The Complexity of Holant Problems over Boolean Domain with Non-Negative Weights*†

Jiabao Lin[1] and Hanpin Wang‡[2]

1 **Key Laboratory of High Confidence Software Technologies (MOE), School of Electronics Engineering and Computer Science, Peking University, Beijing, China**
joblin@pku.edu.cn

2 **Key Laboratory of High Confidence Software Technologies (MOE), School of Electronics Engineering and Computer Science, Peking University, Beijing, China**
whpxhy@pku.edu.cn

Abstract

Holant problem is a general framework to study the computational complexity of counting problems. We prove a complexity dichotomy theorem for Holant problems over the Boolean domain with non-negative weights. It is the first complete Holant dichotomy where constraint functions are not necessarily symmetric.

Holant problems are indeed read-twice #CSPs. Intuitively, some #CSPs that are #P-hard become tractable when restricted to read-twice instances. To capture them, we introduce the Block-rank-one condition. It turns out that the condition leads to a clear separation. If a function set $\mathcal{F}$ satisfies the condition, then $\mathcal{F}$ is of affine type or product type. Otherwise (a) Holant($\mathcal{F}$) is #P-hard; or (b) every function in $\mathcal{F}$ is a tensor product of functions of arity at most 2; or (c) $\mathcal{F}$ is transformable to a product type by some real orthogonal matrix. Holographic transformations play an important role in both the hardness proof and the characterization of tractability.

1998 ACM Subject Classification F.1.3 Complexity Measures and Classes

Keywords and phrases counting complexity, dichotomy, Holant, #CSP

Digital Object Identifier 10.4230/LIPIcs.ICALP.2017.29

1 Introduction

There has been considerable interest in several frameworks to study the complexity of counting problems. One natural framework is the counting Constraint Satisfaction Problem (#CSP) [18, 2, 19, 4, 22, 3, 8, 7, 1]. Another is Graph Homomorphism (GH) [30, 27, 21, 5, 20, 25, 6, 9], which can be seen as a special case of #CSP. Such frameworks express a large class of counting problems in the Sum-of-Product form. It is known that if P $\neq$ NP, then there exists a problem that is neither in P nor NP-complete [29]. And there is an analogue of Ladner's Theorem for the class #P. However, for these frameworks, various beautiful dichotomy theorems have been proved, classifying all problems in the broad class into those which are computable in polynomial time (in P) and those which are #P-hard. A natural question is: For how broad a class of counting problems can one prove a dichotomy theorem?

* A full version containing detailed proofs is available at https://arxiv.org/abs/1611.00975.

† This work was supported by the National Natural Science Foundation of China (Grants No. 61170299, 61370053 and 61572003).

‡ Hanpin Wang is the corresponding author.

44th International Colloquium on Automata, Languages, and Programming (ICALP 2017).
Editors: Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl;
Article No. 29; pp. 29:1–29:14
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

While GH can express many interesting graph parameters, Freedman, Lovász and Schrijver [24] showed that the number of perfect matchings of a graph cannot be represented as a homomorphism function. Inspired by holographic algorithms [32, 31], Cai, Lu and Xia [14] proposed a more refined framework called Holant Problems. Here we give a brief introduction. In this paper, constraint functions are defined over the Boolean domain, if not specified. Let $\mathcal{F}$ denote a set of algebraic complex-valued functions. A *signature grid* Ω is a tuple $(G, \mathcal{F}, \pi)$ where $G = (V, E)$ is an undirected graph, and π is a map that maps each vertex $v \in V$ to some function $f_v \in \mathcal{F}$ and its incident edges $E(v)$ to the input variables of f_v. The counting problem on Ω is to compute

$$\text{Holant}_\Omega = \sum_{\sigma:E\to\{0,1\}} \prod_{v\in V} f_v(\sigma|_{E(v)}),$$

where $\sigma|_{E(v)}$ is the restriction of σ to $E(v)$. All such signature grids constitute the set of instances of the problem Holant($\mathcal{F}$). For example, consider the problem of counting perfect matchings (#PM) on graph G. In a perfect matching, every vertex is saturated by exactly one edge. Such constraint on a vertex of degree n can be expressed as an Exact-One function $f : \{0,1\}^n \to \{0,1\}$, which takes the value 1 if and only if its input has Hamming weight 1. If every vertex is assigned such a function, then the value Holant_Ω is exactly the number of perfect matchings. Let $\mathcal{F}$ denote the set of all Exact-One functions, then Holant($\mathcal{F}$) represents the problem #PM.

The Holant framework is general enough: #CSPs can be viewed as special Holant problems where all equality functions are available [14]. However, the very generality makes it more difficult to prove a dichotomy. A function is *symmetric* if the function values only depend on the Hamming weights of inputs, like the Exact-One functions. Satisfactory progress has been made in the complexity classification of Holant problems specified by sets of symmetric functions [13, 28, 26, 11, 10]. And in the process, some unexpected tractable classes were discovered. They give many deep insights into both tractability and hardness.

It still remains open whether a complete dichotomy exists, since the definition of Holant problems does not require that constraint functions be symmetric. Such restriction is stringent and generally it is not imposed in #CSP. Cai, Lu and Xia [16] proved a dichotomy without symmetry for a special family of Holant problems, called Holant*, where all unary functions are assumed to be available. But without this assumption, as in [11], more tractable classes will be released, which makes the hardness proof very different.

We prove a dichotomy theorem for Holant problems with non-negative algebraic real weights. It is the first complete Holant dichotomy where constraint functions are not necessarily symmetric and no auxiliary function is assumed to be available. This generalizes the results on Boolean #CSP in [18, 19], and the dichotomies in [28, 11] restricted to non-negative case. Our proof starts with an infinitary condition, but finally obtains an explicit criterion (Theorem 19).

A simple observation is that, Holant problems are indeed *read-twice* #CSPs where every variable in an instance appears exactly twice (see subsection 2.4). Intuitively, some #CSPs that are #P-hard become tractable when restricted to read-twice instances. To capture them, we need insights into what makes a problem hard in #CSP. Inspired by dichotomy theorems over general domains [5, 23, 8, 7], we introduce the Block-rank-one condition for Holant problems (see subsection 7.1). It is known that non-block-rank-one structures imply hardness in #CSP. So our condition is necessary for tractability since it is imposed on the functions defined by read-twice instances. Surprisingly, on the Boolean domain, the Block-rank-one condition is also *sufficient* and leads to a clear separation:

I. Function set $\mathcal{F}$ satisfies the condition. Then #CSP($\mathcal{F}$) is in P, and hence its subproblem Holant($\mathcal{F}$) is also in P.

II. Function set $\mathcal{F}$ violates the condition. Then (a) Holant($\mathcal{F}$) is #P-hard or (b) #CSP($\mathcal{F}$) is #P-hard but Holant($\mathcal{F}$) is tractable.

First we discuss Part II. We can prove #P-hardness directly, or further induce an orthogonal holographic transformation. After performing the transformation, we have to handle real-valued functions. Luckily, we can even prove a dichotomy theorem for a family of *complex-valued* Holant problems (Theorem 9). And towards this theorem, we prove a lemma (Lemma 6) on how to "extract" a function from its tensor powers. The proof is non-constructive and the idea can simplify some existing proofs. For example, it can be shown directly that the two problems $\#\text{CSP}^d(\mathcal{F}\cup\{[1,0]^{\otimes d},[0,1]^{\otimes d}\})$ and $\#\text{CSP}^d(\mathcal{F}\cup\{[1,0],[0,1]\})$ in [28] are equivalent under polynomial-time Turing reduction.

Now consider Part I. It can be derived that $\mathcal{F}$ is of affine type or $\mathcal{F}$ is of product type, exactly the criterion given by Dyer, Goldberg and Jerrum [19]. Dichotomies for #CSP over general domains [1, 23, 3, 8] are very different from those over the Boolean domain [18, 19]. Our proof builds a connection between them.

The Block-rank-one condition is a little conceptual. To obtain the structure of $\mathcal{F}$, we introduce an equivalent notion, called *balance*, for Holant problems (see subsection 7.2). The equivalence is simply built on the concept of *vector representation* in [8], which was used to design a polynomial-time algorithm for #CSP. Back to non-negative #CSP, we find that actually the notions of weak balance and balance (different from our version for Holant) in [8] are equivalent, without assuming FP $\neq$ #P. Therefore, to decide the complexity of a problem #CSP($\mathcal{F}$), we only need to decide whether $\mathcal{F}$ is of weak balance.

2 Preliminaries

2.1 Functions and Signatures

Let $\mathbb{C}$ and $\mathbb{R}_+$ denote the set of algebraic complex numbers and the set of algebraic non-negative real numbers, respectively. Throughout this paper, we refer to them simply as complex and non-negative numbers.

Given a function $f:\{0,1\}^n\to\mathbb{C}$, we will often write it as a vector of dimension 2^n whose entries are the function values, indexed by $\mathbf{x}\in\{0,1\}^n$ lexicographically. This vector is called a *signature*. If the values of an n-ary function only depend on the Hamming weights of inputs, then the function is called *symmetric* and can be expressed as $[f_0,f_1,...,f_n]$ where f_k is the function value for inputs of Hamming weight k. For example, the ternary logic OR function has the signature $[0,1,1,1]$.

Generally, given a function f of arity n, we can express it as a $2^r\times 2^{n-r}$ matrix $(1\le r\le n)$, denoted by $M_{[r]}(f)$. The rows and columns are indexed by $\mathbf{x}\in\{0,1\}^r$ and $\mathbf{y}\in\{0,1\}^{n-r}$ respectively, and $f(\mathbf{x},\mathbf{y})$ is the $(\mathbf{x},\mathbf{y})^{\text{th}}$ entry of the matrix. And the matrices $\{M_{[r]}(f)\,|\,r\in[n]\}$ are called the *signature matrices* of f. When the integer r is clear from the context, we simply write M_f.

In most cases, if not confused, we identify functions, signatures and signature matrices. But in section 7, we shall distinguish a function from its matrix representations.

Given an n-ary function f and a permutation π on $[n]$, we define the function f_π : For $x_1,x_2,...,x_n\in\{0,1\}$, $f_\pi(x_1,x_2,...,x_n)=f(x_{\pi(1)},x_{\pi(2)},...,x_{\pi(n)})$.

A function F is *reducible* if F_π is a tensor product of two functions (of arity ≥ 1) for some permutation π. Otherwise F is called *irreducible*. A function is called *degenerate* if it is a tensor product of some unary functions. Otherwise we call it *non-degenerate*.

Given a positive integer k, we use $=_k$ to denote the k-ary *equality* function $[1, 0, ..., 0, 1]$. And we use $\neq_2$ to denote the binary *disequality* function $[0, 1, 0]$.

In the following, we define three classes of complex-valued functions. Let $\mathcal{T}$ denote the set of functions that can be expressed as a tensor product of functions of arity at most 2.

The *support* of an n-ary function f, denoted by $\text{supp}(f)$, is the set $\{\mathbf{x} \in \mathbb{Z}_2^n \mid f(\mathbf{x}) \neq 0\}$. A Boolean relation is *affine* if it is the set of solutions to a system of linear equations over the field $\mathbb{Z}_2$. We say that f has affine support if its support is affine.

▶ **Definition 1.** A function f of arity n is *affine* if its support is affine and there is a constant $\lambda \in \mathbb{C}$ such that for all $\mathbf{x} \in \text{supp}(f)$, $f(\mathbf{x}) = \lambda \cdot i^{Q(\mathbf{x})}$, where $i = \sqrt{-1}$ and Q is a quadratic polynomial $Q(x_1, ..., x_n) = \sum_{i=1}^{n} a_i x_i^2 + 2\sum_{1\leq i<j\leq n} b_{ij} x_i x_j$ with $a_i \in \mathbb{Z}_4$ and $b_{ij} \in \{0, 1\}$. We use $\mathcal{A}$ to denote the set of all affine functions.

▶ **Definition 2.** A function f is of *product type* if it can be expressed as a product of unary functions, binary functions of the form $=_2$ and $\neq_2$ (on not necessarily disjoint subsets of variables). We use $\mathcal{P}$ to denote the set of all functions of product type.

2.2 Holographic Reductions

To introduce the holographic reductions, we define bipartite Holant problems. $\text{Holant}(\mathcal{F} \mid \mathcal{G})$ denotes the Holant problem on bipartite graphs $H = (U, V, E)$ where each vertex in U (V) is assigned a function from $\mathcal{F}$ ($\mathcal{G}$). A Holant problem $\text{Holant}(\mathcal{F})$ can seen as the bipartite problem $\text{Holant}(=_2 \mid \mathcal{F})$.

Let T be a 2×2 matrix and let $\mathcal{F}$ be a function set. Whenever we write $T\mathcal{F}$, the functions in $\mathcal{F}$ are viewed as column vectors and, $T\mathcal{F} = \{T^{\otimes n} f \mid f \in \mathcal{F} \text{ and } n = \text{arity}(f)\}$. Similarly, $\mathcal{F}T = \{fT^{\otimes n} \mid f \in \mathcal{F} \text{ and } n = \text{arity}(f)\}$ where the functions in $\mathcal{F}$ are expressed as row vectors.

Let T be a matrix in $GL_2(\mathbb{C})$. We say there is a *holographic reduction* defined by T from $\text{Holant}(\mathcal{F} \mid \mathcal{G})$ to $\text{Holant}(\mathcal{F}' \mid \mathcal{G}')$, if $\mathcal{F}T \subseteq \mathcal{F}'$ and $T^{-1}\mathcal{G} \subseteq \mathcal{G}'$. The holographic reduction maps a signature grid $\Omega = (G, \mathcal{F} \mid \mathcal{G}, \pi)$ to $\Omega' = (G, \mathcal{F}' \mid \mathcal{G}', \pi')$: For each vertex v of G, π' assigns the function $f_v T$ or $T^{-1} f_v$ to v, depending on which part v belongs to.

▶ **Theorem 3** (Valiant's Holant Theorem [32])**.** *Let T be any matrix in $GL_2(\mathbb{C})$. Suppose that the holographic reduction defined by T maps a signature grid Ω to Ω'. Then* $\text{Holant}_\Omega = \text{Holant}_{\Omega'}$.

We will use $\leq_\mathsf{T}$ to denote polynomial-time Turing reductions and use $\equiv_\mathsf{T}$ to denote the equivalence relation under polynomial-time Turing reductions.

▶ **Theorem 4.** *Let F be a function set and let H be an orthogonal matrix ($H^\mathsf{T} H = I$). Then* $\text{Holant}(H\mathcal{F}) \equiv_\mathsf{T} \text{Holant}(\mathcal{F})$.

2.3 Realizability

Let $\mathcal{F}$ be a set of functions. An $\mathcal{F}$-gate [15] Γ is a tuple $(G, \mathcal{F}, \pi)$ where $G = (V, E, D)$ is a graph with regular edges E and some dangling edges D. Other than these dangling edges, the gate Γ is the same as a signature grid: π maps each vertex $v \in V$ to some function $f_v \in \mathcal{F}$ and it incident edges (including the dangling ones) to the input variables of f_v. We denote the edges in E by $1, 2, ..., m$ and the dangling edges in D by $m + 1, m + 2, ..., m + n$. Then we can define a function f for Γ:

$$f(y_1, y_2, ..., y_n) = \sum_{x_1, x_2, ..., x_m \in \{0,1\}} F(x_1, x_2, ..., x_m, y_1, y_2, ..., y_n)$$

where $(y_1, y_2, ..., y_n) \in \{0,1\}^n$ is an assignment on the dangling edges and $F(\mathbf{x}, \mathbf{y})$ denotes the product of evaluations at all vertices of V. We say the function f is *realizable* from the function set $\mathcal{F}$. We use $\mathsf{S}(\mathcal{F})$ to denote the set of functions realizable from $\mathcal{F}$.

Given a function f, we use $f^{x_i=c}$ to denote the function obtained by pinning the ith input variable of f to $c \in \{0,1\}$.

2.4 Weighted Counting CSP

Let $\mathcal{F}$ be a set of complex-valued functions. Then the problem #CSP($\mathcal{F}$) is defined as follows. An input instance I of the problem consists of a finite set of variables $V = \{x_1, ..., x_n\}$ and a finite set of constraints $\{C_1, ..., C_m\}$. Each C_i has the form $(F_i, \mathbf{x}_i)$ where $F_i \in \mathcal{F}$ and $\mathbf{x}_i$ is a tuple of (not necessarily distinct) variables from V. The instance I *defines* a function F_I over $\mathbf{x} = (x_1, ..., x_n) \in \{0,1\}^n$: $F_I(\mathbf{x}) = \prod_{i=1}^m F_i(\mathbf{x}_i)$ for $\mathbf{x} \in \{0,1\}^n$. The output is the sum: $Z(I) = \sum_{\mathbf{x}\in\{0,1\}^n} F_I(\mathbf{x})$.

Holant problems are indeed read-twice #CSPs. Given a signature grid, we assume that the numbering of its vertices and edges is also given. If these edges are viewed as variables, then the signature grid is a #CSP instance where every variable appears exactly twice. So we also say that a signature grid defines a function. And the concept of realizability can be defined in the CSP language.

Cai, Lu and Xia [17] proved a dichotomy for complex-weighted #CSP over the Boolean domain.

▶ **Theorem 5** ([17]). *Let $\mathcal{F}$ be a set of complex-valued functions. Then the problem* #CSP($\mathcal{F}$) *is computable in polynomial time if $\mathcal{F} \subseteq \mathcal{A}$ or $\mathcal{F} \subseteq \mathcal{P}$. Otherwise* #CSP($\mathcal{F}$) *is* #P*-hard.*

3 Decomposition

In Holant problems, sometimes we are able to realize a function $F = f \otimes g$, but do not know how to realize the function f directly, which can be technically beneficial. Fortunately, under certain conditions, if F is realizable, then we may assume that f is freely available.

In this section, we prefer to prove the lemmas in the CSP language. If not specified, the functions we discussed are over a fixed finite domain and take complex values.

Let m be a positive integer. We use $f^{\otimes m}$ to denote the m-th tensor power of f. $f^{\otimes m}$ can be seen as m copies of f: $f^{\otimes m}(\mathbf{x}_1, ..., \mathbf{x}_m) = f(\mathbf{x}_1)\cdots f(\mathbf{x}_m)$. Let I be a #CSP instance that contains m constraints: $(f, \mathbf{x}_1), (f, \mathbf{x}_2), ..., (f, \mathbf{x}_m)$. We replace these m tuples by one tuple $(f^{\otimes m}, \mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_m)$ and then obtain a new instance I'. It is easy to see that $Z(I) = Z(I')$.

▶ **Lemma 6.** *For any function set $\mathcal{F}$ and function f,* $\text{Holant}(\mathcal{F} \cup \{f\}) \leq_{\mathsf{T}} \text{Holant}(\mathcal{F} \cup \{f^{\otimes d}\})$ *for all $d \geq 1$.*

Proof. Impose induction on d. Let n denote the arity of f.

The base case, $d = 1$, is trivial. Now suppose that the conclusion holds for all $d < k$ ($k \geq 2$). In the problem $\text{Holant}(\mathcal{F} \cup \{f^{\otimes k}\})$, we may assume that the functions $f^{\otimes(mk)}$ are freely available for integers $m > 0$. There are two cases to consider:

- There exists an instance I of $\text{Holant}(\mathcal{F} \cup \{f\})$ such that $Z(I) \neq 0$ and f appears p times where $p = qk + r$ ($q \geq 0, 0 < r < k$). Let $C_1, ..., C_p$ be the p constraints that have the form $(f, \mathbf{x}_i)$. We replace the first qk constraints by one tuple $C_1' = (f^{\otimes(qk)}, \mathbf{x}_1, ..., \mathbf{x}_{qk})$, and the last r constraints by one tuple $C_2' = (f^{\otimes k}, \mathbf{x}_{qk+1}, ..., \mathbf{x}_p, \mathbf{y})$ where $\mathbf{y}$ denotes a list of new distinct variables, of length $(k-r)n$. After the substitution, we get a function

$F(\mathbf{x}, \mathbf{y})$ where $\mathbf{x}$ denotes the variables of the original instance I. Every variable in $\mathbf{x}$ occurs twice, so by summing on them we can realize the following function:

$$\sum_{\mathbf{x}} F(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{x}} F_I(\mathbf{x}) f^{\otimes(k-r)}(\mathbf{y}) = Z(I) f^{\otimes(k-r)}(\mathbf{y}).$$

Because $Z(I) \neq 0$, we have $\text{Holant}(\mathcal{F} \cup \{f^{\otimes(k-r)}\}) \leq_{\mathsf{T}} \text{Holant}(\mathcal{F} \cup \{f^{\otimes k}\})$. And by the induction hypothesis, $\text{Holant}(\mathcal{F} \cup \{f\}) \leq_{\mathsf{T}} \text{Holant}(\mathcal{F} \cup \{f^{\otimes(k-r)}\})$. Therefore, the conclusion holds.

- For all I with $Z(I) \neq 0$, f appears a multiple of k times. Given an instance I of $\text{Holant}(\mathcal{F} \cup \{f\})$, we show how to compute $Z(I)$ with the help of the oracle for $\text{Holant}(\mathcal{F} \cup \{f^{\otimes k}\})$. First we check whether the number p of constraints containing f is a multiple of k. If not, we simply output 0. Otherwise we replace all such constraints by one tuple $(f^{\otimes p}, \mathbf{x})$ as in case (1), and then obtain an instance I' of $\text{Holant}(\mathcal{F} \cup \{f^{\otimes k}\})$. Clearly $Z(I) = Z(I')$, and we can compute $Z(I')$ by accessing the oracle.

In either case, there exists a polynomial-time Turing reduction. This completes the induction. ◀

Note that our proof only shows the *existence* of polynomial-time Turing reductions, but does not produce such reductions *constructively* for given function sets. Based on Lemma 6, we can prove a more general one.

▶ **Lemma 7.** *Let $\mathcal{F}$ be a set of functions, and f, g be two functions. Suppose that there exists an instance I of $\text{Holant}(\mathcal{F} \cup \{f, g\})$ such that $Z(I) \neq 0$, and the number of occurrences of g in I is greater than that of f. Then $\text{Holant}(\mathcal{F} \cup \{f, f \otimes g\}) \leq_{\mathsf{T}} \text{Holant}(\mathcal{F} \cup \{f \otimes g\})$.*

4 When A Non-trivial Equality Function Appears

Let $\text{Holant}^c(\mathcal{F})$ denote the problem $\text{Holant}(\mathcal{F} \cup \{[1,0],[0,1]\})$. We have the following theorem:

▶ **Theorem 8.** *Let λ be any nonzero complex number that is not a root of unity. For any set $\mathcal{F}$ of complex-valued functions, $\text{Holant}^c(\mathcal{F} \cup \{[1,0,\lambda]\})$ is computable in polynomial time if $\mathcal{F} \subseteq \mathscr{T}$ or $\mathcal{F} \subseteq \mathscr{P}$. Otherwise the problem is #P-hard.*

The conclusion still holds if we remove the unary functions $[1,0]$ and $[0,1]$:

▶ **Theorem 9.** *Let λ be any nonzero complex number that is not a root of unity. For any set $\mathcal{F}$ of complex-valued functions, $\text{Holant}(\mathcal{F} \cup \{[1,0,\lambda]\})$ is computable in polynomial time if $\mathcal{F} \subseteq \mathscr{T}$ or $\mathcal{F} \subseteq \mathscr{P}$. Otherwise the problem is #P-hard.*

Proof. We can interpolate $[1,0]^{\otimes 2}$ and $[0,1]^{\otimes 2}$ using $[1,0,\lambda]$. Then by Lemma 6, $\text{Holant}^c(\mathcal{F} \cup \{[1,0,\lambda]\}) \leq_{\mathsf{T}} \text{Holant}(\mathcal{F} \cup \{[1,0,\lambda]\})$. ◀

Intuitively, we can interpolate all functions of the form $[a,0,b]$, using the binary function $[1,0,\lambda]$. By connecting with these binary funtions, a function f may range arbitrarily. To avoid #P-hardness, the structure of the support of f must be simple enough.

5 $\mathscr{P}$-transformability

We start with some simple facts from linear algebra. Let $M = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \\ b_1 & b_2 & \cdots & b_n \end{bmatrix}$ $(n \geq 2)$ be a non-negative matrix of rank 2. Then $A = MM^{\mathsf{T}} = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$ satisfying $a, c > 0$. Moreover, by Cauchy-Schwarz inequality, $\det A = ac - b^2 > 0$.

▶ **Lemma 10.** *If $a \neq c$ or $b \neq 0$, then A has two distinct positive eigenvalues α and β.*

The following lemma is a simple case of the Spectral Theorem for real symmetric matrices.

▶ **Lemma 11.** *There is an orthogonal matrix H such that $HAH^{\mathsf{T}} = \begin{bmatrix}\alpha & 0\\ 0 & \beta\end{bmatrix}$, where α and β are the eigenvalues of A.*

Let f be a non-negative binary function. If f is non-degenerate and affine, then $f = a[1, 0, 1]$ or $f = a[0, 1, 0]$ for some $a > 0$.

▶ **Lemma 12.** *Let $f = (a, b, c, d)$ be a non-negative function. Suppose that f is non-degenerate and $f \notin \mathcal{A}$. Then for any function set $\mathcal{F}$ with $f \in \mathsf{S}(\mathcal{F})$,* Holant$(\mathcal{F})$ *is #*P*-hard or $\mathcal{F} \subseteq \mathcal{T}$ or $\mathcal{F} \subseteq H\mathcal{P}$ for some orthogonal matrix H.*

Proof. Since $f \in \mathsf{S}(\mathcal{F})$, the symmetric matrix

$$A = \begin{bmatrix}a & b\\ c & d\end{bmatrix}\begin{bmatrix}a & c\\ b & d\end{bmatrix} = \begin{bmatrix}a^2+b^2 & ac+bd\\ ac+bd & c^2+d^2\end{bmatrix}$$

is also realizable. Because f is non-degenerate, $a^2 + b^2, c^2 + d^2 > 0$ and $ac + bd \geq 0$. We claim that $ac + bd \neq 0$ or $a^2 + b^2 \neq c^2 + d^2$. Suppose $ac + bd = 0$, then $ac = bd = 0$ since f is non-negative. So $f = \begin{bmatrix}a & 0\\ 0 & d\end{bmatrix}$ or $f = \begin{bmatrix}0 & b\\ c & 0\end{bmatrix}$. In both cases, as $f \notin \mathcal{A}$, $a^2 + b^2 \neq c^2 + d^2$.

By Lemma 10 and Lemma 11, there is some orthogonal matrix H such that $HAH^{\mathsf{T}} = \begin{bmatrix}\alpha & 0\\ 0 & \beta\end{bmatrix}$, where α and β are the two distinct positive eigenvalues of A. Now we perform the transformation H and obtain the following equivalence:

$$\text{Holant}(\{[\alpha, 0, \beta]\} \cup H\mathcal{F}) \equiv_{\mathsf{T}} \text{Holant}(\{A\} \cup \mathcal{F}) \equiv_{\mathsf{T}} \text{Holant}(\mathcal{F}).$$

The latter equivalence follows from the fact $A \in \mathsf{S}(f) \subseteq \mathsf{S}(\mathcal{F})$. β/α is nonzero and not a root of unity, so if $H\mathcal{F} \not\subseteq \mathcal{T}$ and $H\mathcal{F} \not\subseteq \mathcal{P}$, the problem is #P-hard by Theorem 9. ◀

6 On Special Functions of Arity 4

In this section, we consider some special functions of arity 4, and complete the preparation for the hardness part of our dichotomy.

▶ **Lemma 13.** *Let f be a function of arity 4, whose signature matrix has the form*

$$M_f = \begin{bmatrix}f_{0000} & f_{0001} & f_{0010} & f_{0011}\\ f_{0100} & f_{0101} & f_{0110} & f_{0111}\\ f_{1000} & f_{1001} & f_{1010} & f_{1011}\\ f_{1100} & f_{1101} & f_{1110} & f_{1111}\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & a\\ 0 & b & c & 0\\ 0 & c & b & 0\\ a & 0 & 0 & 1\end{bmatrix}$$

where $a, b, c \geq 0$ and at least two of them are positive. Then Holant(f) *is #*P*-hard if $f \neq [1, 0, 1, 0, 1]$.*

We prove a dichotomy for function sets that contain certain functions of arity 4.

▶ **Lemma 14.** *Let f be a non-negative function of arity 4. And $\begin{bmatrix}f_{0000} & f_{0011}\\ f_{1100} & f_{1111}\end{bmatrix} = \begin{bmatrix}a & b\\ b & c\end{bmatrix}$ where $b \neq 0$ and $ac > b^2$. Then for any function set $\mathcal{F}$ containing f,* Holant$(\mathcal{F})$ *is #*P*-hard or $\mathcal{F} \subseteq \mathcal{T}$ or $\mathcal{F} \subseteq H\mathcal{P}$ for some orthogonal matrix H.*

Proof. We can show that Holant$(\mathcal{F})$ is #P-hard by Theorem 5 and Lemma 13, or there is some non-negative binary function $f \notin \mathcal{A} \cup \mathcal{P}$ such that Holant$(\mathcal{F} \cup \{f\}) \leq_{\mathsf{T}}$ Holant$(\mathcal{F})$. Then the conclusion follows from Lemma 12. ◀

7 The Dichotomy

7.1 The Block-rank-one Condition Captures the Dichotomy

Given a function f of arity n, we use $f^{[t]}$, for each $t \in [n]$, to denote the function

$$f^{[t]}(x_1, ..., x_t) = \sum_{x_{t+1},...,x_n \in \{0,1\}} f(x_1, ..., x_t, x_{t+1}, ..., x_n).$$

Recall that Holant problems are read-twice #CSPs and every #CSP instance *defines* a function (subsection 2.4). We adopt the notation in [7], defining the following set of functions for a given $\mathcal{F}$:

$$\mathcal{W}_{\mathcal{F}} = \{F^{[t]} \mid F \text{ is a function defined by an instance of } \operatorname{Holant}(\mathcal{F}) \text{ and } 1 \leq t \leq \text{arity of } F\}.$$

Note that the functions in $\mathcal{W}_{\mathcal{F}}$ are not necessarily realizable from $\mathcal{F}$. The following two lemmas show how $\mathcal{W}_{\mathcal{F}}$ and $\mathsf{S}(\mathcal{F})$ are related:

▶ **Lemma 15.** *Let $f \in \mathcal{W}_{\mathcal{F}}$ be a function of arity n. Then there is a function $g \in \mathsf{S}(\mathcal{F})$ of arity $2n$, such that for all $x_1, x_2, ..., x_n \in \{0, 1\}$, $f(x_1, x_2, ..., x_n) = g(x_1, x_1, x_2, x_2, ..., x_n, x_n)$.*

▶ **Lemma 16.** *For $f \in \mathsf{S}(\mathcal{F})$, $f^2 \in \mathcal{W}_{\mathcal{F}}$.*

Let M be a non-negative matrix. We say M is *block-rank-one* if every two rows of it are linearly dependent or orthogonal. Given a non-negative function f of arity n, we say f is *block-rank-one* if either $n = 1$ or the matrix $M_{[n-1]}(f)$ is block-rank-one.

Now we impose a condition on $\mathcal{W}_{\mathcal{F}}$:

Block-rank-one: All functions in $\mathcal{W}_{\mathcal{F}}$ are block-rank-one.

We can classify those function sets that do not satisfy this condition:

▶ **Lemma 17.** *Let $\mathcal{F}$ be a set of non-negative functions. If $\mathcal{F}$ does not satisfy the Block-rank-one condition, then* $\operatorname{Holant}(\mathcal{F})$ *is* #P*-hard or $\mathcal{F} \subseteq \mathcal{T}$ or $\mathcal{F} \subseteq H\mathcal{P}$ for some orthogonal matrix H.*

Proof. Let $f \in \mathcal{W}_{\mathcal{F}}$ be a function of arity n. Then by Lemma 15, there is a function $g \in \mathsf{S}(\mathcal{F})$ of arity $2n$, such that for all $x_1, x_2, ..., x_n \in \{0, 1\}$, $f(x_1, x_2, ..., x_n) = g(x_1, x_1, x_2, x_2, ..., x_n, x_n)$.

Now suppose that f is not block-rank-one. By definition, $n \geq 2$ and the two columns of $M_{[n-1]}(f)$ are linearly independent but not orthogonal. Then the first and the last columns of the matrix $M = M_{[2n-2]}(g)$, $g^{x_{2n-1}=x_{2n}=0}$ and $g^{x_{2n-1}=x_{2n}=1}$, are also linearly independent but not orthogonal. Let h denote the 4×4 matrix $M^{\mathsf{T}}M$. Then $h_{0011} = h_{1100} > 0$ and $h_{0000}h_{1111} > h_{0011}^2$. Since $g \in \mathsf{S}(\mathcal{F})$, h is also realizable. Thus $\operatorname{Holant}(\mathcal{F} \cup \{h\}) \leq_{\mathsf{T}} \operatorname{Holant}(\mathcal{F})$. By Lemma 14, $\operatorname{Holant}(\mathcal{F})$ is #P-hard or $\mathcal{F} \subseteq \mathcal{T}$ or $\mathcal{F} \subseteq H\mathcal{P}$ for some orthogonal H. ◀

Surprisingly, the Block-rank-one condition has *captured* the dichotomy. We have the crucial lemma below:

▶ **Lemma 18.** *Let $\mathcal{F}$ be a set of non-negative functions. If $\mathcal{F}$ satisfies the Block-rank-one condition, then $\mathcal{F} \subseteq \mathcal{A}$ or $\mathcal{F} \subseteq \mathcal{P}$.*

Therefore, if $\mathcal{F}$ satisfies the Block-rank-one condition, then $\operatorname{Holant}(\mathcal{F})$ is in polynomial time. So our dichotomy is quite simple and it is decidable in polynomial time [12]:

▶ **Theorem 19.** *Let $\mathcal{F}$ be a set of non-negative functions. The problem* Holant($\mathcal{F}$) *is computable in polynomial time if $\mathcal{F}$ satisfies one of the following three conditions:*
- $\mathcal{F} \subseteq \mathcal{T}$;
- $\mathcal{F} \subseteq \mathcal{A}$;
- $\mathcal{F} \subseteq H\mathcal{P}$ *for some real orthogonal matrix H.*

Otherwise Holant($\mathcal{F}$) *is* #P*-hard.*

The remaining is to prove Lemma 18. To obtain the structure of $\mathcal{F}$, it is more convenient to consider directly the set $\mathcal{F}$ and the functions realizable from it. So in the next subsection, we will introduce a notion equivalent to the Block-rank-one condition. This notion restricts the function set $\mathsf{S}(\mathcal{F})$.

7.2 Balance

We define the notion of *balance* for non-negative Holant problems. The notion was introduced for non-negative #CSP by Cai, Chen and Lu [8].

▶ **Definition 20** (Balance). Let $\mathcal{F}$ be a set of non-negative functions. $\mathcal{F}$ is called *balanced* if for any function $f \in \mathsf{S}(\mathcal{F})$, every signature matrix in $\{M_{[r]}(f) \,|\, 1 \le r \le \text{arity}(f)\}$ is block-rank-one. A non-negative function f is *balanced* if the set $\{f\}$ is balanced.

Note that in the definition above, when $r = \text{arity}(f)$, the matrix $M_{[r]}(f)$ is a column vector and hence trivially block-rank-one.

Balanced sets satisfy the Block-rank-one condition. Generally, we have the following lemma.

▶ **Lemma 21.** *Let $\mathcal{F}$ be a set of non-negative functions. Suppose that $\mathcal{F}$ is balanced. Then for any $f \in \mathcal{W}_{\mathcal{F}}$, every matrix in $\{M_{[r]}(f) \,|\, 1 \le r \le \text{arity}(f)\}$ is block-rank-one.*

Proof. Let $f \in \mathcal{W}_{\mathcal{F}}$ be a function of arity n. Then by Lemma 15, there exists a function $g \in \mathsf{S}(\mathcal{F})$ of arity $2n$, such that for all $x_1, x_2, ..., x_n \in \{0,1\}$,

$$f(x_1, x_2, ..., x_n) = g(x_1, x_1, x_2, x_2, ..., x_n, x_n).$$

Therefore, for any $r \in [n]$, $M_{[r]}(f)$ is a submatrix of $M_{[2r]}(g)$. Because $\mathcal{F}$ is balanced, $M_{[2r]}(g)$ is block-rank-one. Hence so is $M_{[r]}(f)$. ◀

Let f be a non-negative function of arity n. And let $s_1, ..., s_n$ be n non-negative unary functions. We call $(s_1, ..., s_n)$ a *vector representation* of f if for all $\mathbf{x} \in \{0,1\}^n$, either $f(\mathbf{x}) = 0$ or $f(\mathbf{x}) = s_1(x_1) \cdots s_n(x_n)$.

▶ **Lemma 22** ([8]). *Let f be a non-negative function of arity n. If $f^{[t]}$ is block-rank-one for all $t \in [n]$, then f has a vector representation.*

▶ **Lemma 23.** *Let $\mathcal{F}$ be a set of non-negative functions that satisfies the Block-rank-one condition. Then every function in $\mathsf{S}(\mathcal{F})$ has a vector representation.*

Proof. Let f be a function in $\mathsf{S}(\mathcal{F})$ of arity n. By Lemma 16, $f^2 \in \mathcal{W}_{\mathcal{F}}$. Then f^2 has a vector representation $(s_1, ..., s_n)$ by Lemma 22. Let $(s'_1, ..., s'_n)$ be n non-negative unary functions such that for all $i \in [n]$, $s'_i(a) = \sqrt{s_i(a)}$ for $a \in \{0,1\}$. Then $(s'_1, ..., s'_n)$ is a vector representation of the function f. ◀

Now we are able to prove the equivalence between the notion of balance and the Block-rank-one condition.

▶ **Lemma 24.** *Let $\mathcal{F}$ be a set of non-negative functions. $\mathcal{F}$ is balanced if and only if $\mathcal{F}$ satisfies the Block-rank-one condition.*

Proof. The necessity follows directly from Lemma 21. We only need to show the sufficiency.

Let f be an n-ary function in $\mathsf{S}(\mathcal{F})$, with $n \geq 2$. And suppose that $M = M_{[r]}(f)$ is not block-rank-one for some $r \in [n]$. Then there exist two rows of M, indexed by some $\mathbf{x}, \mathbf{y} \in \{0,1\}^r$, which are linearly independent but not orthogonal. So we can realize a signature $g = MM^{\mathsf{T}}$. Its submatrix

$$h = \begin{bmatrix} g(\mathbf{x},\mathbf{x}) & g(\mathbf{x},\mathbf{y}) \\ g(\mathbf{y},\mathbf{x}) & g(\mathbf{y},\mathbf{y}) \end{bmatrix} = \begin{bmatrix} a & b \\ b & c \end{bmatrix}$$

is of full rank and $a, b, c > 0$. But by Lemma 23, g has a vector representation $(s_1, ..., s_{2r})$, such that for all $\mathbf{u} \in \mathrm{supp}(g)$, $g(\mathbf{u}) = s_1(u_1) \cdots s_{2r}(u_{2r})$. Let $s = s_1 \otimes \cdots \otimes s_r$ and $t = s_{r+1} \otimes \cdots \otimes s_{2r}$. Then

$$h = \begin{bmatrix} s(\mathbf{x})t(\mathbf{x}) & s(\mathbf{x})t(\mathbf{y}) \\ s(\mathbf{y})t(\mathbf{x}) & s(\mathbf{y})t(\mathbf{y}) \end{bmatrix},$$

which is singular. A contradiction. ◀

Having shown the equivalence, we turn to consider some properties of balanced sets. There are two basic facts about balance. Later we will often use them but without explicit reference.

▶ **Lemma 25.** *If $\mathcal{F} \subseteq \mathcal{G}$ and $\mathcal{G}$ is balanced, then $\mathcal{F}$ is also balanced.*

▶ **Lemma 26.** *If $f \in \mathsf{S}(\mathcal{F})$ and $\mathcal{F}$ is balanced, then $\mathcal{F} \cup \{f\}$ is also balanced.*

In Boolean #CSP, the two unary functions $[1,0]$ and $[0,1]$ can be simulated [19]. And the function $[1,1]$ is the unary equality function, which is freely available. These unary functions make it more convenient to construct certain functions. But in Holant problems, generally we do not know how to realize or simulate them. Fortunately, we can circumvent this difficulty by the lemma below. It follows from Lemma 28 and Lemma 30.

▶ **Lemma 27.** *If $\mathcal{F}$ is balanced, then the set $\mathcal{F} \cup \{[1,0],[0,1],[1,1]\}$ is balanced.*

▶ **Lemma 28.** *If $\mathcal{F}$ is balanced, then $\mathcal{F} \cup \{[1,0],[0,1]\}$ is balanced.*

▶ **Lemma 29.** *Suppose that $\mathcal{F}$ is a balanced set of non-negative functions. Let f be an n-ary function in $\mathsf{S}(\mathcal{F})$ and let F denote the function f^2. Then for each $t \in [n]$, there exists a constant $\lambda_t > 0$ such that $F^{[t]} = \lambda_t (f^{[t]})^2$.*

Proof. Impose induction on t. The base case $t = n$ is trivial where $\lambda_n = 1$.

Suppose that $F^{[t]} = \lambda_t (f^{[t]})^2$ for $t = k+1 \leq n$. Consider the case $t = k$. For all $\mathbf{x} \in \{0,1\}^k$,

$$F^{[k]}(\mathbf{x}) = F^{[k+1]}(\mathbf{x},0) + F^{[k+1]}(\mathbf{x},1) = \lambda_{k+1} \left[\left(f^{[k+1]}(\mathbf{x},0) \right)^2 + \left(f^{[k+1]}(\mathbf{x},1) \right)^2 \right].$$

Note that the function $F^{[k+1]} \in \mathcal{W}_{\mathcal{F}}$ since $F = f^2 \in \mathcal{W}_{\mathcal{F}}$. Because $\mathcal{F}$ is balanced, $F^{[k+1]}$ is block-rank-one by Lemma 24. Thus the function $f^{[k+1]} = \sqrt{F^{[k+1]}/\lambda_{k+1}}$ is also block-rank-one, which implies that the two column vectors of the matrix $M_{[k]}(f^{[k+1]})$, denoted by $\mathbf{v}_0$ and $\mathbf{v}_1$, are orthogonal or linearly dependent:

- $\mathbf{v}_0$ and $\mathbf{v}_1$ are orthogonal. Then for all $\mathbf{x} \in \{0,1\}^k$,

$$F^{[k]}(\mathbf{x}) = \lambda_{k+1}\left[\left(f^{[k+1]}(\mathbf{x},0)\right)^2 + \left(f^{[k+1]}(\mathbf{x},1)\right)^2\right]$$
$$= \lambda_{k+1}\left(f^{[k+1]}(\mathbf{x},0) + f^{[k+1]}(\mathbf{x},1)\right)^2 = \lambda_{k+1}\left(f^{[k]}(\mathbf{x})\right)^2.$$

- $\mathbf{v}_0$ and $\mathbf{v}_1$ are linearly dependent. Without loss of generality, we assume that $\mathbf{v}_1 = \lambda\mathbf{v}_0$ for some $\lambda \geq 0$. Then for all $\mathbf{x} \in \{0,1\}^k$,

$$F^{[k]}(\mathbf{x}) = \lambda_{k+1}(1+\lambda^2)\left(f^{[k+1]}(\mathbf{x},0)\right)^2 = \lambda_{k+1}\frac{1+\lambda^2}{(1+\lambda)^2}\left(f^{[k]}(\mathbf{x})\right)^2.$$

In either case, the conclusion holds. This completes the induction. ◀

▶ **Lemma 30.** *If $\mathcal{F}$ is balanced, then $\mathcal{F} \cup \{[1,1]\}$ is balanced.*

Proof. Suppose that $[1,1] \notin \mathsf{S}(\mathcal{F})$, otherwise we are done. Let g be an n-ary function in $\mathsf{S}(\mathcal{F} \cup \{[1,1]\})$. We need to show that all the matrices in $\{M_{[r]}(g) \mid 1 \leq r \leq \text{arity}(g)\}$ are block-rank-one.

Let Γ denote the gate that realizes g. If there is an isolated vertex with a dangling edge in Γ, assigned the function $[1,1]$, then we remove this vertex; If there are two adjacent vertices, both assigned the function $[1,1]$, then we delete the pair. Repeat removing until no such vertices. Finally we obtain a new gate Γ'. If Γ' has no dangling edges, then we are done. Suppose not. Let h denote the function that Γ' realizes. And for all $x_1,...,x_n \in \{0,1\}$, $g(x_1,...,x_n) = 2^s h(x_{i_1},...,x_{i_t})$ where $1 \leq i_1 < \cdots < i_t \leq n$ and s denotes the number of pairs we delete. It suffices to prove that the signature matrices of h are all block-rank-one.

Note that $h = f^{[t]}$ for some $f \in \mathsf{S}(\mathcal{F})$ and $1 \leq t \leq \text{arity}(f)$. Let F denote the function f^2. Then by Lemma 29, there is a constant $\lambda_t > 0$ such that $F^{[t]} = \lambda_t(f^{[t]})^2$. Therefore, for any $r \in [t]$, the two matrices $M_{[r]}(f^{[t]})$ and $M_{[r]}(F^{[t]})$ are both block-rank-one or neither. Since $F^{[t]} \in \mathcal{W}_{\mathcal{F}}$, all of its signature matrices are block-rank-one by Lemma 21. Thus every matrix in $\{M_{[r]}(f^{[t]}) \mid 1 \leq r \leq t\}$ is block-rank-one. ◀

With these unary functions, we are able to prove two more lemmas:

▶ **Lemma 31.** *Let $\mathcal{F}$ be a set of non-negative functions and let $g = [1,0,1,0]$. If $\mathcal{F} \cup \{g\}$ is balanced, then $\mathcal{F} \subseteq \mathcal{A}$.*

▶ **Lemma 32.** *Let $\mathcal{F}$ be a set of non-negative functions and let $g = [a,0,...,0,b]$ be a general equality function where* arity$(g) \geq 3$ *and $a,b > 0$. If $\mathcal{F} \cup \{g\}$ is balanced, then $\mathcal{F} \subseteq \mathcal{A}$ or $\mathcal{F} \subseteq \mathcal{P}$.*

7.3 Proof Sketch of Lemma 18

Suppose that a function set $\mathcal{F}$ satisfies the Block-rank-one condition. Then the set $\mathcal{G} = \mathcal{F} \cup \{[1,0],[0,1],[1,1]\}$ is balanced. So it suffices to prove that $\mathcal{G} \subseteq \mathcal{A}$ or $\mathcal{G} \subseteq \mathcal{P}$.

First we consider the case $\mathcal{G} \subseteq \mathcal{T}$. In this case, every nondegenerate binary function in $\mathsf{S}(\mathcal{G})$ has the form $[a,0,b]$ or $(0,a,b,0)$. Thus all of them are of product type. Since the set $\mathcal{P}$ is closed under tensor product, $\mathcal{G} \subseteq \mathcal{P}$.

Now suppose that $\mathcal{G} \not\subseteq \mathcal{T}$. Then there is an irreducible function $f \in \mathsf{S}(\mathcal{G})$ of arity $n \geq 3$. For $1 \leq i < j \leq n$ and $a,b \in \{0,1\}$, we use f_{ij}^{ab} denote the column vector $M_{[n-2]}(f^{x_i=a,x_j=b})$. And we define the $2^{n-2} \times 2^2$ matrices $M_{ij} = (f_{ij}^{00}, f_{ij}^{01}, f_{ij}^{10}, f_{ij}^{11})$.

Since f is irreducible and $\mathcal{G}$ is balanced, any two elements of the support of f differ at two or more bits. Thus we have:

$$\langle f_{ij}^{00}, f_{ij}^{01}\rangle = 0,\ \langle f_{ij}^{00}, f_{ij}^{10}\rangle = 0,$$
$$\langle f_{ij}^{11}, f_{ij}^{01}\rangle = 0,\ \langle f_{ij}^{11}, f_{ij}^{10}\rangle = 0,$$

where $\langle\cdot,\cdot\rangle$ denotes the inner product. Therefore, for every pair (i,j), the 4×4 matrix $B_{ij} = (M_{ij})^{\mathsf{T}} M_{ij}$ has the form

$$\begin{bmatrix} a & 0 & 0 & b \\ 0 & x & y & 0 \\ 0 & y & z & 0 \\ b & 0 & 0 & c \end{bmatrix}.$$

By Cauchy-Schwarz inequality, $ac \geq b^2$ and $xz \geq y^2$. If for all $1 \leq i < j \leq n$, B_{ij} is diagonal, then there exists a function $g = [a, 0, ..., 0, b] \in \mathsf{S}(\mathcal{G})$ where $\text{arity}(g) \geq 3$ and $a, b > 0$. If some B_{ij} is not diagonal, then $B_{ij} = a[1, 0, 1, 0, 1]$ for some $a > 0$ due to the balance of $\mathcal{G}$. In this case, we can further realize the function $a[1, 0, 1, 0]$. According to Lemma 32 or Lemma 31, $\mathcal{G} \subseteq \mathcal{A}$ or $\mathcal{G} \subseteq \mathcal{P}$.

8 Conclusion

To determine the complexity of a problem Holant($\mathcal{F}$), the proofs of previous Holant dichotomies often start with a non-trivial function in $\mathcal{F}$. This works well for symmetric functions, but the structure of an asymmetric one can be very intricate. In [16], we have already seen that asymmetry poses great challenges in arity reduction and gadget construction, even assuming the presence of all unary functions. In fact, similar difficulty arises on higher domains, where it is tough to obtain an explicit dichotomy. The #CSP dichotomies over general domains [23, 8, 7] are more abstract than those over the Boolean domain, but they offer great insights into sum-of-product computation. Inspired by them, we introduce the Block-rank-one condition for Holant problems, which leads to a clear classification. At the beginning of our work, we were not sure whether the condition is sufficient for tractability. Lemma 24 and Lemma 27 make it possible to absorb the results in [19] and reach the destination.

Acknowledgements. The authors are grateful to Jin-Yi Cai for his careful reading of an earlier version of this paper.

References

1 Andrei A. Bulatov. The complexity of the counting constraint satisfaction problem. *J. ACM*, 60(5):34, 2013.

2 Andrei A. Bulatov and Víctor Dalmau. Towards a dichotomy theorem for the counting constraint satisfaction problem. *Inf. Comput.*, 205(5):651–678, 2007.

3 Andrei A. Bulatov, Martin E. Dyer, Leslie A. Goldberg, Markus Jalsenius, Mark Jerrum, and David Richerby. The complexity of weighted and unweighted #CSP. *J. Comput. Syst. Sci.*, 78(2):681–688, 2012.

4 Andrei A. Bulatov, Martin E. Dyer, Leslie A. Goldberg, Markus Jalsenius, and David Richerby. The complexity of weighted Boolean #CSP with mixed signs. *Theor. Comput. Sci.*, 410(38-40):3949–3961, 2009.

5 Andrei A. Bulatov and Martin Grohe. The complexity of partition functions. *Theor. Comput. Sci.*, 348(2):148–186, 2005.

6 Jin-Yi Cai and Xi Chen. A decidable dichotomy theorem on directed graph homomorphisms with non-negative weights. In *Proceedings of 51th Annual IEEE Symposium on Foundations of Computer Science*, pages 437–446, 2010.

7 Jin-Yi Cai and Xi Chen. Complexity of counting CSP with complex weights. In *Proceedings of the 44th Symposium on Theory of Computing*, pages 909–920, 2012.

8 Jin-Yi Cai, Xi Chen, and Pinyan Lu. Non-negatively weighted #CSP: An effective complexity dichotomy. In *Proceedings of the 26th Annual IEEE Conference on Computational Complexity*, pages 45–54, 2011.

9 Jin-Yi Cai, Xi Chen, and Pinyan Lu. Graph homomorphisms with complex values: A dichotomy theorem. *SIAM J. Comput.*, 42(3):934–1029, 2013.

10 Jin-Yi Cai, Zhiguo Fu, Heng Guo, and Tyson Williams. A Holant dichotomy: Is the FKT algorithm universal? In *IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 1259–1276, 2015.

11 Jin-Yi Cai, Heng Guo, and Tyson Williams. A complete dichotomy rises from the capture of vanishing signatures: Extended abstract. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, pages 635–644, 2013.

12 Jin-Yi Cai, Heng Guo, and Tyson Williams. Holographic algorithms beyond matchgates. In *Proceedings of ICALP*, pages 271–282, 2014.

13 Jin-Yi Cai, Sangxia Huang, and Pinyan Lu. From Holant to #CSP and back: Dichotomy for Holant^c problems. *Algorithmica*, 64(3):511–533, 2012.

14 Jin-Yi Cai, Pinyan Lu, and Mingji Xia. Holant problems and counting CSP. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, pages 715–724, 2009.

15 Jin-Yi Cai, Pinyan Lu, and Mingji Xia. Computational complexity of Holant problems. *SIAM J. Comput.*, 40(4):1101–1132, 2011.

16 Jin-Yi Cai, Pinyan Lu, and Mingji Xia. Dichotomy for Holant^* problems of Boolean domain. In *Proceedings of the Twenty-second Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1714–1728, 2011.

17 Jin-Yi Cai, Pinyan Lu, and Mingji Xia. The complexity of complex weighted Boolean #CSP. *J. Comput. Syst. Sci.*, 80(1):217–236, 2014.

18 Nadia Creignou and Miki Hermann. Complexity of generalized satisfiability counting problems. *Inf. Comput.*, 125(1):1–12, 1996.

19 Martin E. Dyer, Leslie A. Goldberg, and Mark Jerrum. The complexity of weighted Boolean #CSP. *SIAM J. Comput.*, 38(5):1970–1986, 2009.

20 Martin E. Dyer, Leslie A. Goldberg, and Mike Paterson. On counting homomorphisms to directed acyclic graphs. *J. ACM*, 54(6), 2007.

21 Martin E. Dyer and Catherine S. Greenhill. The complexity of counting graph homomorphisms. *Random Struct. Algorithms*, 17(3-4):260–289, 2000.

22 Martin E. Dyer and David Richerby. On the complexity of #CSP. In *Proceedings of the 42nd ACM Symposium on Theory of Computing*, pages 725–734, 2010.

23 Martin E. Dyer and David Richerby. An effective dichotomy for the counting constraint satisfaction problem. *SIAM J. Comput.*, 42(3):1245–1274, 2013.

24 Michael Freedman, László Lovász, and Alexander Schrijver. Reflection positivity, rank connectivity, and homomorphism of graphs. *J. Amer. Math. Soc.*, 20(1):37–51, 2007.

25 Leslie A. Goldberg, Martin Grohe, Mark Jerrum, and Marc Thurley. A complexity dichotomy for partition functions with mixed signs. *SIAM J. Comput.*, 39(7):3336–3402, 2010.

26 Heng Guo, Pinyan Lu, and Leslie G. Valiant. The complexity of symmetric Boolean parity Holant problems. *SIAM J. Comput.*, 42(1):324–356, 2013.

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

28 Sangxia Huang and Pinyan Lu. A dichotomy for real weighted Holant problems. In *Proceedings of the 27th Conference on Computational Complexity*, pages 96–106, 2012.

29 Richard E. Ladner. On the structure of polynomial time reducibility. *J. ACM*, 22(1):155–171, 1975.

30 László Lovász. Operations with structures. *Acta Math. Hung.*, 18(3-4):321–328, 1967.

31 Leslie G. Valiant. Accidental algorithms. In *Proceedings of 47th Annual IEEE Symposium on Foundations of Computer Science*, pages 509–517, 2006.

32 Leslie G. Valiant. Holographic algorithms. *SIAM J. Comput.*, 37(5):1565–1594, 2008.