSP($\mathbf{R}_{k,c}$) is, given an input $\mathbf{I}$, determine if the hypergraph $\mathbf{I}$ is c-rainbow colorable. This simply means that a k-uniform hypergraph $\mathbf{I}$ is c-rainbow colorable if and only if each vertex can be assigned a color such that each edge contains all colors.

In Figure 3 we have that a valid homomorphism $\mathbf{I} \to \mathbf{R}_{4,3}$ is sending $D, F \to 1$, $A, E \to 2$ and $B, C \to 3$.

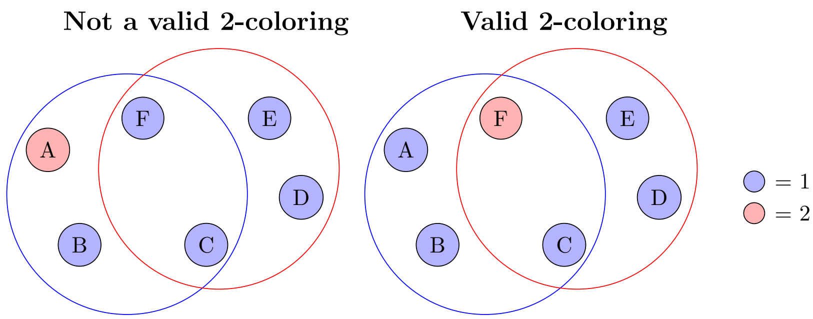

Figure 2: Two colorings of graph **I**. The left coloring is not valid since (C, D, E, F) forms a monochromatic edge

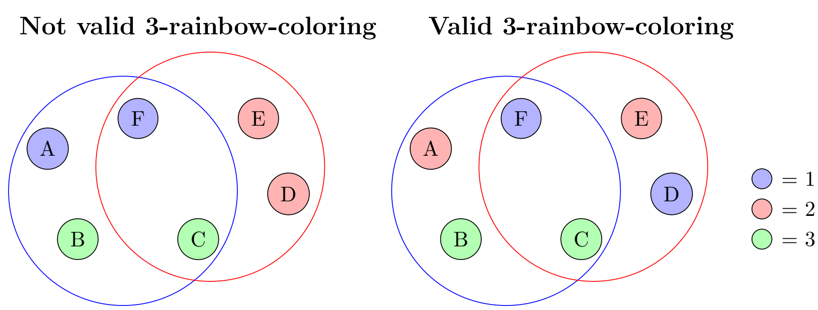

Figure 3: Two colorings of graph **I**. The left coloring is not a valid rainbow coloring as the edge (A,B,C,F) does not contain a vertex of the red color.

2.2.4 Linearly ordered coloring of r-uniform hypergraphs

For Linearly ordered coloring or LO coloring we use a relation on the set $[l]$ $\mathbf{LO}_l^r = ([l]; \{(a_1, a_2, ..., a_r) \in [l]^r : (a_1, a_2, ..., a_r)$ has unique maximum$\}$.

Definition 2.9. *A r-uniform hypergraph $\boldsymbol{I}$ is l-**LO colorable** if and only if there exists a homomorphism $\boldsymbol{I} \to \boldsymbol{LO}_l^r$.*

Note then that the problem CSP($\mathbf{LO}_l^r$) is, given an input **I**, determine if the hypergraph **I** is c-LO colorable. This simply means that a r-uniform hypergraph **I** is c-LO colorable if and only if each vertex can be assigned a color (in the form of a number) such that for each edge only one vertex is given the color/number of the highest value.

In Figure 4 we have that a valid homomorphism $\mathbf{I} \to \mathbf{LO}_2^4$ is sending $B, C, E, F \to 1$ and $A, D \to 2$.

2.3 The PCSP's we study

Now we present the PCSP's we study in this thesis related to each of the different colorings.

- Graph coloring: PCSP($\mathbf{K}_k, \mathbf{K}_q$) for $q \geq k \geq 3$. The task is to determine whether a graph is k-colorable or not even q-colorable.
- Rainbow coloring: PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) for $k > q \geq 2$. The task is to determine whether a k-uniform hypergraph is q-rainbow colorable or not even 2-colorable.

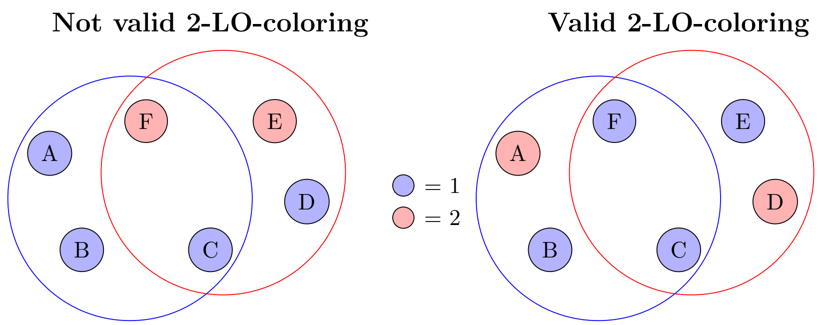

Figure 4: Two colorings of graph **I**. The left coloring is not valid as the for the edge (C, D, E, F) the maximum color is 2, but both F and E have this color so the maximum color is not unique and therefore not a valid LO-coloring.

- LO coloring: PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) for $k \geq l \geq 2$ and $r \geq 3$. The task is to determine whether an r-uniform hypergraph is l-LO colorable or not even k-LO colorable.

As graph coloring has the most applications and is the most actively researched this the main interest of the thesis. Therefore we want to figure out what gadget reductions exist within graph coloring PCSP's and gadget reductions that exist between graph coloring and the PCSP's corresponding to the other colorings. As this could potentially yield new NP-hardness results or give information about what potential future complexity results imply regarding the complexity of other colorings.

2.3.1 Gadget reductions

The reason for we want to find specifically gadget reductions is that they are a commonly used in theoretical computer science to prove complexity results. Furthermore there exists theorems stating when they exist or do not exist which we use in this thesis. Gadget reductions were used before term *gadget reduction* was even introduced. Let us give a definition.

Definition 2.10. *Let $\boldsymbol{I}_1 = (I_1; R_1)$ be an input to PCSP(A, B). A* ***gadget reduction*** *from PCSP(A, B) to PCSP(C, D) is an algorithm that in polynomial time transforms the $\boldsymbol{I}_1 = (I_1; R_1)$ into an input $\boldsymbol{I}_2 = (\boldsymbol{I}_2; R_2)$ of PCSP(C, D) such that each tuple in R_1 defines a new set of tuples which together form the tuples of R_2. It is required that if $\boldsymbol{I}_1$ is a YES-instance then so is $\boldsymbol{I}_2$, and if $\boldsymbol{I}_1$ is a NO-instance then so is $\boldsymbol{I}_2$.*

We can see gadget reductions as each constraint that needs to be satisfied being transformed into a set of constraints of another problem that must be satisfied. This set of constraints becomes the gadget that in a sense simulates our original constraint. The new problem consists of several gadgets that correspond to one of the constraints of the original problem. Gadget reductions play an important role in the algebraic theory of CSP's where there exists certain conditions for when they exist [4]. For this thesis however it is not necessary to dive deeper into this algebraic theory.

2.4 Known NP-hardness results and trivial reductions

We go through each of the PCSP's and present the known complexity results and also show trivial reductions that exist within each class of PCSP's. Some of which are immediate from the following observation.

Observation 2.11. *If for relations on sets A_1, A_2, B_1, B_2 the homomorphisms $\boldsymbol{A}_1 \to \boldsymbol{A}_2$ and $\boldsymbol{B}_2 \to \boldsymbol{B}_1$ exist, then PCSP$(\boldsymbol{A}_1, \boldsymbol{B}_1)$ can be gadget reduced to PCSP$(\boldsymbol{A}_2, \boldsymbol{B}_2)$ in polynomial time.*

Proof. The following is a valid gadget reduction.

```
function PCSP(A_1, B_1)(I):
    return PCSP(A_2, B_2)(I)
```

YES instances are mapped to YES because if $\mathbf{I} \to \mathbf{A}_1$ and $\mathbf{A}_1 \to \mathbf{A}_2$ then $\mathbf{I} \to \mathbf{A}_2$. Next NO instances are mapped to NO because if $\mathbf{I} \not\to \mathbf{B}_1$ and $\mathbf{B}_2 \to \mathbf{B}_1$ then $\mathbf{I} \not\to \mathbf{B}_2$. ☐

2.4.1 Graph coloring PCSP($\mathbf{K}_k, \mathbf{K}_q$)

We have that PCSP($\mathbf{K}_k, \mathbf{K}_q$) is NP-hard if $2k - 1 \geq q \geq k$ for $k \geq 3$ ([5]) and if $(\binom{k}{\lfloor k/2 \rfloor} - 1) \geq q \geq k$ for $k \geq 4$ ([21]). Furthermore we know that CSP($\mathbf{K}_2$), i.e 2-coloring, is in P which in turn implies that PCSP($\mathbf{K}_2, \mathbf{K}_q$) is in P for $q \geq 2$.

We state a few trivial reductions as theorems which we illustrate together with the known NP-hard results in Figure 5.

Theorem 2.12. *There exists a gadget reduction PCSP($\boldsymbol{K}_k, \boldsymbol{K}_q$) $\to$ PCSP($\boldsymbol{K}_k, \boldsymbol{K}_{q-1}$) for $q > k \geq 2$.*

Proof. Follows immediately from Observation 2.11 ☐

Theorem 2.13. *There exists a reduction PCSP($\boldsymbol{K}_k, \boldsymbol{K}_q$) $\to$ PCSP($\boldsymbol{K}_{k+1}, \boldsymbol{K}_{q+1}$) for $q \geq k \geq 2$.*

Proof. This can be shown with the following reduction:

```
function PCSP(K_k, K_q)(I):
    add a new node v to I and create an edge between v and all other vertices.
    return PCSP(K_{k+1}, K_{q+1})(I)
```

YES-instances are mapped to YES-instances as we can color the original vertices with k colors and we can color the new vertex v with $(k + 1)$. Furthermore NO-instances are mapped to NO-instances because the new vertex v must receive a unique color, this means we must color the original vertices with q colors which cannot be done as our assumption was that we have a NO-instance to PCSP($\mathbf{K}_k, \mathbf{K}_q$). ☐

Corollary 2.14. *There exists a reduction PCSP($\boldsymbol{K}_k, \boldsymbol{K}_q$) $\to$ PCSP($\boldsymbol{K}_{k'}, \boldsymbol{K}_{q'}$) if $k' \geq k \geq 2$ and $k' - k \geq q' - q$.*

Proof. This is obtained by the transitivity of reductions combined with the results of Theorem 2.12 and Theorem 2.13. ☐

2.4.2 Rainbow coloring

We have that PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) NP-hard for $k > q$ if $q \leq 5$ (Theorem 2 [14]). Furthermore PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) is NP-hard if for some $t \geq 1$ and $d \geq 2$ we have that $k \geq td + \lfloor \frac{d}{2} \rfloor$ and $q \leq t(d-1) + 1$ and $k > q \geq 2$ ([1] Theorem 1.3). A concrete consequence of this is that for $k \geq 3$ that PCSP($\mathbf{R}_{k,k-2\lfloor\sqrt{k}\rfloor}, \mathbf{H}_{k,2}$) is NP hard ([1] Corollary 1.4). Furthermore PCSP($\mathbf{R}_{k,k}, \mathbf{H}_{k,2}$) is in P for $k \geq 2$ [13].

We state a few trivial reductions as theorems which we illustrate together with the known NP-hard results in Figure 6.

Theorem 2.15. *We have a gadget reduction PCSP($\boldsymbol{R}_{k,q}, \boldsymbol{H}_{k,2}$) $\to$ PCSP($\boldsymbol{R}_{k,q-1}, \boldsymbol{H}_{k,2}$) for $k > q \geq 3$.*

Proof. Follows immediately from Observation 2.11 ☐

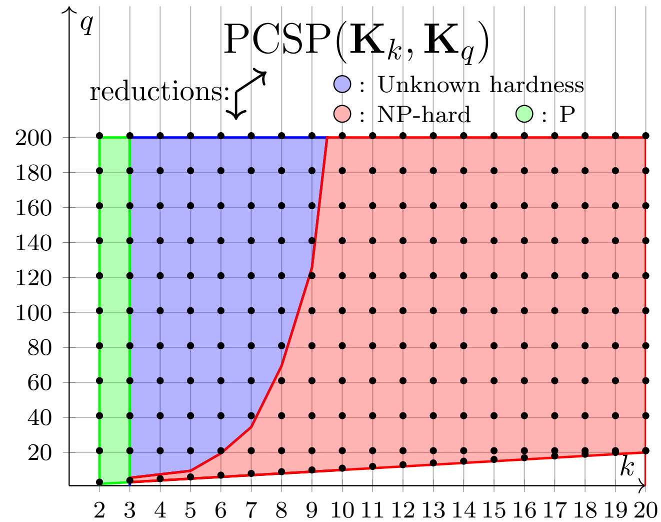

Figure 5: Illustration of complexity results and trivial reduction. As we increase k we quickly get many NP-hard problems, for $k = 20$ we have that PCSP($\mathbf{K}_{20}, \mathbf{K}_{184755}$) is NP-hard.

Theorem 2.16. *We have a gadget reduction PCSP*($\boldsymbol{R}_{k,q}, \boldsymbol{H}_{k,2}$) $\rightarrow$ *PCSP*($\boldsymbol{R}_{k+1,q}, \boldsymbol{H}_{k+1,2}$) *for* $k > q \geq 2$.

Proof. This can be shown with the following reduction:

function PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$)($\mathbf{I}$):
    for each edge $(x_1, ..., x_k)$ replace it with $(x_1, ..., x_k, x_k)$
    return PCSP($\mathbf{R}_{k+1,q}, \mathbf{H}_{k+1,2}$)($\mathbf{I}$)

YES instances are mapped to YES-instances since a valid q-rainbow-coloring of the original hypergraph is valid also for the new hypergraph. Furthermore NO-instances are mapped to NO-instances as if the new hypergraph were to admit a 2-coloring then so would the original one with the same coloring.

□

Corollary 2.17. *There exists a reduction PCSP*($\boldsymbol{R}_{k,q}, \boldsymbol{H}_{k,2}$) $\rightarrow$ *PCSP*($\boldsymbol{R}_{k',q'}, \boldsymbol{H}_{k',2}$) *if* $k' \geq k > q \geq q' \geq 2$

Proof. This is obtained by the transitivity of reductions combined with the results of Theorem 2.15 and Theorem 2.16. □

2.4.3 LO-coloring PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$)

We have that PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) is NP-hard if $r \geq k - l + 4$ ([19] Corollary 27) for $k \geq l \geq 2$.

We state a few trivial reductions as theorems which we illustrate together with the known NP-hard results in Figure 7.

Theorem 2.18. *We have a gadget reduction PCSP*($\boldsymbol{LO}_l^r, \boldsymbol{LO}_k^r$) $\rightarrow$ *PCSP*($\boldsymbol{LO}_l^r, \boldsymbol{LO}_{k-1}^r$) *for* $k > l \geq 2$.

Proof. Follows immediately from Observation 2.11 □

Theorem 2.19. *We have a reduction PCSP*($\boldsymbol{LO}_l^r, \boldsymbol{LO}_k^r$) $\rightarrow$ *PCSP*($\boldsymbol{LO}_{l+1}^r, \boldsymbol{LO}_{k+1}^r$) *for* $k \geq l \geq 2$.

PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$)

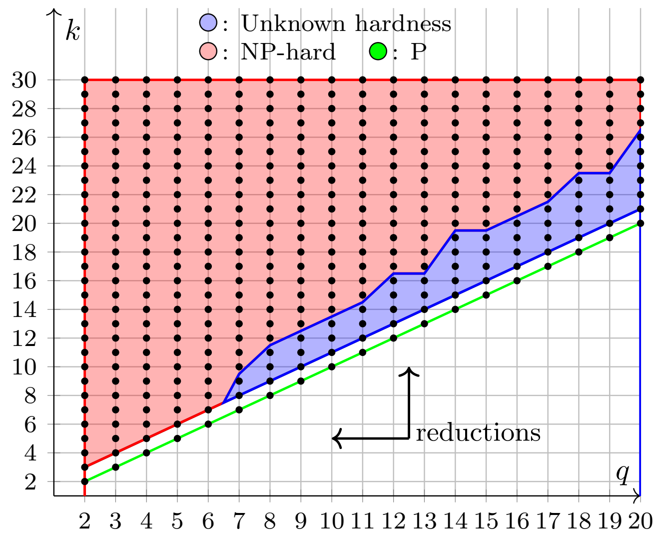

Figure 6: Illustration of complexity results and trivial reductions.

Proof. This can be shown with the following reduction:

```
function PCSP(LO^r_l, LO^r_k)(I):
    add a new node v to I
    for each original vertex u in the hypergraph I add an edge of the form (u, ..., u, v) of arity r.
    return PCSP(LO^r_{l+1}, LO^r_{k+1})(I)
```

YES-instances are mapped to YES-instances as we can LO-color the original vertices with colors $1, ..., l$ and we can color the new vertex v with $(l+1)$. Furthermore NO-instances are mapped to NO-instances because the new vertex v must receive a unique maximal color, this means we must color the original vertices with at most k colors which cannot be done as our assumption was that we have a NO-instance to PCSP($\mathbf{LO}^r_l, \mathbf{LO}^r_k$).

□

Corollary 2.20. *There exists a reduction PCSP($\boldsymbol{LO}^r_l, \boldsymbol{LO}^r_k$) $\rightarrow$ PCSP($\boldsymbol{LO}^r_{l'}, \boldsymbol{LO}^r_{k'}$) if $l' \geq l \geq 2$ and $l' - l \geq k' - k$.*

Proof. This is obtained by the transitivity of reductions combined with the results of Theorem 2.18 and Theorem 2.19. □

2.5 Polymorphisms and theorems

The main goal of this thesis is to classify what reductions do and do not exist between the different types of colorings that that could yield new NP-hardness results for some PCSP's. One way to do this is to just try and gadget reduce various PCSP's to each other and see if it works. This can however become really difficult and if it turns out that no reductions exist one might end up searching forever for something that does not exist. There is however an algebraic approach which gives specific requirements for when gadget reductions exist and do not exist between PCSP's. We present this approach now starting with some definitions in order to understand the important theorems.

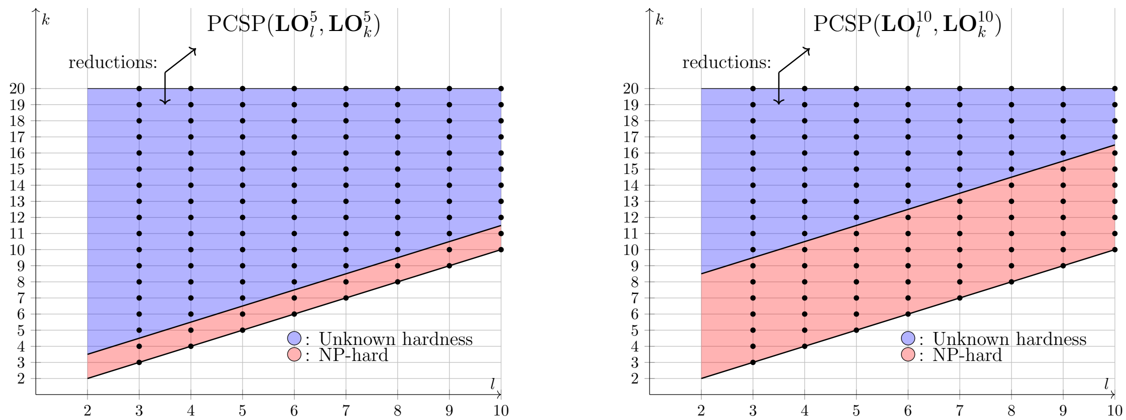

Figure 7: Illustration of complexity results and trivial reductions for $r = 5$ and $r = 10$

Definition 2.21. *The* ***n'th power*** *of a relation on a set* $\boldsymbol{A} = (A; R^A)$ *is the relation on* A^n*;* $\boldsymbol{A}^n = (A^n; R^{A^n})$*. Tuples in* R^{A^n} *are of the form* $(x_1, ..., x_{ar(R^A)})$ *where* $x_i = (x_{i,1}, ..., x_{i,n})$ $\forall i \in [ar(R^A)]$ *such that for all* $j \in [n]$ *we have that* $(x_{1,j}, ..., x_{ar(R^A),j}) \in R^A$*.*

We can view the tuples in R^{A^n} as coming from matrices M of size $\text{ar}(R) \times n$ such that the columns of M are tuples of R^A. Let the rows represent elements of A^n. For any such M the rows form a tuple of size $\text{ar}(R)$ and exactly these tuples are the elements that exist in R^{A^n}.

An example of this is the 6'th power of $\mathbf{K}_3$. Here we have the tuple $((1,1,2,2,3,3),(2,3,1,3,1,2))$ from the matrix

$$\begin{bmatrix} 1 & 1 & 2 & 2 & 3 & 3 \\ 2 & 3 & 1 & 3 & 1 & 2 \end{bmatrix}$$

where all columns are valid relations in $\mathbf{K}_3$. The relation $\mathbf{K}_3^6$ does not however contain the tuple $((1,1,1,1,1,1),(1,3,2,3,2,2))$ as they have the same element in the first coordinate, i.e. for the matrix

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 3 & 2 & 3 & 2 & 2 \end{bmatrix}$$

the first column is not a valid relation in $\mathbf{K}_3$.

Another example is the tuple $((1,2,1,1),(2,1,2,1),(1,1,1,2))$ in $(\mathbf{LO}_2^3)^4$ as for the matrix

$$\begin{bmatrix} 1 & 2 & 1 & 1 \\ 2 & 1 & 2 & 1 \\ 1 & 1 & 1 & 2 \end{bmatrix}$$

we have that each column forms a tuple in the relation $\mathbf{LO}_2^3$.

Definition 2.22. *An* ***n-ary polymorphism*** *from* $\boldsymbol{A}$ *to* $\boldsymbol{B}$ *is a homomorphism from* $\boldsymbol{A}^n$ *to* $\boldsymbol{B}$*. That is a mapping* $f : A^n \to B$ *such that for any combination of n tuples* $(a_{11}, ..., a_{ar(R)1}), ..., (a_{1n}, ..., a_{ar(R)n}) \in R^A$ *we have that*

$$(f(a_{11}, ..., a_{1n}), ..., f(a_{ar(R^A)1}, ..., a_{ar(R^A)n})) \in R^B$$

We use the notation $\text{Pol}(\mathbf{A}, \mathbf{B})$ to denote the set of all polymorphisms from $\mathbf{A}$ to $\mathbf{B}$. We denote $\text{Pol}(\mathbf{A}, \mathbf{A})$ by $\text{Pol}(\mathbf{A})$. Note that n-ary polymorphisms f from $\mathbf{K}_k$ to $\mathbf{K}_q$ can be seen a q-colorings of the graph $\mathbf{K}_k^n$.

A simple example of a polymorphism of $\text{Pol}(\mathbf{A})$ is a dictator, also known as a projection.

Definition 2.23. *A **dictator** on a set A is an operation $p_i^{(n)} : A^n \to A$ of the form $p_i^{(n)}(x_1, x_2, ..., x_n) = x_i$.*

A dictator $p_2^{(6)}$ on $\mathbf{K}_3$ would be $p_2^{(6)} : \mathbf{K}_3^6 \to \mathbf{K}_3$ sending each element of K_3^6 to its second coordinate, for example $p_2(1, 2, 3, 3, 3, 1) = 2$. It is easy to see that this is a valid polymorphism in $\mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_3)$. For the example $((1, 1, 2, 2, 3, 3), (2, 3, 1, 3, 1, 2))$ which is a tuple in the relation $\mathbf{K}_3^6$ we obtain that $(p_2^{(6)}(1, 1, 2, 2, 3, 3), p_2^{(6)}(2, 3, 1, 3, 1, 2)) = (1, 3)$ must be a valid relation in $\mathbf{K}_3$ which it is.

We can also call functions of the form $p_i^{(n)}(x_1, x_2, ..., x_n) = x_i$ in $\mathrm{Pol}(\mathbf{A}, \mathbf{B})$ dictators. Note that such dictators can only exist if $A \subseteq B$ because otherwise it cannot be a well defined mapping. Dictators can however be found in some $\mathrm{Pol}(\mathbf{A}, \mathbf{B})$ even if $A \not\subseteq B$ if we slightly adjust the function. The set $\mathrm{Pol}(\mathbf{A}, \mathbf{B})$ is usually in the context of a corresponding $\mathrm{PCSP}(\mathbf{A}, \mathbf{B})$ which promises the existence of a homomorphism $\phi : \mathbf{A} \to \mathbf{B}$. In such cases dictators can be defined as $p_i^{(n)}(x_1, ..., x_n) = \phi(x_i)$.

Definition 2.24. *An n-ary function $f : A^n \to B$ is called a **minor** of an m-ary function $g : A^m \to B$ if there exists a map $\pi : [m] \to [n]$ if*

$$f(x_1, ..., x_n) = g(x_{\pi(1)}, ..., x_{\pi(m)})$$

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

Definition 2.25. *Let $O(A, B) = \{f : A^n \to B | n \geq 1\}$. A **minion** $\mathcal{M}$ on a pair of sets (A, B) is a non-empty subset of $O(A, B)$ that is closed under taking minors. For a fixed n we denote the set of n-ary functions from of $\mathcal{M}$ by $\mathcal{M}^{(n)}$.*

We have that for any $\mathrm{PCSP}(\mathbf{A}, \mathbf{B})$ that $\mathrm{Pol}(\mathbf{A}, \mathbf{B})$ is a minion. We can also have homorphisms between minions.

Definition 2.26. *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two minions. A mapping $\phi : \mathcal{M}_1 \to \mathcal{M}_2$ is called a **minion homomorphism** if*

1. *it preserves arities; ar(g)=ar($\phi(g)$) for all $g \in \mathcal{M}$ and*
2. *if $f(x_1, ..., x_n) = g(x_{\pi(1)}, ..., x_{\pi(m)})$ for some $f \in \mathcal{M}^{(n)}$, $g \in \mathcal{M}^{(m)}$, and $\pi : [m] \to [n]$, then*

$$\phi(f)(x_1, ..., x_n) = \phi(g)(x_{\pi(1)}, ..., x_{\pi(m)})$$

Definition 2.27. *Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be two minions. **There exists a minion homomorphism** from $\mathcal{M}_1$ to $\mathcal{M}_2$ if there exists a mapping $\phi : \mathcal{M}_1 \to \mathcal{M}_2$ such that ϕ is a minion homomorphism.*

We denote the existence of a minion homomorphism from $\mathcal{M}_1$ to $\mathcal{M}_2$ by $\mathcal{M}_1 \to \mathcal{M}_2$. Now we are ready to introduce our first important theorem.

Theorem 2.28. *[5] Let $(\boldsymbol{A}_1, \boldsymbol{B}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{B}_2)$ be two finite PCSP templates. There exists a polynomial time gadget reduction from PCSP($\boldsymbol{A}_2, \boldsymbol{B}_2$) to PCSP($\boldsymbol{A}_1, \boldsymbol{B}_1$) if and only if Pol($\boldsymbol{A}_1, \boldsymbol{B}_1$) $\to$ Pol($\boldsymbol{A}_2, \boldsymbol{B}_2$).*

Finding a minion homorphism $\mathrm{Pol}(\mathbf{A}_1, \mathbf{B}_1) \to \mathrm{Pol}(\mathbf{A}_2, \mathbf{B}_2)$ can be a tedious task as there are infinitely many potential homomorphisms one could test and there is no clear efficient method to finding one. To alleviate this we thankfully have the so called free structure offering a more systematic approach to proving the existence or absence of minion homomorphisms. We introduce the approach.

Definition 2.29. *Let $\boldsymbol{A} = (A, R^A)$ be a finite relation on the set $A = [n]$, and $\mathcal{M}$ be a minion. The **free structure** of $\mathcal{M}$ generated by $\boldsymbol{A}$ is a relation on the set $\mathcal{M}^{(n)}$ (n-ary functions of $\mathcal{M}$) similar to $\boldsymbol{A}$ denoted $\mathcal{F}_{\mathcal{M}}(\boldsymbol{A}) = (\mathcal{M}^{(n)}, R^{\mathcal{F}_{\mathcal{M}}(\boldsymbol{A})})$. Let R^A be of arity k, and let $m = |R^A|$ and $R^A = \{r_1, r_2, ..., r_m\}$.*

Note that $r_i \in A^k = [n]^k$ for each $i \in [m]$. We have that $R^{\mathcal{F}_{\mathcal{M}}(\boldsymbol{A})}$ is defined as the set of all k-tuples $(f_1, ..., f_k) \in F_{\mathcal{M}}(A)$ such that there exists an m-ary function $g \in \mathcal{M}$ that satisfies

$$f_1(x_1, x_2, ..., x_n) = g(x_{r_1(1)}, x_{r_2(1)}, ..., x_{r_m(1)})$$

$$\vdots$$

$$f_k(x_1, x_2, ..., x_n) = g(x_{r_1(k)}, x_{r_2(k)}, ..., x_{r_m(k)})$$

Let us give a couple examples of this, let $\mathcal{M} = \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)$ and $\mathbf{A} = \mathbf{K}_3$. The free structure $\mathcal{F}_{\text{Pol}(\mathbf{K}_4,\mathbf{K}_5)}(\mathbf{K}_3)$ consists of the set $\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^{(3)}$ with a relation consisting of tuples (f_1, f_2) if and only if

$$f_1(x_1, x_2, x_3) = g(x_1, x_1, x_2, x_2, x_3, x_3)$$

$$f_2(x_1, x_2, x_3) = g(x_2, x_3, x_1, x_3, x_1, x_2)$$

For some $g \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^{(6)}$ where the indexes of the columns form all the tuples of the relation $\mathbf{K}_k$. When we have so few variables we can denote them by x, y, z instead and obtain

$$f_1(x, y, z) = g(x, x, y, y, z, z)$$

$$f_2(x, y, z) = g(y, z, x, z, x, y).$$

Another example is when $\mathcal{M} = \text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)$ and $\mathbf{A} = \mathbf{LO}_2^4$. The free structure here is $\mathcal{F}_{\text{Pol}(\mathbf{LO}_3^4,\mathbf{LO}_4^4)}(\mathbf{LO}_2^4)$ consisting of the set $\text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)^{(2)}$. This free structure has the tuple (f_1, f_2, f_3, f_4) if and only if

$$f_1(x, y) = g(y, x, x, x)$$

$$f_2(x, y) = g(x, y, x, x)$$

$$f_3(x, y) = g(x, x, y, x)$$

$$f_4(x, y) = g(x, x, x, y)$$

For some $g \in \text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)^{(4)}$.

Now we are ready to introduce an important theorem that helps us determine the existence of minion homomorphisms.

Theorem 2.30. *[5] For two PCSP-templates $(\boldsymbol{A}_1, \boldsymbol{B}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{B}_2)$ we have that $\mathcal{F}_{Pol(\boldsymbol{A}_1,\boldsymbol{B}_1)}(\boldsymbol{A}_2) \to \boldsymbol{B}_2$ if and only if there exists a minion homomorphism from Pol$(\boldsymbol{A}_1, \boldsymbol{B}_1)$ to Pol$(\boldsymbol{A}_2, \boldsymbol{B}_2)$.*

The strength of this theorem is that to figure out the existence of a minion homomorphism we do not need information about polymorphisms of all different arities. It is enough to study $\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)$ consisting of polymorphisms of arity $|A_2|$ which contains tuples determined by the existence of polymorphisms of only one other arity. The free structure for a minion $\mathcal{M}$ and relation on set $\mathbf{A}$ is in essence the most general structure $\mathcal{F}$ such that there exists a minion homomorphism from $\mathcal{M}$ to $\text{Pol}(\mathbf{A}, \mathcal{F})$. The free strucure $\mathcal{F}$ has homomorphism to any $\mathbf{B}$ similar to $\mathbf{A}$ such that there is a minion homomorphism from $\mathcal{M}$ to $\text{Pol}(\mathbf{A}, \mathbf{B})$. Note how this reflects the powerful statement of Theorem 2.30. The general idea is in order to figure out whether a minion $\mathcal{M}$ has a minion homomorphism to another minion $\text{Pol}(\mathbf{A}, \mathbf{B})$ one starts from the most general minion $\text{Pol}(\mathbf{A}, \mathcal{F})$ that $\mathcal{M}$ has a minion homorphism to.

Then putting Theorem 2.30 together with Theorem 2.28 yields the following corollary.

Corollary 2.31. *[5] For two PCSP-templates $(\boldsymbol{A}_1, \boldsymbol{B}_1)$ and $(\boldsymbol{A}_2, \boldsymbol{B}_2)$ we have that $\mathcal{F}_{Pol(\boldsymbol{A}_1,\boldsymbol{B}_1)}(\boldsymbol{A}_2) \to \boldsymbol{B}_2$ if and only if PCSP$(\boldsymbol{A}_2, \boldsymbol{B}_2)$ can be gadget reduced in polynomial time to PCSP$(\boldsymbol{A}_1, \boldsymbol{B}_1)$.*

If one is unfamiliar with the free structure and how these theorems are applied it could be a good idea to warm up with a few simple examples before diving in to the proofs in this thesis. These can be found in the appendix (Section A) which can be used to prove the existence of some of the trivial reductions in Section 2.4 with the help of the presented theorems.

3 Results

3.1 Introduction

In this section we present and prove the existence or absence of gadget reductions between graph coloring PCSP's and PCSP's of the other coloring types. We also prove the absence of some more gadget reductions between graph coloring PCSP's.

Several theorems in this thesis state that there is no gadget reduction from PCSP($\mathbf{A}_2, \mathbf{B}_2$) to PCSP($\mathbf{A}_1, \mathbf{B}_1$) for some $\mathbf{A}_1, \mathbf{A}_2, \mathbf{B}_1, \mathbf{B}_2$. These are proven by Corollary 2.31 stating that it is enough to prove $\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2) \not\to \mathbf{B}_2$. We use one of two techniques to prove this statement in our theorems. The first technique is to show that there exists a tuple $(a, a, ..., a)$ of arity $\text{ar}(R^{\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)})$ in $R^{\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)}$ when no such tuple exists in $\mathbf{B}_2$. It is then impossible for a homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2) \to \mathbf{B}_2$ to exist as we would then have that $(\phi(a), \phi(a), ..., \phi(a))$ of arity $\text{ar}(R^{\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)})$ is a tuple in $\mathbf{B}_2$ which by assumption is not the case. The second technique is can be used if $\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)$ has the structure of a graph meaning that $\text{ar}(R^{\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)}) = 2$ and $\mathbf{B}_2 = \mathbf{K}_k$ for some k. The technique is to prove that $\mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2)$ has a clique of size $c > k = |\mathbf{K}_k| = |\mathbf{B}_2|$. This contradicts the existence of a homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{A}_1,\mathbf{B}_1)}(\mathbf{A}_2) \to \mathbf{B}_2$ as all elements of the clique cannot be mapped to distinct elements in $\mathbf{K}_k$ which is needed for the relation to be preserved.

3.2 Graph coloring

In this section we investigate the existence of gadget reductions between graph coloring PCSP's not covered by the trivial reductions in Section 2.4.

3.2.1 PCSP($\mathbf{K}_k, \mathbf{K}_q$) to PCSP($\mathbf{K}_3, \mathbf{K}_c$) for $k \geq 4$.

The following theorem is an already known result (Proposition 10.4 [5]). We however reformulate the proof so that it follows a similar structure to later proofs in this thesis. We also generalise this theorem in Section 3.2.2.

Theorem 3.1. *There is no gadget reduction from PCSP($\boldsymbol{K}_4, \boldsymbol{K}_q$) to PCSP($\boldsymbol{K}_3, \boldsymbol{K}_6$) for any $q \geq 4$.*

Proof. By Corollary 2.31 this is equivalent to there not existing a homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{K}_3,\mathbf{K}_6)}(\mathbf{K}_4) \to \mathbf{K}_q$. In this free structure we have the tuple (h_1, h_2) for $h_1, h_2 \in \text{Pol}(\mathbf{K}_3, \mathbf{K}_6)^{(4)}$ if there exists a $g \in \text{Pol}(\mathbf{K}_3, \mathbf{K}_6)^{(12)}$ such that

$$h_1(a, b, c, d) = g(a, a, a, b, b, b, c, c, c, d, d, d)$$

$$h_2(a, b, c, d) = g(b, c, d, a, c, d, a, b, d, a, b, c).$$

If the tuple (f, f) exists in $\mathcal{F}_{\text{Pol}(\mathbf{K}_3,\mathbf{K}_6)}(\mathbf{K}_4)$ for some $f \in \text{Pol}(\mathbf{K}_3, \mathbf{K}_6)^{(4)}$ the homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{K}_3,\mathbf{K}_6)}(\mathbf{K}_4) \to \mathbf{K}_q$ cannot exist as $(\phi(f), \phi(f))$ is not a tuple in $\mathbf{K}_q$ as $\phi(f) = \phi(f)$. We show this to be the case. The tuple (f, f) exists in the relation if there is a $g \in \text{Pol}(\mathbf{K}_3, \mathbf{K}_6)^{(12)}$ such that

$$f(a, b, c, d) = g(a, a, a, b, b, b, c, c, c, d, d, d)$$

$$f(a, b, c, d) = g(b, c, d, a, c, d, a, b, d, a, b, c)$$

meaning that we must have a 6-coloring of $\mathbf{K}_3^{12}$ such that

$$g(a, a, a, b, b, b, c, c, c, d, d, d) = g(b, c, d, a, c, d, a, b, d, a, b, c).$$

for any $a, b, c, d \in [3]$. Let the 6 colors be $(1, 1)$, $(1, 2)$, $(1, 3)$, $(2, 1)$, $(2, 2)$ and $(2, 3)$. We color elements $x = (x_1, ..., x_{12})$ as follows:

1. $g(x) = (1, a)$ if x follows one of the patterns $(a, a, a, b, b, b, c, c, c, d, d, d)$ or $(b, c, d, a, c, d, a, b, d, a, b, c)$ and $c \neq d$

2. $g(x) = (2, x_{12})$ otherwise

The nontrivial property we need to verify is that there is no monochromatic edge of the color $(1, a)$. In particular that there would be an assignment of values in $[3]$ to variables a, b, c, d such that $(a, a, a, b, b, b, c, c, c, d, d, d)$ and $(b, c, d, a, c, d, a, b, d, a, b, c)$ differ in every coordinate and $c \neq d$ (and thereby form a monochromatic edge). Assume without loss of generality that $c = 1$ and $d = 2$. If $a = c$, $a = d$, $b = c$ or $b = d$ then the 2nd, 3rd, 5th or 6th coordinate has the same value for $(a, a, a, b, b, b, c, c, c, d, d, d)$ and $(b, c, d, a, c, d, a, b, d, a, b, c)$ meaning that they do not have an edge between them. For this not to be the case we must have $a = b = 3$ causing the 1st coordinate to be the same for $(a, a, a, b, b, b, c, c, c, d, d, d)$ and $(b, c, d, a, c, d, a, b, d, a, b, c)$ yielding that they do not to have an edge between them. This shows that our coloring yields no monochromatic edges concluding that $\phi : \mathcal{F}_{\mathrm{Pol}(\mathbf{K}_3,\mathbf{K}_6)}(\mathbf{K}_4) \not\to \mathbf{K}_q$ meaning that there is no gadget reduction from $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_q)$ to $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_6)$

□

We provide an alternate proof using Theorem 2.30.

Proof. By Theorem 2.30 it is enough to show that a minion homomorphism $\mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6) \to \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$ does not exist. This can be done directly by showing that there is a $g \in \mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6)$ such that

$$g(a, a, a, b, b, b, c, c, c, d, d, d) = g(b, c, d, a, c, d, a, b, d, a, b, c)$$

for any $a, b, c, d \in [3]$ (as shown in the proof of Theorem 3.1) but no f in $\mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$ such that

$$f(a, a, a, b, b, b, c, c, c, d, d, d) = f(b, c, d, a, c, d, a, b, d, a, b, c)$$

for any $a, b, c, d \in [4]$ because then by setting $a = 1,\ b = 2,\ c = 3,\ d = 4$ we would have that

$$f(1, 1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4) = f(2, 3, 4, 1, 3, 4, 1, 2, 4, 1, 2, 3)$$

which is impossible as (1,1,1,2,2,2,3,3,3,4,4,4) and (2,3,4,1,3,4,1,2,4,1,2,3) are neighbors in $\mathbf{K}_4^{12}$. This contradicts the existence of a minion homomorphism from $\mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6) \to \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$ because as $\mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6)$ is closed under taking minors we have a $g_1 \in \mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6)^{(4)}$ such that

$$g(a, a, a, b, b, b, c, c, c, d, d, d) = g_1(a, b, c, d) = g(b, c, d, a, c, d, a, b, d, a, b, c)$$

If we then were to have a minion homomorphism $\phi : \mathrm{Pol}(\mathbf{K}_3, \mathbf{K}_6) \to \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$ this would mean that

$$\phi(g)(a, a, a, b, b, b, c, c, c, d, d, d) = \phi(g_1)(a, b, c, d) = \phi(g)(b, c, d, a, c, d, a, b, d, a, b, c)$$

$$\Longrightarrow$$

$$\phi(g)(a, a, a, b, b, b, c, c, c, d, d, d) = \phi(g)(b, c, d, a, c, d, a, b, d, a, b, c)$$

for some $\phi(g) = f \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$. This gives us the contradiction as we have shown that no such f exists in $\mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_q)$. □

It follows from Theorem 3.1 that there is no gadget reduction $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q) \to \mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_c)$ for any $q \geq k \geq 4$ and $c \geq 6$ because by Section 2.4 we have the reductions $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_c) \to \mathrm{PCSP}(\mathbf{K}_3 \mathbf{K}_6)$ and $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_{q'}) \to \mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q)$ for some $q' \geq 4$ which together with Theorem 3.1 contradict the existence of such a gadget reduction. This means that no NP-hardness result of $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q)$ for $q \geq k \geq 4$ can imply any new NP-hardness results for problems of the form $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_c)$ with a gadget reduction. We illustrate Theorem 3.1 in Figure 8.

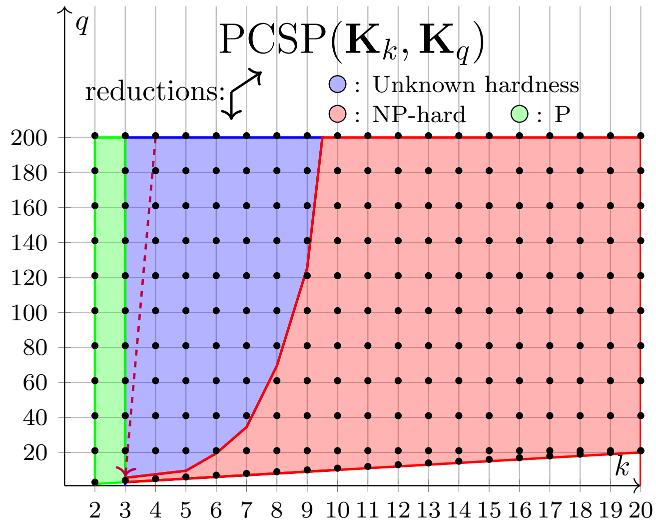

Figure 8: The purple dashed arrow symbolises the absence of a gadget reduction

3.2.2 PCSP($\mathbf{K}_k, \mathbf{K}_q$) to PCSP($\mathbf{K}_l, \mathbf{K}_c$) for $k > l \geq 3$.

Theorem 3.2. *There is no gadget reduction from PCSP($\boldsymbol{K}_{l+1}, \boldsymbol{K}_q$) to PCSP($\boldsymbol{K}_l, \boldsymbol{K}_{2l}$) for any $q \geq l+1$.*

Proof. This proof follows the ideas of the second proof of Theorem 3.1 we can show that there exists a $g \in \text{Pol}(\mathbf{K}_l, \mathbf{K}_{2l})^{(l(l+1))}$ such that

$$g(x_1, x_1, x_1, ..., x_{l+1}, x_{l+1}, x_{l+1}) =$$

$$g(x_2, x_3, x_4, ..., x_{l-2}, x_{l-1}, x_l)$$

to prove the absence of a minion homomorphism from $\text{Pol}(\mathbf{K}_l, \mathbf{K}_{2l})$ to $\text{Pol}(\mathbf{K}_{l+1}, \mathbf{K}_q)$. The $l(l+1)$ coordinates creates $l(l+1)$ pairs of indexes which are exactly to the tuples in $\mathbf{K}_{l+1}$. Such a g can be represented as a $(2l)-$coloring of $\mathbf{K}_l^{(l(l+1))}$ such that the equality

$$g(x_1, x_1, x_1, ..., x_{l+1}, x_{l+1}, x_{l+1}) = g(x_2, x_3, x_4, ..., x_{l-2}, x_{l-1}, x_l)$$

holds. Let the $2l$ colors be denoted as $(1,1), (1,2), ..., (1,l), (2,1), (2,2), ..., (2,l)$. We can define $g(y) = g(y_1, y_2, y_3, ..., y_{l(l+1)})$ as follows:

- $g(y) = (1, y_1)$ if y follows one of the patterns above and $y_l \neq y_{l+1}$.
- $g(y) = (2, y_{l+1})$ otherwise.

The nontrivial property to verify is that there is no monochromatic edge of the color $(1, y_1)$. In particular that there would be an assignment of values in $[l]$ to variables $x_1, .., x_l$ such that $(x_1, x_1, x_1, ..., x_{l+1}, x_{l+1}, x_{l+1})$ and $(x_2, x_3, x_4, ..., x_{l-2}, x_{l-1}, x_l)$ differ in every coordinate and $x_l \neq x_{l+1}$ (and thereby form a monochromatic edge). We show that this cannot be the case by showing that two such vertices have a common value at at least one coordinate, that is $x_i = x_j$ for some $i \neq j$. We know that $x_l \neq x_{l+1}$ (requirement for vertices having color $(1, y_1)$). Now for any two vertices of color $(1, y_1)$ we can define two cases can define two cases:

- **Case 1:** For some $i < l$ we have that $x_i = x_l$ or $x_i = x_{l+1}$

- **Case 2:** $x_i \neq x_l$ and $x_i \neq x_{l+1}$ for all $i < l$

If **Case 1** holds then we have $x_i = x_j$ for $j = l$ or $j = l+1$ and we are done since we then have have a coordinate for which the two vertices share a value. Next if **Case 2** holds this means that we need to assign each of the variables $x_1, ..., x_{l-1}$ one value out of $l-2$ total values, as two of the l colors are reserved for x_l and x_{l+1}. By the pigeon hole principal at least two variables x_i and x_j are given the same value implying that we also here have a coordinate where two vertices share the same value. We have now proven the existence of a $g \in \mathrm{Pol}(\mathbf{K}_l, \mathbf{K}_{2l})^{(l(l+1))}$ that the required equality holds.

Furthermore,there exists no $f \in \mathrm{Pol}(\mathbf{K}_{l+1}, \mathbf{K}_q)$ such that

$$f(x_1, x_1, x_1, ..., x_{l+1}, x_{l+1}, x_{l+1}) =$$

$$f(x_2, x_3, x_4, ..., x_{l-2}, x_{l-1}, x_l)$$

because assigning each x_i the value i gives us that any f would yield a monochramtic edge between $(1, 1, 1, ..., l+1, l+1, l+1)$ and $(2, 3, 4, ..., l-2, l-1, l)$. This means that g cannot to be mapped to a function $f \in \mathrm{Pol}(\mathbf{K}_{l+1}, \mathbf{K}_q)$ such that the equality is preserved contradicting the existence of the minion homomorphism which in turn proves the absence of a gadget reduction from $\mathrm{PCSP}(\mathbf{K}_{l+1}, \mathbf{K}_q)$ to $\mathrm{PCSP}(\mathbf{K}_l, \mathbf{K}_{2l})$ for any $q \geq l+1$.

□

It follows from Theorem 3.2 that there is no gadget reduction $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q) \to \mathrm{PCSP}(\mathbf{K}_l, \mathbf{K}_c)$ for any $q \geq k \geq l+1$ and $c \geq 2l$ because by Section 2.4 we have the reductions $\mathrm{PCSP}(\mathbf{K}_l, \mathbf{K}_c) \to \mathrm{PCSP}(\mathbf{K}_l \mathbf{K}_{2l})$ and $\mathrm{PCSP}(\mathbf{K}_{l+1}, \mathbf{K}_{q'}) \to \mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q)$ for some $q' \geq l+1$ which together with Theorem 3.2 contradict the existence of such a gadget reduction. This means that no NP-hardness result of $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q)$ can through a gadget reduction imply any new hardness results for problems of the form $\mathrm{PCSP}(\mathbf{K}_l, \mathbf{K}_c)$ for any $k > l \geq 3$. We illustrate Theorem 3.2 in Figure 9.

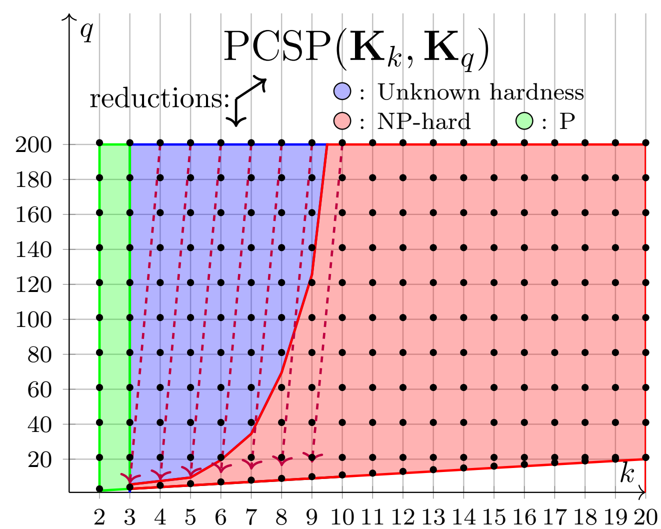

Figure 9: The purple dashed arrows symbolise the absence of a gadget reduction

3.2.3 $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_k)$ to $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_q)$ for $q > k \geq 3$

It is of interest to classify when gadget reductions exist from $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_q)$ to $\mathrm{PCSP}(\mathbf{K}_{k'}, \mathbf{K}_{q'})$ for $q' > q$. We know that a reduction exists when $k' - k \geq q' - q$. To gain some knowledge on whether a gadget reduction exists for other cases we study the special case of whether we have a gadget reduction from PCSP($\mathbf{K}_3, \mathbf{K}_k$) to PCSP($\mathbf{K}_4, \mathbf{K}_q$) for $q - k \geq 2$. We begin with a gadget reduction that does not exist.

Theorem 3.3. *There is no gadget reduction from PCSP($\boldsymbol{K}_3, \boldsymbol{K}_5$) to PCSP($\boldsymbol{K}_4, \boldsymbol{K}_8$).*

Proof. Using Corollary 2.31 we know that it is enough to show that there exists no homomorphism $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3) \to \mathbf{K}_5$. The free structure contains the tuple (h_1, h_2) where $h_1, h_2 \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_8)^{(3)}$ if there exists a $g \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_8)^{(6)}$ such that

$$h_1(a, b, c) = g(a, a, b, b, c, c)$$

$$h_2(a, b, c) = g(b, c, a, c, a, b).$$

We show that the homomorphism does not exist by proving that $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ contains a 6-clique, i.e 6 functions $f_1, ..., f_6$ that all are related to each other. This contradicts the existence of the homomorphism because if it existed all 6 functions would have to be mapped to 6 different elements to preserve the relation. The problem is that $\mathbf{K}_5$ only contains 5 elements making the homomorphism impossible. We define the functions as follows.

- $f_1(a, b, c) = a$
- $f_2(a, b, c) = b$
- $f_3(a, b, c) = c$
- $f_4(a, b, c) = a$ if $a = b = c$, otherwise $f_4(a, b, c) = a + 4$
- $f_5(a, b, c) = b$ if $a = b = c$, otherwise $f_5(a, b, c) = b + 4$
- $f_6(a, b, c) = c$ if $a = b = c$, otherwise $f_6(a, b, c) = c + 4$

Note that all these colorings can in essence be seen as dictators with the requirement that $f_1(a, a, a) = ... = f_6(a, a, a)$. Let us show that all of these are neighbors with the following cases:

- **f_1 and f_2 are neighbors:** Let g be the dictator $p_1^{(6)}$. The mapping $p_1^{(6)}$ is obviously a valid homomorphism and it maps vertices of the form (a, a, b, b, c, c) to $a = f_1(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $b = f_2(a, b, c)$ exactly as g should.

- **f_1 and f_3 are neighbors:** Let g be the dictator $p_2^{(6)}$. The mapping $p_2^{(6)}$ is obviously a valid homomorphism and it maps vertices of the form (a, a, b, b, c, c) to $a = f_1(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $c = f_3(a, b, c)$ exactly as g should.

- **f_2 and f_3 are neighbors:** Let g be the dictator $p_4^{(6)}$. The mapping $p_4^{(6)}$ is obviously a valid homomorphism and it maps vertices of the form (a, a, b, b, c, c) to $b = f_2(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $c = f_3(a, b, c)$ exactly as g should.

- **f_i and f_j are neighbors for $i \in \{1, 2, 3\}$ and $j \in \{4, 5, 6\}$:** Let

  $$g(x) = \begin{cases} f_j(a, b, c) & \text{if } x \text{ is of the form } (b, c, a, c, a, b) \\ p_{2i}^{(6)}(x) & \text{else,} \end{cases}$$

  g is a valid polymorphisms due to both f_j and $p_{2i}^{(6)}$ being polymorphisms mapping elements to two distinct sets guaranteeing that we have no monochromatic edges, except that they send (a, a, a) to the same color but this yields no monochromatic edges. Furthermore vertices of the form (b, c, a, c, a, b) are sent to $f_j(a, b, c)$ as they should and vertices of the form (a, a, b, b, c, c) are mapped to $f_i(a, b, c) = p_{2i}^{(6)}(a, a, b, b, c, c)$ as they should.

- f_4 **and** f_5 **are neighbors:** Let

$$g(x) = \begin{cases} p_1^{(6)}(x) + 4 & \text{if } x \text{ is of the form } (a,a,b,b,c,c) \text{ or } (b,c,a,c,a,b) \text{ such that } a,b,c \text{ are not all equal.} \\ p_1^{(6)}(x) & \text{else,} \end{cases}$$

  g is a valid polymorphisms due to both f_j and $p_1^{(6)}$ being polymorphisms mapping elements to two distinct sets guaranteeing that we have no monochromatic edges. For a, b, c not being all equal we have that the g maps vertices of the form (a, a, b, b, c, c) to $a + 4 = f_4(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $b + 4 = f_5(a, b, c)$ exactly as g should.

- f_4 **and** f_6 **are neighbors:** Let

$$g(x) = \begin{cases} p_2^{(6)}(x) + 4 & \text{if } x \text{ is of the form } (a,a,b,b,c,c) \text{ or } (b,c,a,c,a,b) \text{ such that } a,b,c \text{ are not all equal.} \\ p_1^{(6)}(x) & \text{else,} \end{cases}$$

  g is a valid polymorphisms due to both f_j and $p_1^{(6)}$ being polymorphisms mapping elements to two distinct sets guaranteeing that we have no monochromatic edges. For a, b, c not being all equal we have that the g maps vertices of the form (a, a, b, b, c, c) to $a + 4 = f_4(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $c + 4 = f_6(a, b, c)$ exactly as g should.

- f_5 **and** f_6 **are neighbors:** Let

$$g(x) = \begin{cases} p_4^{(6)}(x) + 4 & \text{if } x \text{ is of the form } (a,a,b,b,c,c) \text{ or } (b,c,a,c,a,b) \text{ such that } a,b,c \text{ are not all equal.} \\ p_1^{(6)}(x) & \text{else,} \end{cases}$$

  g is a valid polymorphisms due to both f_j and $p_1^{(6)}$ being polymorphisms mapping elements to two distinct sets guaranteeing that we have no monochromatic edges. For a, b, c not being all equal we have that the g maps vertices of the form (a, a, b, b, c, c) to $b + 4 = f_5(a, b, c)$ and vertices of the form (b, c, a, c, a, b) to $c + 4 = f_6(a, b, c)$ exactly as g should.

We have now shown that $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ has a 6-clique proving that the homomorphism does not exist allowing us to conclude that there is no gadget reduction from $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_5)$ to $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_8)$ □

We illustrate this result in Figure 10. Note that it would have been surprising if such a gadget reduction existed as it would have yielded a new NP-hardness result for an extensively researched problem. This result can with a similar proof be expanded to show the non-existence of a gadget reduction from $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_{2k-1})$ to $\mathrm{PCSP}(\mathbf{K}_{k+1}, \mathbf{K}_{2k+2})$ for $k \geq 3$. It follows that there is no gadget reduction from $\mathrm{PCSP}(\mathbf{K}_k, \mathbf{K}_{2k-1})$ to $\mathrm{PCSP}(\mathbf{K}_{k+1}, \mathbf{K}_q)$ for $q \geq 2k + 2$ due to the trivial reductions from Section 2.4.

From the trivial reductions in Section 2.4 we know that a reduction exists from $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_7)$ to $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_8)$. So the remaining interesting question is if a gadget reduction exists from $\mathrm{PCSP}(\mathbf{K}_3, \mathbf{K}_6)$ to $\mathrm{PCSP}(\mathbf{K}_4, \mathbf{K}_8)$ which is equivalent to the existence of a homomorphism from $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3) \to \mathbf{K}_6$. The free structure contains the tuple (h_1, h_2) where $h_1, h_2 \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_8)^{(3)}$ if there exists a $g \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_8)^{(6)}$ such that

$$h_1(a, b, c) = g(a, a, b, b, c, c)$$

$$h_2(a, b, c) = g(b, c, a, c, a, b).$$

A standard way to prove the existence of the homomorphism $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3) \to \mathbf{K}_6$ is by defining it by mapping $f \in \mathrm{Pol}(\mathbf{K}_4, \mathbf{K}_8)^{(3)}$ to $f(a, b, c)$ for some $a, b, c \in [4] : a \neq b \neq c \neq a \in [4]$. This however does not work because this would mean that some f are mapped to 7 or 8 which are not in $\mathbf{K}_6$. We could handle all f mapped to 7 by instead mapping them to $f(d,d,d)$ for $d \in [4]\backslash\{a,b,c\}$. However, we cannot then handle the f mapped to 8.

The next approach we can try is to instead prove the absence of the homomorphism $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3) \to \mathbf{K}_6$. There are two good ways one can go about this. The first one is showing that the tuple (f,f) exists in the relation for some $f \in \mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)^{(3)}$ because such an f. This however does not work as this implies the existence of a g such that

$$g(a,a,b,b,c,c) = g(b,c,a,c,a,b)$$

for any $a,b,c,\in[4]$ which cannot hold as we would have the equality

$$g(1,1,2,2,3,3) = g(2,3,1,3,1,2)$$

yielding a monochromatic edge.

The second way we can try proving the non-existence of the homomorphism $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3) \to \mathbf{K}_6$ is by showing that $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ has a 7-clique as the 7 elements of the clique would then have to be mapped to 7 different elements in $\mathbf{K}_6$ which is impossible as $\mathbf{K}_6$ only contains 6 elements. We show however that this method does not work either.

Proposition 3.4. *$\mathcal{F}_{Pol(\boldsymbol{K}_4,\boldsymbol{K}_8)}(\boldsymbol{K}_3)$ does not contain a 7-clique.*

Proof. Assume we do have a 7-clique $\{f_1,...,f_7\}$ in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$. It must be the case that $f_i(a,a,a) = f_j(a,a,a)$ for all $i,j \in [7] : i \neq j$ and $a \in [4]$. Because (f_i,f_j) is a tuple in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ by assumption meaning that there exists a $g \in \mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)^{(6)}$ such that

$$f_i(x,y,z) = g(x,x,y,y,z,z)$$

$$f_j(x,y,z) = g(y,z,x,z,x,y).$$

This implies that

$$f_i(a,a,a) = g(a,a,a,a,a,a) = f_j(a,a,a),$$

which in turn implies that $f_i(a,a,a) = f_j(a,a,a)$ for all $i,j \in [7] : i \neq j$ and $a \in [4]$. Next it must be the case that $f_i(1,2,3) \neq f_j(1,2,3)$ for all $i \neq j : i,j \in [7]$. Because if this were not the case we would have that since (f_i,f_j) is a tuple in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ we have that

$$g(1,1,2,2,3,3) = f_i(1,2,3) = f_j(1,2,3) = g(2,3,1,3,1,2)$$

which would yield a monochratic edge between $(1,1,2,2,3,3)$ and $(2,3,1,3,1,2)$ in $\mathbf{K}_4^6$ meaning that g would not be a valid polymorphism which is not the case. We conclude that we cannot have $f_i(1,2,3) = f_j(1,2,3)$ for $i \neq j$. Now assume that $f_i(4,4,4) = x \in [8]$ for all $i \in 7$. Then it must be the case the set $\{f_1(1,2,3),...,f_7(1,2,3)\} = [8]\backslash\{x\}$ as $f_i(1,2,3) \neq f_i(4,4,4)$ for all $i \in [7]$ because otherwise f_i would not be a valid polymorphism as $(1,2,3)$ and $(4,4,4)$ would form a monochromatic edge in $\mathbf{K}_4^3$. Similarly we have that if $f_i(3,3,3) = y$ for all $i \in [7]$ that the set $\{f_1(4,1,2),...,f_7(4,1,2)\} = [8]\backslash\{y\}$.

Now assume without loss of generality that $f_i(a,a,a) = a$ for all $i \in [7]$. Now for elements $a,b,c,d \in [4]$ such that they are all different we have that $\{f_1(a,b,c),...,f_7(a,b,c)\} = [8]\backslash\{d\}$ because $f_i(a,b,c) \neq f_i(d,d,d) = d$ for all $i \in [7]$. This means that for any $a,b,c \in [4] : a \neq b \neq c \neq a$ we have that $f_i(a,b,c) = 5$ for some $i \in [7]$ because $5 \neq f_i(d,d,d)$ for all $i \in [7]$ and $d \in [4]$. Now assume w.l.o.g that $f_1(1,2,3) = 5$. We also must for some $i \in [7]$ have that $f_i(4,2,3) = 5$. If $i \neq 1$ then f_1 and f_i cannot be neighbors as then for some $g \in \mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_8)^{(6)}$ we would have that

$$g(1,1,2,2,3,3) = f_1(1,2,3) = 1 = f_i(4,2,3) = g(2,3,4,3,4,2)$$

implying that

$$g(1,1,2,2,3,3) = g(2,3,4,3,4,2)$$

which is a contradiction to g being a valid polymorphism. This means that we must have that $f_i(4,2,3) = 5$ for $i = 1$, i.e $f_1(1,2,3) = f_1(4,2,3)$. This can be generalised to if $f_1(a,b,c) = 5$ then $f_1(a,b,c) = f_1(a,b,d) = f_1(a,d,c) = f_1(d,b,c) = 5$ for $a,b,c,d \in [4]$ such that a,b,c,d are all different. Using the generalisation and assumption that $f_1(1,2,3) = 5$ we obtain

$$f_1(1,2,3) = 5 = f_1(4,2,3) = f_1(4,1,3) = f_1(4,1,2) = f_1(3,1,2)$$

implying that $f_1(1,2,3) = f_1(3,1,2)$ contradicting that f_1 is a valid polymorphism. This allows us to conclude that $F_{\text{Pol}(\mathbf{K}_4,\mathbf{K}_8)}(\mathbf{K}_3)$ does not contain a 7-clique.

□

This makes determining the existence of a gadget reduction PCSP($\mathbf{K}_3, \mathbf{K}_6$) to PCSP($\mathbf{K}_4, \mathbf{K}_8$) quite challenging as none of these useful methods work.

There is however another gadget reduction that we can prove the absence of.

Theorem 3.5. *There is no gadget reduction from PCSP($\boldsymbol{K}_3, \boldsymbol{K}_6$) to PCSP($\boldsymbol{K}_4, \boldsymbol{K}_{36}$)*

Proof. Using Corollary 2.31 we know that it is enough to show that there exists no homomorphism $\mathcal{F}_{\text{Pol}(\mathbf{K}_4,\mathbf{K}_{36})}(\mathbf{K}_3) \to \mathbf{K}_6$. The free structure contains the tuple (h_1, h_2) where $h_1, h_2 \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_{36})^{(3)}$ if there exists a $g \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_{36})^{(6)}$ such that

$$h_1(a,b,c) = g(a,a,b,b,c,c)$$

$$h_2(a,b,c) = g(b,c,a,c,a,b).$$

We show that the homomorphism does not exist by proving that $\mathcal{F}_{\text{Pol}(\mathbf{K}_4,\mathbf{K}_{36})}(\mathbf{K}_3)$ contains a 7-clique, i.e 7 functions $f_1, ..., f_7$ that all are related to each other. This contradicts the existence of the homomorphism because if it existed all 7 functions would have to be mapped to 7 different elements to preserve the relation. The problem is that $\mathbf{K}_6$ only contains 6 elements making the homomorphism impossible. The functions f_i we can see as 36-colorings of $\text{Pol}(\mathbf{K}_4, \mathbf{K}_{36})^{(3)}$ and g as a 36-coloring of $\text{Pol}(\mathbf{K}_4, \mathbf{K}_{36})^{(6)}$. Let the 36 colors be (1,1),...,(1,4), (2,1),...,(2,4), (3,1),...,(3,7), (4,1),...,(4,7), (5,1),...,(5,7), (6,1),...,(6,7). We define each f_i using the colors (2,1),...,(2,4), (3,1),...,(3,7), (4,1),...,(4,7), (5,1),...,(5,7), (6,1),...,(6,7) and reserve the colors (1,1),...,(1,4) for later:

- $f_i(a,a,b) = f_i(a,b,a) = f_i(b,a,a) = (2,a)$. If a is in the majority of the coordinates the vertex is given the color a.
- $f_i(a,b,c)$ for $(a,b,c) \in \{1,2,3\}^3$ for $a \neq b \neq c \neq a$: See Table 1.
- $f_i(a,b,c)$ for $(a,b,c) \in \{1,2,4\}^3$ for $a \neq b \neq c \neq a$: We can color this in an analagous way to when $(a,b,c) \in \{1,2,3\}^3$ for $a \neq b \neq c \neq a$ using the colors (4,1),...,(4,7) instead.
- $f_i(a,b,c)$ for $(a,b,c) \in \{1,3,4\}^3$ for $a \neq b \neq c \neq a$: We can color this in an analagous way to when $(a,b,c) \in \{1,2,3\}^3$ for $a \neq b \neq c \neq a$ using the colors (5,1),...,(5,7) instead.
- $f_i(a,b,c)$ for $(a,b,c) \in \{2,3,4\}^3$ for $a \neq b \neq c \neq a$: We can color this in an analagous way to when $(a,b,c) \in \{1,2,3\}^3$ for $a \neq b \neq c \neq a$ using the colors (4,1),...,(4,7) instead.

In order to show that these form a 7-clique we must show that there exists a $g \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_{36})^{(6)}$ such that

$$f_i(a,b,c) = g(a,a,b,b,c,c)$$

$$f_j(a,b,c) = g(b,c,a,c,a,b)$$

For any $i,j \in [7] : i \neq j$ We define g in the following way:

$$g(x) = \begin{cases} f_i(a,b,c) & \text{if } x \text{ is of the form } (a,a,b,b,c,c) \\ f_j(a,b,c) & \text{if } x \text{ is of the form } (b,c,a,c,a,b) \\ (1, p_1^{(6)}(x)) & \text{else} \end{cases}$$

$f_1(1,2,3) = (3,1)$	$f_2(1,2,3) = (3,2)$	$f_3(1,2,3) = (3,3)$	$f_4(1,2,3) = (3,4)$	$f_5(1,2,3) = (3,5)$
$f_1(1,3,2) = (3,2)$	$f_2(1,3,2) = (3,3)$	$f_3(1,3,2) = (3,4)$	$f_4(1,3,2) = (3,5)$	$f_5(1,3,2) = (3,6)$
$f_1(2,1,3) = (3,3)$	$f_2(2,1,3) = (3,4)$	$f_3(2,1,3) = (3,5)$	$f_4(2,1,3) = (3,6)$	$f_5(2,1,3) = (3,7)$
$f_1(2,3,1) = (3,4)$	$f_2(2,3,1) = (3,5)$	$f_3(2,3,1) = (3,6)$	$f_4(2,3,1) = (3,7)$	$f_5(2,3,1) = (3,1)$
$f_1(3,1,2) = (3,5)$	$f_2(3,1,2) = (3,6)$	$f_3(3,1,2) = (3,7)$	$f_4(3,1,2) = (3,1)$	$f_5(3,1,2) = (3,2)$
$f_1(3,2,1) = (3,6)$	$f_2(3,2,1) = (3,7)$	$f_3(3,2,1) = (3,1)$	$f_4(3,2,1) = (3,2)$	$f_5(3,2,1) = (3,3)$
$f_6(1,2,3) = (3,6)$	$f_7(1,2,3) = (3,7)$			
$f_6(1,3,2) = (3,7)$	$f_7(1,3,2) = (3,1)$			
$f_6(2,1,3) = (3,1)$	$f_7(2,1,3 = (3,2))$			
$f_6(2,3,1) = (3,2)$	$f_7(2,3,1) = (3,3)$			
$f_6(3,1,2) = (3,3)$	$f_7(3,1,2) = (3,4)$			
$f_6(3,2,1) = (3,4)$	$f_7(3,2,1) = (3,5)$			

Table 1: Colorings of vertices $(a, b, c) \in \{1, 2, 3\}^3$ for $a \neq b \neq c \neq a$.

We need to check that the coloring g does not yield a monochromatic edge. There is trivially no monochromatic edge of a color $(1, x)$. Furthermore there is no monochromatic edge of a color$(2, x)$ as a vertex of such a color has an x in four of its entries meaning that any two vertices given this color have at least 2 coordinates where they both have an x meaning that they do not have an edge between them. Lastly we must make sure that there is no monochromatic edge of a color $(3, x)$, $(4, x)$, $(5, x)$, or $(6, x)$. It is enough to show it for $(3, x)$ as if there is no mononochromatic edge of a color $(3, x)$ there is not a monochromatic edge of the other colors. What we need to check is that no monochromatic edge exists between vertices of the form $(a_1, a_1, b_1, b_1, c_1, c_1)$ and $(b_2, c_2, a_2, c_2, a_2, b_2)$ for $a_i, b_i, c_i \in \{1, 2, 3\} : a_i \neq b_i \neq c_i \neq a_i$ for $i \in [2]$. The only way these vertices share an edge is if $a_1 = a_2$, $b_1 = b_2$ and $c_1 = c_2$ because if for example $b_1 = a_2$ then the vertices would have the same value in the third coordinate implying that the vertices do not share an edge. We know however that these vertices are given different colors if $a_1 = a_2$, $b_1 = b_2$ and $c_1 = c_2$ as

$$f_i(a_1, b_1, c_1) = g(a_1, a_1, b_1, b_1, c_1, c_1)$$

$$f_j(a_1, b_1, c_1) = g(b_1, c_1, a_1, c_1, a_1, b_1),$$

and the colorings f_i and f_j are defined so that $f_i(a_1, b_1, c_1) \neq f_j(a_1, b_1, c_1)$ for $i \neq j$ implying that $g(a_1, a_1, b_1, b_1, c_1, c_1) \neq g(b_1, c_1, a_1, c_1, a_1, b_1)$. This concludes that we indeed have a 7-clique in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_4,\mathbf{K}_{36})}(\mathbf{K}_3)$ proving that the homomorphism does not exist and we can thereby conclude that there is no gadget reduction from PCSP($\mathbf{K}_3, \mathbf{K}_6$) to PCSP($\mathbf{K}_4, \mathbf{K}_{36}$)

□

This result is illustrated in Figure 10. It would be interesting to investigate the existence of a gadget reduction from PCSP($\mathbf{K}_3, \mathbf{K}_6$) to PCSP($\mathbf{K}_4, {}_q$) for some q closer to 8 then 36. In this thesis we unfortunately did not have the time to determine the existence of such gadget reductions.

3.3 Rainbow coloring

3.3.1 PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) to PCSP($\mathbf{K}_l, \mathbf{K}_c$)

First we show the absence of a specific gadget reductions and then combine the result with the existing reductions from Section 2.4 to show that no NP-hardness result of rainbow coloring PCSP's can imply any new hardness results for graph coloring PCSP's.

Theorem 3.6. *There is no gadget reduction from PCSP($\boldsymbol{R}_{k,k-1}, \boldsymbol{H}_{k,2}$) to PCSP($\boldsymbol{K}_c, \boldsymbol{K}_{2c}$) for any $k, c \geq 3$*

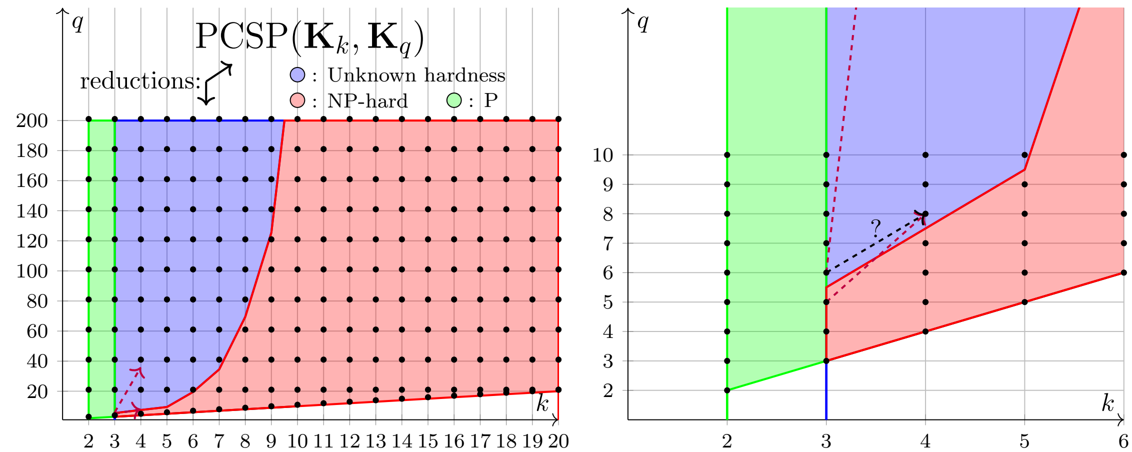

Figure 10: The purple dashed arrow symbolises the absence of a gadget reduction. The left graph is a zoomed in version of the right one. The black dashed arrow symbolises the gadget reduction PCSP($\mathbf{K}_3, \mathbf{K}_6$) to PCSP($\mathbf{K}_4, \mathbf{K}_8$) that we would like to know the existence of.

Proof. Corollary 2.31 gives us that it is enough to show that there exists no homomorphism $\phi : \mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{R}_{k,k-1}) \to \mathbf{H}_{k,2}$. In $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_3,\mathbf{K}_6)}(\mathbf{R}_{k,k-1})$ we have the tuple $(f_1, f_2, ..., f_k)$ for $f_1, f_2, ..., f_k \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^{k-1}$ if there exists a $g \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^M$ such that

$$f_1(x_1, ..., x_{k-1}) = g(x_{i_{1,1}}, x_{i_{1,2}}, ..., x_{i_{1,M}})$$
$$f_2(x_1, ..., x_{k-1}) = g(x_{i_{2,1}}, x_{i_{2,2}}, ..., x_{i_{2,M}})$$
$$\vdots$$
$$f_k(x_1, ..., x_{k-1}) = g(x_{i_{k,1}}, x_{i_{k,2}}, ..., x_{i_{k,M}})$$

where each index $i_{k,l} \in [k-1]$. The columns of indexes form exactly the all the tuples in $\mathbf{R}_{k,k-1}$. That means that we have $M = k!\frac{k-1}{2}$ columns. We can calculate M by first imagining us having k different indexes, giving us $k!$ orderings, next we choose one out of the $k-1$ indexes to appear twice giving us $(k-1)k!$ orderings, but since one index appears twice we count each ordering twice, so we divide by 2 and obtain $M = k!\frac{k-1}{2}$.

What we want to show is that in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{R}_{k,k-1})$, $(f_1, f_2, ..., f_k)$ is a tuple in the relation even if $f_1 = f_2 = ... = f_k$ because this would contradict the existence of a homomorphism $\phi : \mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{R}_{k,k-1}) \to \mathbf{H}_{k,2}$ as $(\phi(f_1), \phi(f_2), ..., \phi(f_k)) = ((\phi(f_1), \phi(f_1), ..., \phi(f_1)) = (a, a, ..., a)$ for some $a \in \{1, 2\}$ would have to be a tuple in $\mathbf{H}_{k,2}$ which it is not. We do this by showing that there exists a $g \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^M$, or in other words a $2c$-coloring g of $\mathbf{K}_c^M$, such that

$$g(x_{i_{1,1}}, x_{i_{1,2}}, ..., x_{i_{1,M}}) =$$
$$g(x_{i_{2,1}}, x_{i_{2,2}}, ..., x_{i_{2,M}}) =$$
$$\vdots$$
$$= g(x_{i_{k,1}}, x_{i_{k,2}}, ..., x_{i_{k,M}}).$$

Note that since $f_1, ..., f_k \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^{k-1}$ we have that $x_1, ..., x_{k-1} \in [c]$. Now we color the vertices $y = (y_1, ..., y_M)$ in $\mathbf{K}_c^M$ with $2c$ colors denoted $(1,1), (1,2), ..., (1,c), (2,1), (2,2), ..., (2,c)$ in the following way:

- $(1, x_{i_{1,1}})$ if it follows one of the patterns $(x_{i_{j,1}}, ..., x_{i_{j,M}})$ above.
- $(2, y_1)$ otherwise.

There are two nontrivial properties we need to verify. The first being that given a vertex $y = (y_1, ..., y_M)$ matching one of the patterns above, it is well defined which entry is given by the variable $x_{i_{1,1}}$. The issue would be if x matches more than one pattern but not all patterns. If x matches all patterns it means that $y = (a, a, ..., a)$ for some $a \in [c]$ and then the value of variable $x_{i_{1,1}}$ is obviously a. Now assume y matches the pattern of two of the rows k and l, but not all rows. Let these rows be of the form

$$k = (k_1, ..., k_M)$$

$$l = (l_1, ..., l_M)$$

where each l_j, k_j is some variable x_i for $i \in [k-1]$. Let $A \subset [M] : A \neq [M]$ be a nonempty set such that $y_i = y_j \; \forall i, j \in A$ and $y_i \neq y_j \forall i \in A$ and $j \in [M] \backslash A$. As y follows the pattern of k and l we must have that the set $\{x_1, ..., x_{k-1}\} = \{k_c : c \in A\} \sqcup \{k_c : c \in [M] \backslash A\}$ and that $\{x_1, ..., x_{k-1}\} = \{l_c : c \in A\} \sqcup \{l_c : c \in [M] \backslash A\}$. Note also that if $k \neq l$ all combinations of indexes (x_i, x_j) must exist in the columns of rows k and l above. This yields a contradiction as our assumptions yield that we only have columns of the form (x_i, x_j) if $i \in \{k_c : c \in A\}$, $j \in \{l_c : c \in A\}$ or $i \in \{k_c : c \in [M] \backslash A\}$, $j \in \{l_c : c \in [M] \backslash A\}$, meaning that no columns exist of the form (x_i, x_j) where $i \in \{k_c : c \in A\}$, $j \in \{l_c : c \in [M] \backslash A\}$ which must hold for $k \neq l$ and $A \neq [M]$ being nonempty.

The second nontrivial property to verify is that no edge of the form $(1, x_{i_{1,1}})$ is monochromatic, that is that two vertices represented by two of the rows $(x_{i_{j,1}}, x_{i_{j,2}}, ..., x_{i_{j,M}})$ have an edge between them. This is only the case if they differ in every coordinate and we make sure that this cannot be the case. Let us show that there is no edge between two arbitrary rows, i.e they do not differ in every column. As the columns of all the rows form all tuples in $\mathbf{R}_{k,k-1}$ the columns of our two arbitrary rows should list all possible ordered pairs of $\{x_1, ..., x_{k-1}\}$ including $(x_{i_{1,1}}, x_{i_{1,1}})$, meaning that any two rows colored with color $(1, x_{i_{1,1}})$ do not differ from each other in all coordinates.

Thus, $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})}(\mathbf{R}_{k,k-1}) \not\to \mathbf{H}_{k,2}$ implying that there is no gadget reduction from PCSP($\mathbf{R}_{k,k-1}, \mathbf{H}_{k,2}$) to PCSP($\mathbf{K}_c, \mathbf{K}_{2c}$). ☐

It follows from Theorem 3.6 that there is no gadget reduction PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) $\to$ PCSP($\mathbf{K}_c, \mathbf{K}_{2c+a}$) for any $k > q \geq 2$, $c \geq 3$ and $a \geq 0$ because by Section 2.4 we have the reductions PCSP($\mathbf{K}_c, \mathbf{K}_{2c+a}$) $\to$ PCSP($\mathbf{K}_c, \mathbf{K}_{2c}$) and PCSP($\mathbf{R}_{k,k-1}, \mathbf{H}_{k,2}$) $\to$ PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) which together with Theorem 3.6 contradict the existence of such a gadget reduction. This means that no NP-hardness results of rainbow-coloring PCSP's can imply new hardness results for graph coloring PCSP's using gadget reductions as PCSP($\mathbf{K}_c, \mathbf{K}_{2c}$) is known to be NP-hard for $c \geq 3$ as presented in Section 2.4.

3.3.2 PCSP($\mathbf{K}_l, \mathbf{K}_c$) to PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$)

To show that no NP-hardness results of graph coloring PCSP's cannot imply new hardness results for rainbow coloring PCSP's using gadget reductions requires a few steps. We begin with a theorem.

Theorem 3.7. *There is no gadget reduction from PCSP($\boldsymbol{K}_3, \boldsymbol{K}_c$) to PCSP($\boldsymbol{R}_{2q-3,q}, \boldsymbol{H}_{2q-3,2}$) for $c \geq 3$ and $q \geq 4$.*

Proof. By Corollary 2.31 it is enough to show that $\mathcal{F}_{\mathrm{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})}(\mathbf{K}_3) \not\to \mathbf{K}_c$. In $\mathcal{F}_{\mathrm{Pol}(\mathbf{R}_{2q-3,q})}(\mathbf{K}_3)$ we have the tuple (f_1, f_2) for $f_1, f_2 \in \mathrm{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})^{(3)}$ if and only if

$$f_1(x, y, z) = g(x, x, y, y, z, z)$$

$$f_2(x, y, z) = g(y, z, x, z, x, y)$$

for some some $g \in \text{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})^{(6)}$. If there is a tuple $(f_1, f_2) : f_1 = f_2$ in $\mathcal{F}_{\text{Pol}(\mathbf{R}_{2q-3,q},\mathbf{H}_{k,2})}(\mathbf{K}_3)$ then there exists no homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{R}_{2q-3,q},\mathbf{H}_{k,2})}(\mathbf{K}_3) \to \mathbf{K}_c$ because $(\phi(f_1), \phi(f_2)) = (\phi(f_1), \phi(f_1)) = (a, a)$ for some $a \in [c]$ is not a tuple in $\mathbf{K}_c$. We show that there exists an $f \in \text{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})^{(3)}$ such that (f, f) is a tuple in $\mathcal{F}_{\text{Pol}(\mathbf{R}_{2q-3,q},\mathbf{H}_{k,2})}(\mathbf{K}_3)$. There exists such an f if there exists a $g \in \text{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})^{(6)}$ such that

$$f(x, y, z) = g(x, x, y, y, z, z)$$

$$f(x, y, z) = g(y, z, x, z, x, y)$$

where we must have that

$$g(x, x, y, y, z, z) = g(y, z, x, z, x, y).$$

We show that such a $g : [q] \to [2]$ exists. We define g as follows

$$g(x_1, x_2, x_3, x_3, x_4, x_5, x_6) = \begin{cases} 1 & \text{if } x_1 = 1 \text{ or } x_3 = 1 \\ 2 & \text{else} \end{cases}$$

First we show that g is a valid polymorphism in $\text{Pol}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})^{(6)}$. For any $(2q-3)$ input strings $y_1, ..., y_{2q-3}$ such that each coordinate has all q values (i.e the i'th coordinate of all strings forms a tuple of $\mathbf{R}_{2q-3,q}$ for all i) there is at least one string y_i such that the first and third coordinate are not 1. This is the case because in each column at most $(2q-3) - (q-1) = q-2$ rows have the value 1. Because we have $(2q-3)$ rows and in each column all q colors need to exist yielding at least $q-1$ elements in each column that are not 1. The fact that for each column at most $q-2$ elements have the value 1 implies that at most $2(q-2) = 2q-4$ strings have value 1 in coordinate 1 or 3 yielding at least one string that does not have the value 1 in the first or third coordinate. Next it is obvious that at least one string has value 1 in the first or third coordinate as all columns including the first and third must contain all values. This means that for any given $2q-3$ rows of length 6 so that each column forms a tuple in $\mathbf{R}_{2q-3,q}$ we have that the rows are not all mapped to the same element in $\mathbf{H}_{k,2}$ meaning the mappings of the rows form a tuple in $\mathbf{H}_{k,2}$ allowing us to conclude that g is a valid polymorphism. It is also easy to see that g satisfies the requirement

$$g(x, x, y, y, z, z) = g(y, z, x, z, x, y)$$

because if $x_1 = 1$ or $x_3 = 1$ this means that $x = 1$ or $y = 1$ meaning that (x, x, y, y, z, z) and (y, z, x, z, x, y) are mapped to 1. Furthermore they are both mapped to 2 if $x_1 \neq 1$ and $x_3 \neq 1$ because then x nor y is 1.

We have now shown that $\mathcal{F}_{\text{Pol}(\mathbf{R}_{2q-3,q},\mathbf{H}_{k,2})}(\mathbf{K}_3) \not\to \mathbf{K}_c$ meaning that there is no gadget reduction from $\text{PCSP}(\mathbf{K}_3, \mathbf{K}_c)$ to $\text{PCSP}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})$ for $q \geq 4$, $c \geq 3$. □

It follows from Theorem 3.7 that there is no gadget reduction $\text{PCSP}(\mathbf{K}_{c_1}, \mathbf{K}_{c_2}) \to \text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ for any k such that $2q-3 \geq k > q \geq 4$, $c_2 \geq c_1 \geq 3$ because by Section 2.4 we have the reductions $\text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2}) \to \text{PCSP}(\mathbf{R}_{2q-3,q}, \mathbf{H}_{k,2})$ and $\text{PCSP}(\mathbf{K}_3, \mathbf{K}_c) \to \text{PCSP}(\mathbf{K}_{c_1}, \mathbf{K}_{c_2})$ for some $c \geq 3$ which together with Theorem 3.7 contradict the existence of such a gadget reduction. We call this **Fact 1**. Now we want to show that there exists no graph coloring PCSP can yield any new NP-hardness results for rainbow coloring PCSP's with a gadget reduction. We list **Fact 1** and two other facts that consist of the known NP-hardness results for rainbow coloring presented in Section 2.4:

- **Fact 1:** No gadget reduction exists from $\text{PCSP}(\mathbf{K}_3, \mathbf{K}_c)$ to $\text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ for $2q-3 \geq k \geq q \geq 4$, $c \geq 3$.
- **Fact 2:** $\text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ is NP-hard for $k \geq \lfloor \frac{3q}{2} \rfloor$ and $q \geq 2$. This follows from the result that $\text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ is NP-hard if for some $t \geq 1$ and $d \geq 2$ we have that $k \geq td + \lfloor \frac{d}{2} \rfloor$ and $q \leq t(d-1)+1$ and $k > q \geq 2$ ([1] Theorem 1.3) by setting $t = 1$ and $d = q$.
- **Fact 3** $\text{PCSP}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ is NP-hard for $k > q$ if $q \leq 5$ (Theorem 2 [14]).

Now we show that no graph coloring PCSP can yield any new NP-hardness results for rainbow coloring PCSP's with a gadget reduction. It is enough to do so for two cases:

- **Case 1**: $q \in \{2,3,4,5\}$. By **Fact 3** we already know that PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) for $k > q$ is NP-hard
- **Case 2**: $q \geq 6$. By **Fact 2** we know that PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) for $k \geq \lfloor \frac{3q}{2} \rfloor$ is NP-hard. This means that the only new potential hardness result for a fixed $q \geq 6$ is PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$) for $k < \lfloor \frac{3q}{2} \rfloor \leq 2q - 3$. However, by **Fact 1** there exists no gadget reduction from a graph coloring PCSP to such a PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$)

We illustrate this in Figure 11. With these cases we can conclude that no graph coloring PCSP can prove any new NP-hardness results for rainbow coloring PCSP's with a gadget reduction.

PCSP($\mathbf{R}_{k,q}, \mathbf{H}_{k,2}$)

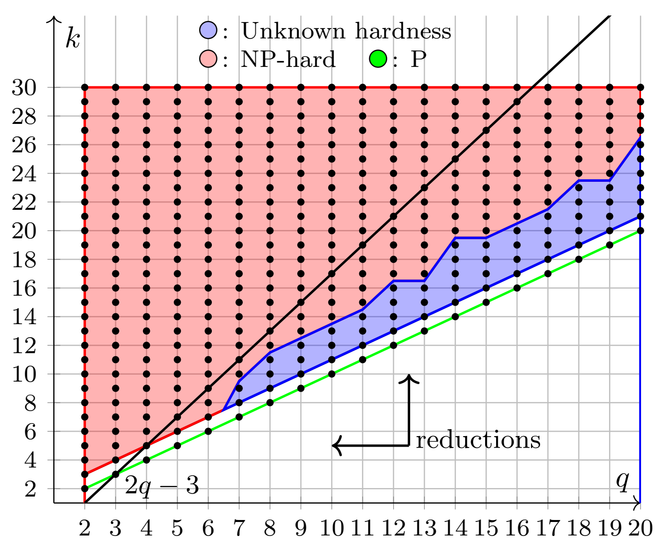

Figure 11: There is no gadget reduction from a graph coloring PCSP to any PCSP below the line $2q - 3$. This illustrates that no graph coloring PCSP can imply any new NP-hardness results for rainbow coloring PCSP's with a gadget reduction.

3.4 LO-coloring

3.4.1 PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) to PCSP($\mathbf{K}_c, \mathbf{K}_q$)

We begin with a theorem.

Theorem 3.8. *There is no gadget reduction from PCSP(*$\boldsymbol{LO}_2^r, \boldsymbol{LO}_k^r$*)) to PCSP(*$\boldsymbol{K}_c, \boldsymbol{K}_{2c}$*) for any* $k \geq 2$ *and* $c, r \geq 3$*.*

Proof. We do this first for the case $r = 3$ and then generalize to all $r \geq 4$. By Corollary 2.31 it is enough to show that $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})}(\mathbf{LO}_2^3) \not\to \mathbf{LO}_k^3$. This statement is proven by showing that the relation cannot be preserved in a mapping from $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})}(\mathbf{LO}_2^3)$ to $\mathbf{LO}_k^3$. In $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})}(\mathbf{LO}_2^3)$ we have the tuple (f_1, f_2, f_3) for $f_1, f_2, f_3 \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^{(2)}$ if

$$f_1(x, y) = g(y, x, x)$$

$$f_2(x,y) = g(x,y,x)$$

$$f_3(x,y) = g(x,x,y)$$

for some $g \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^{(3)}$. The issue that arises is that we in $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{LO}_2^3)$ have the tuple (f_1, f_2, f_3) for $f_1 = f_2 = f_3$, this cannot be preserved in $\mathbf{LO}_k^3$ for any mapping $\phi : \mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{LO}_2^3) \to \mathbf{LO}_k^3$, because $(\phi(f_1), \phi(f_2), \phi(f_3)) = (\phi(f_1), \phi(f_1), \phi(f_1))$ is not a tuple in $\mathbf{LO}_k^3$ as it does not have a unique maximum. This is the case only if there exists a $g \in \mathrm{Pol}(\mathbf{K}_c, \mathbf{K}_{2c})^{(3)}$ such that

$$f_1(x,y) = g(y,x,x)$$

$$f_1(x,y) = g(x,y,x)$$

$$f_1(x,y) = g(x,x,y),$$

that is

$$g(y,x,x) = g(x,y,x) = g(x,x,y).$$

By showing that such a g exists we prove that there does not exist a homomorphism $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{LO}_2^3) \not\to \mathbf{LO}_k^3$. Such a g is a $2c$-coloring of the vertices $[c]^3$. We denote our $2c$ colors by $(1,1),(1,2),...,(1,c)$ and $(2,1),(2,2),...,(2,c)$. We define g in the following way, if input $[c]^3$ has a $b \in [c]$ in all but one entry we give it the color $(1,b)$, otherwise it receives the color $(2,x_1)$ where x_1 is the color of the first entry.

$$g(x_1,x_2,x_3) = \begin{cases} (1,b) & \text{if } x_i \neq b \text{ for at most one } i \in [3] \text{ for some } b \in [c] \\ (2,x_1) & \text{else} \end{cases}$$

What this does is assign the color $(1,x)$ to vertices of the form

$$(y,x,x)$$

$$(x,y,x)$$

$$(x,x,y)$$

$$(x,x,x)$$

for any $x,y \in [c] : x \neq y$. This causes no issue as there exist no edge between these vertices for a fixed x. So all vertices of this form can be colored with the c colors $(1,1),(1,2),...,(1,c)$ while following the requirement that

$$g(y,x,x) = g(x,y,x) = g(x,x,y)$$

for any $x,y \in [c]$. The rest of the vertices that are not of this form are colored with c separate colors using a dictator coloring. We have now defined a valid $2c$-coloring g of the vertices $[c]^3$ with the requirement that

$$g(y,x,x) = g(x,y,x) = g(x,x,y)$$

for any $x,y \in [c]$. This proves that $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{LO}_2^3) \not\to \mathbf{LO}_k^3$ proving that absence of a

a gadget reduction from There is no gadget reduction from $\mathrm{PCSP}(\mathbf{LO}_2^3, \mathbf{LO}_k^3)$) to $\mathrm{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c})$. This result can be easily generalised to $\mathcal{F}_{\mathrm{Pol}(\mathbf{K}_c,\mathbf{K}_{2c})}(\mathbf{LO}_2^r) \not\to \mathbf{LO}_k^r$ for any $r \geq 3$. All that needs to be done is adjust the function g to be a $2c$-coloring of $[c]^r$ which follows all its requirements by being defined:

$$g(x_1,x_2,...,x_r) = \begin{cases} (1,b) & \text{if } x_i \neq b \text{ for at most one } i \in [r] \text{ for some } b \in [c] \\ (2,x_1) & \text{else} \end{cases}$$

This means that we now know that there does not exist any gadget reduction from $\mathrm{PCSP}(\mathbf{LO}_2^r, \mathbf{LO}_k^r)$) to $\mathrm{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c})$ for any $k \geq 2$ and $c, r \geq 3$.

□

It follows from Theorem 3.8 that there is no gadget reduction $\text{PCSP}(\mathbf{LO}_l^r, \mathbf{LO}_k^r) \to \text{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c+a})$ for any $k \geq l \geq 2$ and $c, r \geq 3$ and $a \geq 0$ because by Section 2.4 we have the reductions $\text{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c+a}) \to \text{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c})$ and $\text{PCSP}(\mathbf{LO}_2^r, \mathbf{LO}_k^r) \to \text{PCSP}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)$ which together with Theorem 3.8 contradicts the existence of such a gadget reduction. This means that no NP-hardness results of LO-coloring PCSP's can imply new hardness results for graph coloring PCSP's using gadget reductions as $\text{PCSP}(\mathbf{K}_c, \mathbf{K}_{2c})$ is known to be NP-hard for $c \geq 3$ as presented in Section 2.4.

3.4.2 $\text{PCSP}(\mathbf{K}_c, \mathbf{K}_q)$ to $\text{PCSP}(\mathbf{LO}_2^r, \mathbf{LO}_k^r)$

Theorem 3.9. *There is no gadget reduction from PCSP($\boldsymbol{K}_3, \boldsymbol{K}_q$) to PCSP($\boldsymbol{LO}_2^r, \boldsymbol{LO}_3^r$) for any $q, r \geq 3$.*

Proof. We first show this for the case $r = 6$ and then generalize this to all $r \geq 3$. By Corollary 2.31 it is enough to show that there exists no homomorphism $\mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)}(\mathbf{K}_3) \to \mathbf{K}_q$. In $\mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^{(6)})}(\mathbf{K}_3)$ we have the tuple (f_1, f_2) if and only if there exists a $g \in \text{Pol}(\mathbf{LO}_2^3, \mathbf{LO}_3^3)^{(6)}$ such that

$$f_1(x, y, z) = g(x, y, z, x, y, z)$$

$$f_2(x, y, z) = g(z, x, y, y, z, x).$$

The issue is that the tuple $(f_1, f_2) : f_1 = f_2$ exists in $\mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)}(\mathbf{K}_3)$ for some $f_1 \in \text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)^{(3)}$. This means that no mapping $\phi : \mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)}(\mathbf{K}_3) \to \mathbf{K}_q$ can preserve the relation as $(\phi(f_1), \phi(f_2)) = (\phi(f_1), \phi(f_1))$ is not a tuple in $\mathbf{K}_q$ for any mapping ϕ as $\phi(f_1) = \phi(f_1)$ for all mappings $\phi : \mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)}(\mathbf{K}_3) \to \mathbf{K}_q$. What we need to prove is that such a tuple indeed exists in the free structure. To show that (f_1, f_1) is a tuple in $\mathcal{F}_{\text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)}(\mathbf{K}_3)$ we must show that there exists a $g \in \text{Pol}(\mathbf{LO}_2^6, \mathbf{LO}_3^6)^{(6)}$ such that

$$f_1(x, y, z) = g(x, y, z, x, y, z)$$

$$f_1(x, y, z) = g(z, x, y, y, z, x)$$

i.e,

$$g(x, y, z, x, y, z) = g(z, x, y, y, z, x)$$

for all $x, y, z \in [2]$. This is the same as defining a 3-LO-coloring of $\mathbf{LO}_2^6$ with the requirement that vertices (x, y, z, x, y, z) and (z, x, y, y, z, x) must be given the same color for any $x, y, z \in [2]$. Below is such a coloring g.

$$\begin{cases} g(2,2,2,2,2,2)=3, \\ g(2,2,2,2,2,1)=1, \\ g(2,2,2,2,1,2)=3, \\ g(2,2,2,1,2,2)=3, \\ g(2,2,1,2,2,2)=3, \\ g(2,1,2,2,2,2)=3, \\ g(1,2,2,2,2,2)=3, \\ g(1,1,2,2,2,2)=3, \\ g(1,2,1,2,2,2)=3, \\ g(1,2,2,1,2,2)=2, \\ g(1,2,2,2,1,2)=2, \\ g(1,2,2,2,2,1)=1, \\ g(2,1,1,2,2,2)=3, \\ g(2,1,2,1,2,2)=3, \\ g(2,1,2,2,1,2)=3, \\ g(2,1,2,2,2,1)=2, \\ g(2,2,1,1,2,2)=3, \\ g(2,2,1,2,1,2)=3, \\ g(2,2,1,2,2,1)=2, \\ g(2,2,2,1,1,2)=3, \\ g(2,2,2,1,2,1)=1, \\ g(2,2,2,2,1,1)=1, \end{cases} \quad \begin{cases} g(2,2,2,1,1,1)=1, \\ g(2,2,1,2,1,1)=2, \\ g(2,2,1,1,2,1)=2, \\ g(2,2,1,1,1,2)=3, \\ g(2,1,2,2,1,1)=2, \\ g(2,1,2,1,2,1)=2, \\ g(2,1,2,1,1,2)=3, \\ g(1,2,2,2,1,1)=1, \\ g(1,2,2,1,2,1)=1, \\ g(1,2,2,1,1,2)=3, \\ g(1,1,1,2,2,2)=3, \\ g(2,1,1,2,2,1)=1, \\ g(1,2,1,2,2,1)=2, \\ g(1,1,2,2,2,1)=2, \\ g(2,1,1,2,1,2)=3, \\ g(1,2,1,2,1,2)=3, \\ g(1,1,2,2,1,2)=3, \\ g(2,1,1,1,2,2)=3, \\ g(1,2,1,1,2,2)=3, \\ g(1,1,2,1,2,2)=3, \\ g(2,2,1,1,1,1)=2, \end{cases} \quad \begin{cases} g(2,1,2,1,1,1)=2, \\ g(2,1,1,2,1,1)=1, \\ g(2,1,1,1,2,1)=1, \\ g(2,1,1,1,1,2)=3, \\ g(1,2,2,1,1,1)=1, \\ g(1,2,1,2,1,1)=2, \\ g(1,2,1,1,2,1)=2, \\ g(1,2,1,1,1,2)=1, \\ g(1,1,2,2,1,1)=2, \\ g(1,1,2,1,2,1)=2, \\ g(1,1,2,1,1,2)=1, \\ g(1,1,1,2,2,1)=1, \\ g(1,1,1,2,1,2)=3, \\ g(1,1,1,1,2,2)=3, \\ g(2,1,1,1,1,1)=1, \\ g(1,2,1,1,1,1)=2, \\ g(1,1,2,1,1,1)=2, \\ g(1,1,1,2,1,1)=1, \\ g(1,1,1,1,2,1)=1, \\ g(1,1,1,1,1,2)=3, \\ g(1,1,1,1,1,1)=1. \end{cases}$$

We have that g is a valid 3-LO-coloring if

$$\begin{gathered}(g(x_{1,1},x_{1,2},...,x_{1,6}), \\ g(x_{2,1},x_{2,2},...,x_{2,6}), \\ g(x_{3,1},x_{3,2},...,x_{3,6}), \\ g(x_{4,1},x_{4,2},...,x_{4,6}), \\ g(x_{5,1},x_{5,2},...,x_{5,6}), \\ g(x_{6,1},x_{6,2},...,x_{6,6}))\end{gathered}$$

is a tuple in $\mathbf{LO}_3^6$, meaning it has a unique maximum for any combination of vertices such that in each column there exists a unique maximum, that is each column is a tuple in $\mathbf{LO}_2^6$, this means that each column has one 2 and the rest are 1. That this holds for g can be verified manually. Note that we have given the maximum color 3 to almost all vertices that have a 2 in the 6th element. No other vertices are given the color 3. This means that if the vertex $(x_{i,1},x_{i,2},...,x_{i,6})$ above with $x_{i,6}=2$ is given the color 3 we know that we have a unique maximum. The challenging part to check is if the vertex $(x_{i,1},x_{i,2},...,x_{i,6})$ with $x_{i,6}=2$ is not given the color 3, for these we need to check all combinations of 6 vertices creating 6 columns of tuples in $\mathbf{LO}_2^6$ and make sure they have a unique maximum. There are quite a few combinations to check but it can be done quite simply by hand. One of course also needs to check that the requirement $g(x,y,z,x,y,z)=g(z,x,y,y,z,x)$ holds. Even though this can be checked manually the risk of error is significant so a python program has also been written which takes the coloring and checks that it is a valid 3-LO-coloring of $(\mathbf{LO}_2^6)^6$, this code can be found in the Appendix Section B.

We can now conclude $\mathcal{F}_{\mathrm{Pol}(\mathbf{LO}_2^6,\mathbf{LO}_3^6)}(\mathbf{K}_3)\not\to\mathbf{K}_q$. Next we generalise this to $\mathcal{F}_{\mathrm{Pol}(\mathbf{LO}_2^r,\mathbf{LO}_3^r)}(\mathbf{K}_3)\not\to\mathbf{K}_q$ for any $r\geq 3$. The exact same coloring g can be used to show this but it is now instead defined from $(\mathbf{LO}_2^r)^6\to\mathbf{LO}_3^r$. For $r>6$ all but at most 6 rows is $(1,1,1,1,1,1)$ and we are as previously shown able to give a unique maximal color to one of those 6 rows as all rows $(1,1,1,1,1,1)$ are given the minimal color 1. Next if $r = 3, 4, 5$ we already know that g works as these cases are equivalent to when $r = 6$ given that at least 3, 2 or 1 row respectively is $(1, 1, 1, 1, 1, 1)$.

This means that we indeed have that $\mathcal{F}_{\text{Pol}(\mathbf{LO}_2^r, \mathbf{LO}_3^r)}(\mathbf{K}_3) \not\to \mathbf{K}_q$ for any $r \geq 3$. This implies that no gadget reduction exists from PCSP$(\mathbf{K}_3, \mathbf{K}_q)$ to PCSP$(\mathbf{LO}_2^r, \mathbf{LO}_3^r)$ for $q, r \geq 3$. ☐

It follows from Theorem 3.9 that there is no gadget reduction PCSP$(\mathbf{K}_l, \mathbf{K}_q) \to$ PCSP$(\mathbf{LO}_2^r, \mathbf{LO}_k^r)$ for any $q \geq l \geq 3$, $r \geq 3$ and $k > 2$ because by Section 2.4 we have the reductions PCSP$(\mathbf{K}_3, \mathbf{K}_c) \to$ PCSP$(\mathbf{K}_l, \mathbf{K}_q)$ for some $c \geq 3$ and PCSP$(\mathbf{LO}_2^r, \mathbf{LO}_k^r) \to$ PCSP$(\mathbf{LO}_2^r, \mathbf{LO}_3^r)$ which together with Theorem 3.9 contradicts the existence of such a gadget reduction. This means that no NP-hardness results of graph coloring PCSP's can imply new hardness results for PCSP$(\mathbf{LO}_2^r, \mathbf{LO}_k^r)$ using gadget reductions for $k, r \geq 3$.

3.4.3 PCSP$(\mathbf{K}_l, \mathbf{K}_k)$ yields new NP-hardness results for PCSP$(\mathbf{LO}_l^r, \mathbf{LO}_k^r)$

Theorem 3.10. *There exists a gadget reduction from PCSP$(\boldsymbol{K}_l, \boldsymbol{K}_k)$ to PCSP$(\boldsymbol{LO}_l^r, \boldsymbol{LO}_k^r)$ for $k \geq l \geq 3$ and $r \geq 4$*

Proof. This can be shown with the following reduction:

```
function PCSP(K_l, K_k)(I):
    For every edge (x, y), add a new node z to our set and replace the edge (x, y) the hyperedge
  (x, y, z, ..., z) of arity r
    return PCSP(LO_l^r, LO_k^r)(I)
```

To show that this is valid we must first show that YES instances are mapped to YES instances. This is quite simple, if $\mathbf{I}$ is l-colorable then for the l-LO-coloring of the updated $\mathbf{I}$ we can color all the new nodes z with the color 1 and color all other vertices the same way as for the graph l-coloring, for each edge $(x, y, z, ..., z)$ we then have that x and y are given different colors making the larger one a unique maximum as z's are mapped to 1. Next we show that NO-instances are mapped to NO-instances. Assume that $\mathbf{I}$ is not k-colorable, then for any k-coloring of the updated I which is a hypergraph we have that for some edge $(x, y, z, ...z)$ that x and y are given the same color causing the edge not to have a unique maximum as z exists in the edge at least twice yielding that the coloring is not a valid k-LO-coloring meaning that the updated $\mathbf{I}$ is not k-LO-colorable. ☐

This reduction was found after proving its existence with the algebraic approach. We present this proof.

Proof. By Corollary 2.31 to it is enough to show that there exists a homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)}(\mathbf{K}_l) \to \mathbf{K}_k$. In the free structure we have the tuple (f_1, f_2) for $f_1, f_2 \in \text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)^{(l)}$ if there exists a $g \in \text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)^{(M)}$ such that

$$f_1(x_1, ..., x_l) = g(x_1, x_1, ..., x_1, x_2, x_2, ..., x_l, x_l)$$
$$f_2(x_1, ..., x_l) = g(x_2, x_3, ..., x_l, x_1, x_3, ..., x_{l-2}, x_{l-1})$$

where each column of indexes forms one of the tuples in $\mathbf{K}_l$, each tuple appears in exactly one column. This means that $M = l(l-1) = l^2 - l$. Because all l elements appear in the top row $l-1$ times for each of the other elements it can have underneath it. Let us now define $\phi : \mathcal{F}_{\text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)}(\mathbf{K}_l) \to \mathbf{K}_k$, let $\phi(f) = f(1, 2, ..., l)$ for all $f \in \text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)^{(l)}$. All that is needed for ϕ to be a homomorphism is that it is a well defined map and for any tuple (f_1, f_2) in the free structure f_1 and f_2 are mapped to different elements in order to preserve the relation. The mapping is well defined as f sends input to the set $LO_k^r = [k] = K_k$. Next we need to show that no two related elements can be sent to the same element in $\mathbf{K}_k$ as this would yield a contradiction. Let (f_1, f_2) be a tuple in $\mathcal{F}_{\text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)}(\mathbf{K}_l)$. We have that

$$f_1(1, 2, ..., l) = g(1, 1, ..., 1, 2, 2, ..., l, l)$$

$$f_2(1, 2, ..., l) = g(2, 3, ..., l, 1, 3, ..., l-2, l-1)$$

Next note that the following is a tuple in $(\mathbf{LO}_l^r)^M$ where the first two rows are identical to the ones above:

$$\begin{bmatrix} ((1 & 1 & \dots & 1 & 2 & 2 & \dots & l & l), \\ (2 & 3 & \dots & l & 1 & 3 & \dots & l-2 & l-1), \\ (1 & 1 & \dots & 1 & 1 & 1 & \dots & 1 & 1), \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ (1 & 1 & \dots & 1 & 1 & 1 & \dots & 1 & 1)) \end{bmatrix}$$

In total we have r rows. This is a tuple in $(\mathbf{LO}_l^r)^M$ as the columns of the first two rows are tuples in $\mathbf{K}_l$ meaning that for each column the two first rows are different yielding that one of the elements is greater then the other and is also greater than 1. This means that each column has a unique maximum and is a tuple in $\mathbf{LO}_l^r$ meaning that the above is a tuple in $(\mathbf{LO}_l^r)^M$. This means that the set of images of the rows under g has a unique maximum, the unique maximum cannot be the image of $(1, 1, ..., 1)$ as several rows of the form $(1, 1, ..., 1)$ are mapped there meaning that it cannot be a unique maximum. This means that one of the two first rows are sent to the unique maximum implying that they are mapped to different colors. This shows that two related elements f_1, f_2 in our free structure cannot be mapped to the same element under the homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)}(\mathbf{K}_l) \to \mathbf{K}_k$ implying that $(\phi(f_1), \phi(f_2))$ is a tuple in $\mathbf{K}_k$ for any related f_1, f_2 implying that ϕ is a valid homomorphism allowing us to conclude the existence of a gadget reduction.

□

The theorem gives us the following corollary yielding a great deal of new NP-hardness results for LO-coloring PCSP's

Corollary 3.11. *PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) is NP-hard if $2l - 1 \geq k \geq l$ for $l \geq 3$ and if $(\binom{l}{\lfloor l/2 \rfloor} - 1) \geq k \geq l$ for $l \geq 4$.*

Proof. Theorem 3.10 tells us that if PCSP($\mathbf{K}_l, \mathbf{K}_k$) is NP-hard for $k \geq l \geq 3$ then so is PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) for $r \geq 4$. Then the Corollary is proven by PCSP($\mathbf{K}_l, \mathbf{K}_k$) being NP-hard if $2l - 1 \geq k \geq l$ for $l \geq 3$ ([5]) and if $(\binom{l}{\lfloor l/2 \rfloor} - 1) \geq k \geq l$ for $l \geq 4$ ([21]). □

We illustrate the new NP-hard results in Figure 12 and we can see that we indeed have found a great deal of new NP-hardness results

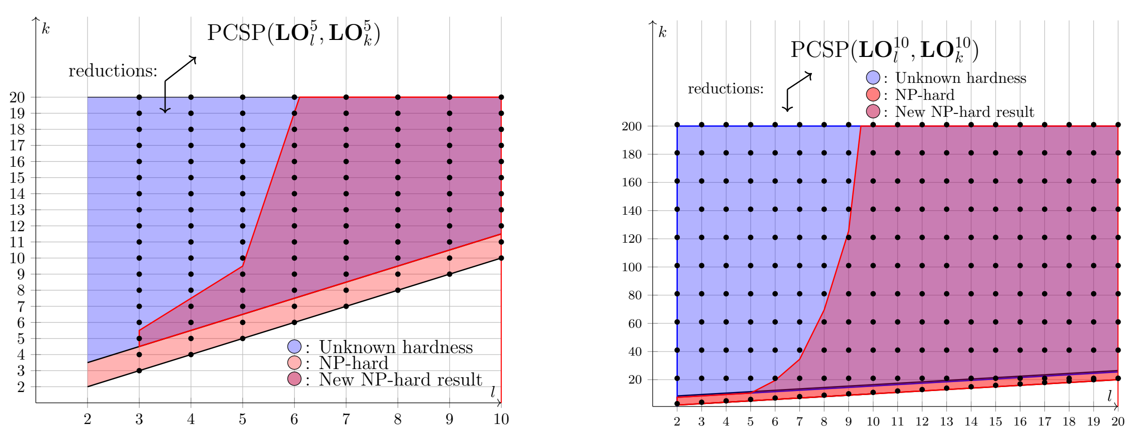

Figure 12: Illustration of complexity results and trivial reductions for $r = 5$ and $r = 10$

4 Conclusion

For graph coloring PCSP's we generalized a previous result and showed that there exists no gadget reduction from PCSP($\mathbf{K}_k, \mathbf{K}_q$) to PCSP($\mathbf{K}_{k'}, \mathbf{K}_{q'}$) for $k' < k$ and $q' \geq 2k'$ meaning that new hardness results for PCSP($\mathbf{K}_{k'}, \mathbf{K}_{q'}$) cannot be proved by a gadget reduction from PCSP($\mathbf{K}_k, \mathbf{K}_q$). We also proved the absence of gadget reductions PCSP($\mathbf{K}_3, \mathbf{K}_5$) $\rightarrow$ PCSP($\mathbf{K}_4, \mathbf{K}_8$) and PCSP($\mathbf{K}_3, \mathbf{K}_6$) $\rightarrow$ PCSP($\mathbf{K}_4, \mathbf{K}_{36}$). An interesting open question if there exists PCSP($\mathbf{K}_3, \mathbf{K}_6$) $\rightarrow$ PCSP($\mathbf{K}_4, \mathbf{K}_8$). Or if one at least could decrease the number 36 to something smaller in the absence of the gadget reduction PCSP($\mathbf{K}_3, \mathbf{K}_6$) $\rightarrow$ PCSP($\mathbf{K}_4, \mathbf{K}_{36}$).

The following results we illustrate in Figure 13. We have shown that now graph coloring and rainbow coloring PCSP's can prove any new NP-hardness results of each other by a gadget reduction Furthermore we showed that LO-coloring PCSP's cannot by a gadget reduction yield any new hardness results for graph coloring PCSP's and that graph coloring PCSP's cannot by a gadget reduction yield any hardness results for PCSP($\mathbf{LO}_2^r, \mathbf{LO}_k^r$) for $r, k \geq 3$. Finally we prove the existence of a gadget reduction PCSP($\mathbf{K}_l, \mathbf{K}_k$) to PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) for $k \geq l \geq 3$ and $r \geq 4$ resulting in a great deal of new hardness results for LO-coloring PCSP's. These new results are that for $r \geq 4$ PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) is NP-hard if $2l - 1 \geq k \geq l$ for $l \geq 3$ and if $(\binom{l}{\lfloor l/2 \rfloor} - 1) \geq k \geq l$ for $l \geq 4$. An interesting question that we have not investigated in this thesis is whether there exists gadget reductions between rainbow coloring and LO-coloring PCSP's.

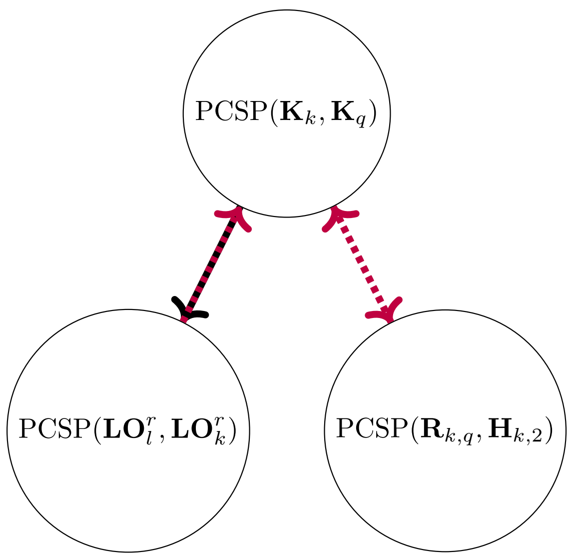

Figure 13: The dashed purple arrows symbolise the absence of a gadget reductions that can yield new complexity results. The black arrow symbolises the existence of PCSP($\mathbf{K}_l, \mathbf{K}_k$) to PCSP($\mathbf{LO}_l^r, \mathbf{LO}_k^r$) for $k \geq l \geq 3$ and $r \geq 4$.

References

[1] Per Austrin, Amey Bhangale, and Aditya Potukuchi. "Improved Inapproximability of Rainbow Coloring". In: *Proceedings of the 31st ACM-SIAM Symposium on Discrete Algorithms (SODA'20). 1479–1495.*https://epubs.siam.org/doi/10.1137/1.9781611975994.90 arXiv: arxiv.org/abs/1810.02784 (2020).

[2] Per Austrin, Vanketsan Guruswami, and Johan Håstad. “(2 + ε)-sat is NP-hard.” In: *SIAM J. Comput. 46, 5 (2017), 1554–1573. DOI:*https://doi.org/10.1137/15M1006507 (2017).

[3] Libor Barto, Diego Battistelli, and Kevin M. Berg. “Symmetric Promise Constraint Satisfaction Problems: Beyond the Boolean Case.” In: *In Proc. 38th International Symposium on Theor- etical Aspects of Computer Science (STACS’21), volume 187 of LIPIcs, pages 10:1–10:16, 2021.DOI:* 10.4230/LIPIcs.STACS.2021.10 (2021).

[4] Libor Barto, Andrei Krokhin, and Ross Willard. “Polymorphisms, and how to use them.” In: *In ”The Constraint Satisfaction Problem: Complexity and Approximability”, Andrei Krokhin and Stanislav Živný (Eds.). Dagstuhl Follow-Ups, Vol. 7. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 1–44. DOI:*https://doi.org/10.4230/DFU.Vol7.15301.1 (2017).

[5] Libor Barto et al. “Algebraic Approach to Promise Constraint Satisfaction”. In: *J. ACM, 68(4):28:1–28:66.* https://doi.org/10.1145/3457606 (2021).

[6] Joshua Brakensiak and Venkatesan Guruswami. “The quest for strong inapproximability results with perfect completeness.” In: *ACM Trans. Algorithms, 17(3):27:1–27:35, 2021. DOI:* 10.1145/3459668 (2021).

[7] Joshua Brakensiek and Vanketsan Guruswami. “Promise constraint satisfaction: Algebraic structure and a symmetric Boolean dichotomy.” In: *ECCC Report No. 183 (2016).* https://eccc.weizmann.ac.il/report/201 (2016).

[8] Joshua Brakensiek and Vanketsan Guruswami. “Promise constraint satisfaction: Structure theory and a symmetric Boolean dichotomy.” In: *In Proceedings of the 29th ACM-SIAM Symposium on Discrete Algorithms (SODA’18). Society for Industrial and Applied Mathematics, Philadelphia, PA, 1782–1801. DOI:* https://doi.org/10.1137/1.9781611975031.117 (2018).

[9] Andrei Bulatov and Peter Jeavons. “Algebraic Structures in Combinatorial Problems.” In: *Technical Report. Technische Universität Dresden.* (2001).

[10] Andrei Bulatov and Peter Jeavons. “Classifying the complexity of constraints using finite algebras.” In: *SIAM J. Comput. 34, 3 (Mar. 2005), 720–742. DOI:*https://doi.org/10.1137/S0097539700376676 (2005).

[11] Andrei A. Bulatov. “A dichotomy theorem for nonuniform CSPs.” In: *In Proceedings of the IEEE 58th Symposium on Foundations of Computer Science (FOCS’17). 319–330.* https://doi.org/10.1109/FOCS.2017.37 (2017).

[12] Tomás Feder and Moshe Y. Vardi. “The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory.” In: *SIAM J. Comput. 28, 1 (Feb. 1998), 57–104. DOI:* https://doi.org/10.1137/S0097539794266766 (2018).

[13] Venkatesan Guruswami and Euiwong Lee. “Strong inapproximability results on balanced rainbow colorable hypergraphs.” In: *Combinatorica, 38(3):547–599, 2018. DOI:* 10.1007/s00493-016-3383-0 (2018).

[14] Venkatesan Guruswami and Sai Sandeep. “Rainbow coloring hardness via low sensitivity polymorphisms.” In: *SIAM J. Discret. Math, 34(1):520–537, 2020. DOI:*10.1137/19M127731X ().

[15] Peter Jeavons. “On the algebraic structure of combinatorial problems.” In: *J. ACM 67, 5 (Aug. 2020), 30:1–30:78*https://doi.org/10.1145/3402029 (1998).

[16] Peter Jeavons, David Cohen, and Marc Gyssens. “Closure properties of constraints”. In: *J. ACM 44, 4 (July 1997), 527–548. DOI:* https://doi.org/10.1145/263867.263489 (1997).

[17] Andrei Krokhin and Stanislav Živný (Eds.) *The Constraint Satisfaction Problem: Complexity and Approximability. Dagstuhl Follow-Ups, Vol. 7. Schloss Dagstuhl – Leibniz-Zentrum für Informatik* https://drops.dagstuhl.de/entities/document/10.4230/DFU.Vol7.15301. 2017.

[18] Richard E. Ladner. “On the structure of polynomial time reducibility.” In: *JACM, vol. 22, pp. 155–171, Jan. 1975. DOI:* https://dl.acm.org/doi/10.1145/321864.321877 (1975).

[19] Tamio-Vesa Nakajima and Stanislav Živný. "Linearly ordered colourings of hypergraphs". In: *ACM Transactions on Computation Theory, Volume 14, Issue 3-4.* https://dl.acm.org/doi/10.1145/3570909 (2023).

[20] Thomas J. Schaefer. "The complexity of satisfiability problems." In: *Proc. 10th ACM Symp. on Theory of Computing, 216-226.* https://dl.acm.org/doi/10.1145/800133.804350 (1978).

[21] Marcin Wrochna and Stanislav Živný. "Improved hardness for H-colourings of G-colourable graphs." In: *In Proceedings of the 14th ACM-SIAM Symposium on Discrete Algorithms (SODA'20). 1426–1435. DOI:* https://doi.org/10.1137/1.9781611975994.86 arXiv: 1907.00872 (2020).

[22] Dmitriy Zhuk. "A proof of CSP dichotomy conjecture." In: *In Proceedings of the IEEE 58th Symposium on Foundations of Computer Science (FOCS'17). 331–342.* https://doi.org/10.1109/FOCS.2017.38 (2017).

[23] Dmitriy Zhuk. "A proof of CSP dichotomy conjecture." In: *J. ACM 67, 5 (Aug. 2020), 30:1–30:78* https://doi.org/10.1145/3402029 (2020).

5 Appendix

A Proofs of minion homomorphisms existing

A.0.1 $\mathbf{Pol}(\mathbf{K}_{k'}, \mathbf{K}_{q'}) \to \mathbf{Pol}(\mathbf{K}_k, \mathbf{K}_q)$ for $3 \leq k \leq k' \leq q' \leq q$

We have from theorem 2.30 that $\text{Pol}(\mathbf{K}_{k'}, \mathbf{K}_{q'}) \to \text{Pol}(\mathbf{K}_k, \mathbf{K}_q)$ for $3 \leq k \leq k' \leq q' \leq q$ if and only if $\mathcal{F}_{\text{Pol}(\mathbf{K}_{k'}, \mathbf{K}_{q'})}(\mathbf{K}_k) \to \mathbf{K}_q$. We show that this homomorphism exists for the specific case $\mathcal{F}_{\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)}(\mathbf{K}_3) \to \mathbf{K}_6$ as it is easier to relate to and follow than a general proof. From this proof it is then easy to see how it can be generalised to any $3 \leq k \leq k' \leq q' \leq q$.

In $\mathcal{F}_{\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)}(\mathbf{K}_3)$ we have the tuple (f_1, f_2) where $f_1, f_2 \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^(3)$ if and only if

$$f_1(x, y, z) = g(x, x, y, y, z, z)$$

$$f_2(x, y, z) = g(y, z, x, z, x, y)$$

For some $g \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^{(6)}$ where the variables in the columns form all tuples in $\mathbf{K}_k$. The desired homomorphism $\mathcal{F}_{\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)}(\mathbf{K}_3) \to \mathbf{K}_6$ can be defined by $f \to f(1, 2, 3)\ \forall f \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^{(3)}$. This is a well-defined mapping to $\mathbf{K}_6$ as f maps elements in $[4]^3$ to $[5] \subset [6] = K_6$. The mapping also preserves the relation as two related elements f_1 and f_2 are mapped to $g(1, 1, 2, 2, 3, 3)$ and $g(2, 3, 1, 3, 1, 2)$ respectively. Note that $(1, 1, 2, 2, 3, 3)$ and $(2, 3, 1, 3, 1, 2)$ differ in every entry meaning that $((1, 1, 2, 2, 3, 3), (2, 3, 1, 3, 1, 2))$ is a tuple in $\mathbf{K}_4^6$. As $g \in \text{Pol}(\mathbf{K}_4, \mathbf{K}_5)^{(6)}$ we have that $g(1, 1, 2, 2, 3, 3) \neq g(2, 3, 1, 3, 1, 2)$ implying that $(g(1, 1, 2, 2, 3, 3), g(2, 3, 1, 3, 1, 2)) = (f_1(1, 2, 3), f_2(1, 2, 3))$ is a tuple in $\mathbf{K}_6$ for any tuple $(f_1, f_2) \in \mathcal{F}_{\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)}(\mathbf{K}_3)$.

This concludes $\mathcal{F}_{\text{Pol}(\mathbf{K}_4, \mathbf{K}_5)}(\mathbf{K}_3) \to \mathbf{K}_6$, for the general case $\mathcal{F}_{\text{Pol}(\mathbf{K}_{k'}, \mathbf{K}_{q'})}(\mathbf{K}_k) \to \mathbf{K}_q$ the valid homomorphism sends $f \to f(1, 2, ..., k)$. We have now shown that $\text{Pol}(\mathbf{K}_{k'}, \mathbf{K}_{q'}) \to \text{Pol}(\mathbf{K}_k, \mathbf{K}_q)$ for $3 \leq k \leq k' \leq q' \leq q$.

A.1 $\mathbf{Pol}(\mathbf{R}_{k',q'}, \mathbf{H}_{k',2}) \to \mathbf{Pol}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ if $2 \leq q' \leq q < k \leq k'$

We show this for a specific case to make easier to follow and understand. The specific case can easily be generalised to all other cases.

We want to show that $\text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2}) \to \text{Pol}(\mathbf{R}_{5,4}, \mathbf{H}_{5,2}))$ and we do this using theorem 2.30. The minion homomorphism exists if and only if there exists a homomorphism of from $\mathcal{F}_{\text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2})}(\mathbf{R}_{5,4}) \to \mathbf{H}_{5,2}$. In the free structure we have the tuple $(f_1, f_2, f_3, f_4, f_5)$ for $f_1, f_2, f_3, f_4, f_5 \in \text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2})^4$ if there exists an $g \in \text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2})^m$ such that

$$f_1(x_1,x_2,x_3,x_4) = f(x_{i_{1,1}}, x_{i_{1,2}}, ..., x_{i_{1,m}})$$
$$f_2(x_1,x_2,x_3,x_4) = f(x_{i_{2,1}}, x_{i_{2,2}}, ..., x_{i_{2,m}})$$
$$f_3(x_1,x_2,x_3,x_4) = f(x_{i_{3,1}}, x_{i_{3,2}}, ..., x_{i_{3,m}})$$
$$f_4(x_1,x_2,x_3,x_4) = f(x_{i_{4,1}}, x_{i_{4,2}}, ..., x_{i_{4,m}})$$
$$f_5(x_1,x_2,x_3,x_4) = f(x_{i_{5,1}}, x_{i_{5,2}}, ..., x_{i_{5,m}})$$

where the indexes $i_{k,l} \in [4]$ and the m columns of indexes form all of the tuples in $\mathbf{R}_{5,4}$. To better motivate that our soon to be defined homomorphism $\mathcal{F}_{\text{Pol}(\mathbf{R}_{6,3},\mathbf{H}_{6,2})}(\mathbf{R}_{5,4}) \to \mathbf{H}_{5,2}$ is valid we complement our $f_i : i \in [5]$ with another row consisting of $f_6 = f_5$, so $i_{6,j} = i_{5,j}$ for all $j \in [m]$:

$$f_1(x_1,x_2,x_3,x_4) = f(x_{i_{1,1}}, x_{i_{1,2}}, ..., x_{i_{1,m}})$$
$$f_2(x_1,x_2,x_3,x_4) = f(x_{i_{2,1}}, x_{i_{2,2}}, ..., x_{i_{2,m}})$$
$$f_3(x_1,x_2,x_3,x_4) = f(x_{i_{3,1}}, x_{i_{3,2}}, ..., x_{i_{3,m}})$$
$$f_4(x_1,x_2,x_3,x_4) = f(x_{i_{4,1}}, x_{i_{4,2}}, ..., x_{i_{4,m}})$$
$$f_5(x_1,x_2,x_3,x_4) = f(x_{i_{5,1}}, x_{i_{5,2}}, ..., x_{i_{5,m}})$$
$$f_6(x_1,x_2,x_3,x_4) = f(x_{i_{6,1}}, x_{i_{6,2}}, ..., x_{i_{6,m}})$$

We define the homomorphism $\phi : \mathcal{F}_{\text{Pol}(\mathbf{R}_{6,3},\mathbf{H}_{6,2})}(\mathbf{R}_{5,4}) \to \mathbf{H}_{5,2}$ by $\phi(f) = f(1,2,3,3) \in [2] = \mathbf{H}_{5,2}$ as sets. This works since each of the columns above contain all variables x_1, x_2, x_3, x_4 as each of the first five indexes of the columns of form a rainbow coloring which means that substituting them for $1, 2, 3, 3$ every column is a tuple in $\mathbf{R}_{6,3}$. As g is a polymorphism in $\text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2})^{(m)}$ we have that the 6 rows are mapped to a tuple in $\mathbf{H}_{6,2}$, meaning that $(\phi(f_1), \phi(f_2), \phi(f_3), \phi(f_4), \phi(f_5), \phi(f_6))$ is a tuple in $\mathbf{H}_{6,2}$, but note that as $f_5 = f_6 \implies \phi(f_5) = \phi(f_6)$ we have that $(\phi(f_1), \phi(f_2), \phi(f_3), \phi(f_4), \phi(f_5))$ is a tuple in $\mathbf{H}_{5,2}$ showing that our mapping preserves the relation and is therefore a valid homomorphism.

This result can be easily generalised to there existing a $\phi : \mathcal{F}_{\text{Pol}(\mathbf{R}_{k',q'},\mathbf{H}_{k',2})}(\mathbf{R}_{k,q}) \to \mathbf{H}_{k,2}$ for any $2 \le q' \le q < k \le k'$. Just define $\phi(f) = f(1, 2, ..., q'-1, q', ..., q')$ for any $f \in \text{Pol}(\mathbf{R}_{k',q'}, \mathbf{H}_{k',2})^{(q)}$. It is a valid homomorphism with a similar motivation to the above.

Thus we not only conclude that $\text{Pol}(\mathbf{R}_{6,3}, \mathbf{H}_{6,2}) \to \text{Pol}(\mathbf{R}_{5,4}, \mathbf{H}_{5,2}))$, but that $\text{Pol}(\mathbf{R}_{k',q'}, \mathbf{H}_{k',2}) \to \text{Pol}(\mathbf{R}_{k,q}, \mathbf{H}_{k,2})$ for all $2 \le q' \le q < k \le k'$.

A.2 $\text{Pol}(\mathbf{LO}^r_{l'}, \mathbf{LO}^r_{k'}) \to \text{Pol}(\mathbf{LO}^r_l, \mathbf{LO}^r_k)$ if $2 \le l \le l' \le k' \le k$

We have from theorem 2.30 that $\text{Pol}(\mathbf{LO}^r_{l'}, \mathbf{LO}^r_{k'}) \to \text{Pol}(\mathbf{LO}^r_l, \mathbf{LO}^r_k)$ if and only if $\mathcal{F}_{\text{Pol}(\mathbf{LO}^r_{l'},\mathbf{LO}^r_{k'})}(\mathbf{LO}^r_l) \to \mathbf{LO}^r_k$. We show that this homomorphism exists for the specific case $\mathcal{F}_{\text{Pol}(\mathbf{LO}^4_3,\mathbf{LO}^4_4)}(\mathbf{LO}^4_2) \to \mathbf{LO}^4_5$ as it is easier to relate to and follow than a general proof. From this proof it is then easy to see how it can be generalised to any $2 \le l \le l' \le k' \le k$ and $r \ge 3$.

In $\mathcal{F}_{\text{Pol}(\mathbf{LO}^4_3,\mathbf{LO}^4_4)}(\mathbf{LO}^4_2)$ we have the tuple (f_1, f_2, f_3, f_4) where $f_1, f_2, f_3, f_4 \in \text{Pol}(\mathbf{LO}^4_3, \mathbf{LO}^4_4)^{(2)}$ are related if and only if

$$f_1(x,y) = g(y,x,x,x)$$
$$f_2(x,y) = g(x,y,x,x)$$
$$f_3(x,y) = g(x,x,y,x)$$
$$f_4(x,y) = g(x,x,x,y)$$

for some $g \in \text{Pol}(\mathbf{LO}^4_3, \mathbf{LO}^4_4)^{(4)}$ where the variables of the columns form all the tuples in $\mathbf{LO}^4_2$. The desired homomorphism $\mathcal{F}_{\text{Pol}(\mathbf{LO}^4_3,\mathbf{LO}^4_4)}(\mathbf{LO}^4_2) \to \mathbf{LO}^4_5$ can be defined by $f \to f(1,2) \forall f \in \text{Pol}(\mathbf{LO}^4_3, \mathbf{LO}^4_4)^{(2)}$.

This is a well-defined mapping to $\mathbf{LO}_5^4$ as f maps elements in $[3]^2$ to $[4] \subset [5] = LO_5^4$. The mapping also preserves the relation as four related elements f_1, f_2, f_3 and f_4 are mapped to $g(2,1,1,1)$, $g(1,2,1,1)$, $g(1,1,2,1)$ and $g(1,1,1,2)$ respectively. Note that for

$$\begin{gathered}(2,1,1,1)\\(1,2,1,1)\\(1,1,2,1)\\(1,1,1,2)\end{gathered}$$

every column is a tuple in $\mathbf{LO}_2^4$ meaning that $((2,1,1,1),(1,2,1,1),(1,1,2,1),(1,1,1,2))$ is a tuple in $(\mathbf{LO}_2^4)^4$. As $g \in \text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)^{(4)}$ we have that $(g(2,1,1,1), g(1,2,1,1), g(1,1,2,1), g(1,1,1,2))$ is a tuple in $\mathbf{LO}_4^4$, i.e has a unique maximum implying that $(f_1(1,2), f_2(1,2), f_3(1,2), f_4(1,2))$ is a tuple in $\mathbf{LO}_5^4$ for any tuple $(f_1, f_2, f_3, f_4) \in \mathcal{F}_{\text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)}(\mathbf{LO}_2^4)$.

This concludes $\mathcal{F}_{\text{Pol}(\mathbf{LO}_3^4, \mathbf{LO}_4^4)}(\mathbf{LO}_2^4) \to \mathbf{LO}_5^4$, for the general case $\text{Pol}(\mathbf{LO}_{l'}^r, \mathbf{LO}_{k'}^r) \to \text{Pol}(\mathbf{LO}_l^r, \mathbf{LO}_k^r)$ the valid homomorphism from the free structure to $\mathbf{LO}_k^r$ sends $f \to f(1,2,...,l)$.

B Code to verify 3-LO-coloring of $(\mathbf{LO}_2^6)^6$

```
#Assigning color to each vertex
values={}
values[(2,2,2,2,2,2)]=3

values[(2,2,2,2,2,1)]=1
values[(2,2,2,2,1,2)]=3
values[(2,2,2,1,2,2)]=3
values[(2,2,1,2,2,2)]=3
values[(2,1,2,2,2,2)]=3
values[(1,2,2,2,2,2)]=3

values[(1,1,2,2,2,2)]=3
values[(1,2,1,2,2,2)]=3
values[(1,2,2,1,2,2)]=2
values[(1,2,2,2,1,2)]=2
values[(1,2,2,2,2,1)]=1
values[(2,1,1,2,2,2)]=3
values[(2,1,2,1,2,2)]=3
values[(2,1,2,2,1,2)]=3
values[(2,1,2,2,2,1)]=2
values[(2,2,1,1,2,2)]=3
values[(2,2,1,2,1,2)]=3
values[(2,2,1,2,2,1)]=2
values[(2,2,2,1,1,2)]=3
values[(2,2,2,1,2,1)]=1
values[(2,2,2,2,1,1)]=1

values[(2,2,2,1,1,1)]=1
values[(2,2,1,2,1,1)]=2
values[(2,2,1,1,2,1)]=2
values[(2,2,1,1,1,2)]=3
```

```
values[(2,1,2,2,1,1)]=2
values[(2,1,2,1,2,1)]=2
values[(2,1,2,1,1,2)]=3
values[(1,2,2,2,1,1)]=1
values[(1,2,2,1,2,1)]=1
values[(1,2,2,1,1,2)]=3
values[(1,1,1,2,2,2)]=3

values[(2,1,1,2,2,1)]=1
values[(1,2,1,2,2,1)]=2
values[(1,1,2,2,2,1)]=2
values[(2,1,1,2,1,2)]=3
values[(1,2,1,2,1,2)]=3
values[(1,1,2,2,1,2)]=3
values[(2,1,1,1,2,2)]=3
values[(1,2,1,1,2,2)]=3
values[(1,1,2,1,2,2)]=3

values[(2,2,1,1,1,1)]=2
values[(2,1,2,1,1,1)]=2
values[(2,1,1,2,1,1)]=1
values[(2,1,1,1,2,1)]=1
values[(2,1,1,1,1,2)]=3
values[(1,2,2,1,1,1)]=1
values[(1,2,1,2,1,1)]=2
values[(1,2,1,1,2,1)]=2
values[(1,2,1,1,1,2)]=1
values[(1,1,2,2,1,1)]=2
values[(1,1,2,1,2,1)]=2
values[(1,1,2,1,1,2)]=1
values[(1,1,1,2,2,1)]=1
values[(1,1,1,2,1,2)]=3
values[(1,1,1,1,2,2)]=3

values[(2,1,1,1,1,1)]=1
values[(1,2,1,1,1,1)]=2
values[(1,1,2,1,1,1)]=2
values[(1,1,1,2,1,1)]=1
values[(1,1,1,1,2,1)]=1
values[(1,1,1,1,1,2)]=3

values[(1,1,1,1,1,1)]=1

count = [0]
def check_unique_max(x,y,z,a,b,c):
    """checks if our dictionary defines a valid polymorphism from (LO_2^6)^6 to LO_3^6 (3-LO-coloring
    list = [values[x],values[y],values[z],values[a],values[b],values[c]]
    #Checks if the maximum color/value is unique
    if list.count(max(list)) != 1:
        print("wrong", list, x,y,z,a,b,c)
        count[0]+=1
```

```
#creating all combinations of 6 columns. listi defines the i'th column. Then
count=0
for i1 in range(6):
    list1= [1,1,1,1,1,1]
    list1[i1]+=1
    for i2 in range(6):
        list2= [1,1,1,1,1,1]
        list2[i2]+=1
        for i3 in range(6):
            list3= [1,1,1,1,1,1]
            list3[i3]+=1
            for i4 in range(6):
                list4= [1,1,1,1,1,1]
                list4[i4]+=1
                for i5 in range(6):
                    list5= [1,1,1,1,1,1]
                    list5[i5]+=1
                    for i6 in range(6):
                        list6= [1,1,1,1,1,1]
                        list6[i6]+=1

                        #count+=1
                        check_unique_max((list1[0],list2[0],list3[0],list4[0],list5[0],list6[0]),(lis

#checking requirement holds for polymorphism
print(count)
for a in [1,2]:
    for b in [1,2]:
        for c in [1,2]:
            if values[(c,a,b,b,c,a)] == values[(a,b,c,a,b,c)]:
                print("ok")
            else:
                print("wrong")
```

TRITA-SCI-GRU 2024:342