comput. complex. **25** (2016), 255 – 304

1016-3328/16/010255-50
published online November 17, 2015
DOI 10.1007/s00037-016-0118-3
computational complexity

A DICHOTOMY FOR REAL WEIGHTED HOLANT PROBLEMS

SANGXIA HUANG AND PINYAN LU

Abstract. Holant is a framework of counting characterized by local constraints. It is closely related to other well-studied frameworks such as the counting constraint satisfaction problem (#CSP) and graph homomorphism. An effective dichotomy for such frameworks can immediately settle the complexity of all combinatorial problems expressible in that framework. Both #CSP and graph homomorphism can be viewed as subfamilies of Holant with the additional assumption that the equality constraints are always available. Other subfamilies of Holant such as Holant* and Holant^c problems, in which we assume some specific sets of constraints to be freely available, were also studied. The Holant framework becomes more expressive and contains more interesting tractable cases with less or no freely available constraint functions, while, on the other hand, it also becomes more challenging to obtain a complete characterization of its time complexity. Recently, a complexity dichotomy for a variety of subfamilies of Holant such as #CSP, graph homomorphism, Holant*, and Holant^c for Boolean domain was proved. In this paper, we prove a dichotomy for the general Holant framework where all the constraints are real symmetric functions on Boolean inputs. This setting already captures most of the interesting combinatorial counting problems defined by local constraints, such as (perfect) matching. This is the first time a dichotomy is obtained for general Holant problems without any auxiliary functions. One benefit of working with the Holant framework is some powerful new reduction technique such as the holographic reduction. Along the proof of our dichotomy, we introduce a new reduction technique, namely realizing a constraint function by approximating it. This new technique is employed in our proof in a situation where it seems that all previous reduction techniques fail; thus, this new idea of reduction might also be of independent interest. Besides proving a dichotomy and developing a new technique, we also obtained some interesting by-products. We prove a dichotomy for #CSP, restricting to instances where each variable appears a multiple of d times for any d. We also prove that counting the number of Eulerian orientations on $2k$-regular graphs is #P-hard for any $k \geq 2$.

Keywords. Holant, #CSP, counting complexity, computational complexity, dichotomy

Subject classification. 68Q17 Computational difficulty of problems

1. Introduction

In order to study the complexity of counting problems, several interesting frameworks characterizing local properties have been proposed. One is called counting constraint satisfaction problems (#CSP) (Bulatov *et al.* 2009; Bulatov 2006, 2013; Bulatov & Dalmau 2007; Cai *et al.* 2011a; Dyer *et al.* 2009; Dyer & Richerby 2010, 2011; Feder & Vardi 1998). Another well-studied framework is called H-coloring or graph homomorphism, which can be viewed as a special case of #CSP (Bulatov & Grohe 2004, 2005; Cai *et al.* 2013a; Dyer *et al.* 2007; Dyer & Greenhill 2000, 2004; Goldberg *et al.* 2010; Hell & Nešetřil 1990). Recently, inspired by Valiant's holographic algorithms (Valiant 2006, 2008), a new refined framework called Holant problems was proposed (Cai *et al.* 2008, 2009, 2013c). One reason why such frameworks are interesting is because the language is *expressive* enough so that they can express many natural counting problems, while *specific* enough so that we can prove *dichotomy theorems* (i.e., every problem in the class is either in P or #P-hard) (Creignou *et al.* 2001). Having a dichotomy is an important property for these languages. Ladner proved that if P $\neq$ NP, then such a dichotomy for NP is *false* (Ladner 1975). His idea can be easily adapted to show a similar dichotomy for P versus #P. Note that although the counting frameworks mentioned above do admit dichotomy theorems, it does not imply P = NP or P = #P, since the expressive power of these languages is more restricted compared to general Turing machines.

We give a brief description of the Holant framework here, and a more formal definition is given in Section 2. A *signature grid*

$\Omega = (G, \mathcal{F}, \pi)$ is a tuple, where $G = (V, E)$ is an undirected graph and $\mathcal{F}$ is a set of functions. In this paper, we study the case where the functions map sets of Boolean variables to some value. Usually, the range of the functions is the field of complex numbers $\mathbb{C}$ or real numbers $\mathbb{R}$ as in Bulatov (2006, 2013); Bulatov & Dalmau (2007); Cai *et al.* (2008, 2009); Dyer *et al.* (2009); Dyer & Richerby (2010, 2011); Feder & Vardi (1998), but it is also interesting to consider functions with finite range, such as counting the number of solutions modulo some integer k, as studied in Faben (2008); Guo *et al.* (2011); Papadimitriou & Zachos (1982); Valiant (1979, 2010). The mapping $\pi: V \to \mathcal{F}$ labels each $v \in V(G)$ with a function $f_v \in \mathcal{F}$, where the arity of f_v equals the degree of v. We consider all edge assignments (0–1 assignments in this paper, since we are considering functions on Boolean variables). An assignment σ for every $e \in E$ gives an evaluation $\prod_{v\in V} f_v(\sigma \mid_{E(v)})$, where $\sigma \mid_{E(v)}$ denotes the sub-string of σ, where only bits corresponding to incident edges of v are chosen. The counting problem on the instance Ω is to compute

$$\text{Holant}_\Omega = \sum_{\sigma} \prod_{v\in V} f_v \left(\sigma \mid_{E(v)}\right).$$

We use the notation Holant$(\mathcal{F})$ to denote the class of Holant problems where all functions are taken from $\mathcal{F}$. For example, consider the Perfect Matching problem on G. This problem corresponds to attaching the Exact-One function at every vertex of G—for each 0–1 edge assignment, the product $\prod_{v\in V} f_v(\sigma \mid_{E(v)})$ evaluates to 1 when the assignment is a perfect matching, and 0 otherwise; therefore, summing over all 0–1 edge assignments gives us the number of perfect matchings in G. If we use the At-Most-One function at every vertex, then we count all (not necessarily perfect) matchings. This framework can also express the partition function of a system, which is well studied in the statistical physics community, see, for example, the Ising model (Ising 1925).

The Holant framework is closely related to other frameworks such as #CSP and graph homomorphism. In fact, in some sense, the Holant framework provides a unified perspective for different frameworks of counting problems. For example, the #CSP framework can be viewed as a special case of the Holant framework in which equality relations of any arity are always assumed to be available in addition to the stated constraints. In Cai *et al.* (2009), a dichotomy for complex weighted Boolean #CSP was discovered and proved with the help from the study of general Holant problem. On the other hand, #CSP excludes the expression of certain important problems such as graph matchings, which, in contrast, are expressible in the Holant framework. Besides #CSP, another two important special families are Holant* problems, in which we assume that all unary functions are available, and Holant^c Problems, where we only assume two special unary functions—the Is-Zero function **0** and the Is-One function **1**—to be available. For all the above families, a dichotomy for complex symmetric functions was proved (Cai *et al.* 2012, 2009). However, a dichotomy for general Holant family remained open before the current work. The framework becomes more expressive in this general setting and, as we prove in this work, there are more tractable families. On the other hand, the proof for a dichotomy also becomes more challenging. A major source of difficulty is the lack of flexibility when we construct gadgets for reduction. One exception is the dichotomy for the general Holant framework for symmetric function in the field $\mathbb{Z}_2$ (Guo *et al.* 2013). A couple of recent works studied the complexity of Holant on regular graphs where all the vertices take a same function (Cai *et al.* 2012; Cai & Kowalczyk 2010; Cai *et al.* 2011b; Kowalczyk & Cai 2010). These works can be viewed as a dichotomy for Holant without freely available functions, but have the constraint that $\mathcal{F}$ only contains one single function. In these papers, due to the lack of freely available equality functions or unary function, the reductions become much more difficult and even sometimes require assistance from computer (Cai *et al.* 2011b). The underlying goal of these two sequences of works is to finally get a dichotomy for Holant.

This work achieves this final goal at least partially. We prove a dichotomy theorem for Holant problem where all functions are symmetric (the values of the functions only depend on the Hamming weight of their inputs) and take real values. Real symmetric functions already capture most of the interesting combinatorial problems and physical system problems. This is the first dichotomy for the Holant framework for a broad set of functions without assuming any freely available functions. Our work uses previous results as our starting points. And we believe that it is an important step to finally achieve the goal to characterize the complexity of Holant problems for any function set $\mathcal{F}$ (complex weighted and asymmetric).

One of the main innovations in this work is a new way of making reductions between counting problems. In previous works, as well as our current work, there are three extensively used reduction methods: (1) gadget construction, (2) polynomial interpolation, and (3) holographic transformation. However, due to the special structure of some functions, we might be in the case where all possible gadget constructions either give trivial functions or are very difficult to analyze, and thus classical reduction methods might not work well. In Section 5, we introduce a new reduction method—realizing a function by approximating it with sufficient precision. The main idea is to construct a series of gadgets that converge to another gadget extremely fast with only a polynomial overhead, so that we would be able to recover the true value in polynomial time by solving a constant dimension integer programming. Although this is still some kind of gadget construction, we do not, and probably cannot, realize the target function precisely. It is also different from polynomial interpolation in two aspects. One is that although we need to construct a sequence of gadgets in both methods, polynomial interpolation produces an instance for each gadget in the sequence, while the new approximation technique only uses the final one as a close enough approximation. Also, in polynomial interpolation, the number of new instances one constructs is usually linear in the size of the instance, which is not affordable in our construction because the gadget size grows exponentially, while the new approximation technique only needs a logarithmic number of iterations due to the fast convergence rate.

Another contribution of this work is a dichotomy for #CSP where each variable appears a multiple of d times, for any positive integer d. We found some tractable cases which are #P-hard for general #CSP. These new cases are still closely related to the tractable cases for the general #CSP, and we characterize them in terms of holographic transformation.

We also prove that counting Eulerian orientations for $2k$-regular graphs is #P-hard. Note that the notion of Eulerian orientation is different from Eulerian circuits in that the former only considers the direction of the edges and thus different Eulerian circuits may correspond to a same Eulerian orientation. Previously, similar problems have been studied, such as counting Eulerian orientations and counting Eulerian circuits in general graphs (Brightwell & Winkler 2005; Mihail & Winkler 1996), and Eulerian circuits in regular graphs (Ge & Stefankovic 2012). All of them were shown to be #P-hard. However, to the best of our knowledge, there is no previous result on counting Eulerian orientations in regular graphs, and we are not aware of any direct reductions between counting Eulerian orientations and counting Eulerian circuits. Instead, we use polynomial interpolation to reduce the calculation of the Tutte polynomial at certain points to counting Eulerian orientation. The construction is easy to analyze in the Holant framework. One of the intriguing parts of this problem is that it arises as a very special case along our proof for which all reduction methods—including the approximation approach we introduced here—failed. Hence, this problem may also serve as a new starting point of reduction in future research.

This is the complete version of Huang & Lu (2012). After the publication of the conference version of this paper, Cai *et al.* gave an effective dichotomy for complex symmetric Boolean Holant problems in Cai *et al.* (2013b), an important step toward further understanding the nature of Holant and other counting problems. While they do not rely on our dichotomy directly, their proof made important use of some of our results, including the #P-hardness of counting Eulerian orientations, and the dichotomy for #CSP^d. Moreover, it turns out that the problem of counting Eulerian orientations is related to a phenomenon Cai *et al.* refer to as *vanishing*, whose influence on tractable counting problems was never fully realized until the resolution of the whole problem.

2. Preliminaries

In this section, we recall some basic definitions and results. Let $\mathcal{F}$ be a set of functions. A *signature grid* is a tuple $\Omega = (G, \mathcal{F}, \pi)$, where $G = (V, E)$ is an undirected graph and $\pi: V \to \mathcal{F}$ labels each $v \in V$ with a function $f_v \in \mathcal{F}$ where the arity of f_v equals the degree of v. The Holant problem on instance Ω is to compute

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

a sum over all 0–1 edge assignments, of the products of the function evaluations at each vertex. Given a set of functions $\mathcal{F}$, we define the problem Holant($\mathcal{F}$):

- Input: A *signature grid* $\Omega = (G, \mathcal{F}, \pi)$;
- Output: Holant_Ω.

We would like to characterize the complexity of Holant problems in terms of its function set $\mathcal{F}$. In order to simplify notations, we sometimes use Holant(X, Y) to denote Holant$(\{X, Y\})$ and Holant$(X, Y, \mathcal{F})$ for Holant$(\{X, Y\} \cup \mathcal{F})$, where X and Y are some given individual functions and $\mathcal{F}$ is a set of functions.

A function f_v can be represented as a truth table. For functions with complex values, it will be more convenient to denote them as a tensor in $(\mathbb{C}^2)^{\otimes \deg(v)}$, or a vector in $\mathbb{C}^{2^{\deg(v)}}$, when we perform holographic transformations. We also call it a *signature*. Replacing a signature $f \in \mathcal{F}$ by a constant multiple cf, where $c \neq 0$, does not change the complexity of Holant($\mathcal{F}$), so we view f and cf as the same signature. A function f on k Boolean variables is *symmetric* if the value of the function depends only on the number of inputs that are assigned 1 (also known as the Hamming weight of the input) and can be expressed by $[f_0, f_1, \ldots, f_k]$, where f_j is the value of f on inputs of Hamming weight j. Thus, for example, we can express the following unary functions Is-Zero $\mathbf{0} = [1, 0]$, Is-One $\mathbf{1} = [0, 1]$. We denote by $=_k$ the Equality signature of arity k, and we have $(=_k) = [1, 0, \ldots, 0, 1]$ (with $(k-1)$ 0's). The binary disequality can be written as $[0, 1, 0]$. A signature is degenerate iff it is a tensor product of unary signatures, which includes all unary signatures themselves. In particular, a symmetric signature in $\mathcal{F}$ is degenerate iff it can be expressed as $\lambda[x,y]^{\otimes k}$.

Some special families of Holant problems have already been widely studied. For example, if $\mathcal{F}$ contains all EQUALITY signatures $\{=_1, =_2, =_3, \ldots\}$, then this is exactly the weighted #CSP problem, denoted as #CSP($\mathcal{F}$). In Cai *et al.* (2009), the following two special families of Holant problems were introduced by assuming that some signatures are freely available. Let $\mathcal{U}$ denote the set of all unary signatures. Then we define $\text{Holant}^*(\mathcal{F}) = \text{Holant}(\mathcal{F} \cup \mathcal{U})$. We use $\text{Holant}^c(\mathcal{F})$ to denote the problem $\text{Holant}(\mathcal{F} \cup \{\mathbf{0}, \mathbf{1}\})$.

There are several special classes of functions. A symmetric signature $[f_0, f_1, \ldots, f_k]$ is called a *generalized Fibonacci signature* if there exist a, b not both zero such that for all $i = 0, \ldots, k-2$ we have $af_i + bf_{i+1} - af_{i+2} = 0$. A k-ary function $f(x_1, \ldots, x_k)$ is *affine* if there exist a $(k+1)$-column Boolean matrix A, a set of $(k+1)$-dimension Boolean vectors $\{\alpha_1, \ldots, \alpha_n\}$, some complex number $c \neq 0$, such that f can be represented as $c\chi_{AX=0} i^{\sum_{j=1}^{n} \langle \alpha_j, X \rangle}$ where $X = (x_1, x_2, \ldots, x_k, 1)$, $\langle \cdot, \cdot \rangle$ is the dot product of two vectors, i is the imaginary unit with $i^2 = -1$, and χ is a 0–1 indicator function such that $\chi_{AX=0}$ is 1 iff $AX = 0$. Note that both the matrix multiplication AX and the dot product are calculated in $\mathbb{Z}_2$. We use $\mathscr{A}$ to denote the set of all affine functions. We use $\mathscr{P}$ to denote the set of functions which can be expressed as a product of unary functions, binary equality functions, and binary disequality functions. These two families capture exactly tractable #CSP problems.

THEOREM 2.1 (Cai *et al.* 2009). *Let $\mathcal{F}$ be a set of functions mapping Boolean inputs to complex numbers. Then #CSP($\mathcal{F}$) is #P-hard unless $\mathcal{F} \subseteq \mathscr{A}$ or $\mathcal{F} \subseteq \mathscr{P}$, in which case the problem is in P.*

To introduce the idea of holographic reductions, it is convenient to consider bipartite graphs. This is without loss of generality. For any general graph, we can make it bipartite by adding on each edge an additional vertex labeled with the EQUALITY function $=_2$ on two inputs.

We use $\text{Holant}(\mathcal{G}|\mathcal{R})$ to denote all counting problems, expressed as Holant problems on bipartite graphs $H = (U, V, E)$, where each signature for a vertex in U or V is from $\mathcal{G}$ or $\mathcal{R}$, respectively. An input instance for the bipartite Holant problem is a bipartite signature grid and is denoted as $\Omega = (H, \mathcal{G}|\mathcal{R}, \pi)$. Signatures in $\mathcal{G}$ are denoted by column vectors (or contravariant tensors); signatures in $\mathcal{R}$ are denoted by row vectors (or covariant tensors) (Dodson & Poston 1991).

One can perform (contravariant and covariant) tensor transformations on the signatures. We will define a simple version of holographic reductions, which are invertible. Suppose $T \in \mathbf{GL}_2(\mathbb{C})$ is a basis. We say that there is an (invertible) holographic reduction from $\operatorname{Holant}(\mathcal{G}|\mathcal{R})$ to $\operatorname{Holant}(\mathcal{G}'|\mathcal{R}')$, if the *contravariant* transformation $G' = T^{\otimes g} G$ and the *covariant* transformation $R = R'T^{\otimes r}$ map $G \in \mathcal{G}$ to $G' \in \mathcal{G}'$ and $R \in \mathcal{R}$ to $R' \in \mathcal{R}'$, and vice versa, where G and R have arity g and r, respectively. Suppose that there is a holographic reduction from $\operatorname{Holant}(\mathcal{G}|\mathcal{R})$ to $\operatorname{Holant}(\mathcal{G}'|\mathcal{R}')$, mapping signature grid Ω to Ω', then $\operatorname{Holant}_\Omega = \operatorname{Holant}_{\Omega'}$. In particular, for invertible holographic reductions from $\operatorname{Holant}(\mathcal{G}|\mathcal{R})$ to $\operatorname{Holant}(\mathcal{G}'|\mathcal{R}')$, one problem is in P iff the other one is in P, and similarly one problem is #P-hard iff the other one is also #P-hard.

In the study of Holant problems, we will often transfer between bipartite and non-bipartite settings. When this does not cause confusion, we do not distinguish signatures between column vectors (or contravariant tensors) and row vectors (or covariant tensors). Whenever we write a transformation as $T^{\otimes n} F$ or $T\mathcal{F}$, we view the signatures as column vectors (or contravariant tensors); whenever we write a transformation as $FT^{\otimes n}$ or $\mathcal{F}T$, we view the signatures as row vectors (or covariant tensors).

Below we list some known dichotomy results and some useful observations regarding functions. The following lemma gives some necessary conditions for symmetric functions to be in $\mathscr{P}$ and $\mathscr{A}$.

LEMMA 2.2. *Let $f = [f_0, f_1, \ldots, f_k]$ be a real-valued symmetric function. If $f \in \mathscr{P}$, then up to scaling, either there exist $c, r \in \mathbb{R}$, such that $f_i = c \cdot r^i$ for $i = 0, \ldots, k$, or f is a binary disequality, or $f_1 = \cdots = f_{k-1} = 0$. If $f \in \mathscr{A} \backslash \mathscr{P}$, then there exists some $c > 0$, such that $|f_i| \in \{0, c\}$, and that either none of f_i's are zero, or $f_i = 0$ for all odd i's, or $f_i = 0$ for all even i's.*

Proof. We first consider the case when $f \in \mathscr{P}$, that is f can be expressed as a product of unary functions, binary equality functions, and binary disequality functions.

Assume f is not all-zero. If the only inputs that make f nonzero are the all-zero inputs or the all-one inputs, then $f = [a, 0, \ldots, 0, b]$ for some $a, b \in \mathbb{R}$. Now we assume otherwise, and let $y_1, \ldots, y_k \in \{0, 1\}$ be an input such that $f(y_1, \ldots, y_k) \neq 0$, and $y_1, \ldots, y_k$ are not all equal. In other words, the weight of the input $w = \sum y_i$ satisfies $0 < w < k$.

We can assume that each variable is only involved in one unary function, because having multiple unary functions on one variable is equivalent to having the product unary function of them on the same variable.

Case 1. Suppose f is a product of unary functions *only*. That is, for input variable $x_i, i = 1, \ldots, k$, we associate a unary function $[a_{i,0}, a_{i,1}]$, and $f(x_1, \ldots, x_k) = \prod_{i=1}^{k} a_{i,x_i}$. If there is some i such that $a_{i,0} = a_{i,1} = 0$, then f is always zero.

Otherwise, there exists some i, $a_{i,0} = 0$ and $a_{i,1} \neq 0$. Since $f(y_1, \ldots, y_k) \neq 0$, it follows that $y_i = 1$. By our assumption that $0 < w < k$, we also have some $j \neq i$ such that $y_j = 0$. Exchanging the value of y_j and y_i does not change the weight, but the value of the function changes from nonzero to zero, contradicting the hypothesis that f is symmetric. Thus such $y_1, \ldots, y_k$ does not exist and $f_1 = \cdots = f_{k-1} = 0$. The case where there exists some i, $a_{i,0} \neq 0$, $a_{i,1} = 0$ can be handled similarly.

Now suppose for all i, we have $a_{i,0} \neq 0$, $a_{i,1} \neq 0$. After scaling, we may assume that $a_{i,0} = 1$ for all i. The fact that all inputs of weight 1 evaluate to the same value implies that $a_{1,1} = a_{2,1} = \cdots = a_{k,1} = r$ for some $r \neq 0$. Hence $f = [c, cr, cr^2, \ldots, cr^k]$.

Case 2. Now consider the case where there are binary equality and inequality functions in the product. We assume that for each pair of variables, either they are not involved in a binary function, or they are involved in exactly one of binary equality and disequality functions, since if they appear in both functions, then the function is 0 and is thus degenerate.

Suppose f is of arity two. Also assume that the unary function on one of the variables is $[a, b]$, and the one on the other variable is $[c, d]$. If the binary function in the product is a binary equality function, then $f = [ac, 0, bd]$. If the function is a binary disequality, then f evaluates to 0 when the input has weight 0 or 2, and either ad or bc when the input has weight 1. Since f is symmetric, it must be that $ad = bc$, and thus $f = [0, r, 0]$ for some $r \in \mathbb{R}$.

Now assume that f has arity at least 3. If there are contradictions, such as $x_1 \neq x_2$, $x_2 \neq x_3$, and $x_3 \neq x_1$, then we know that the function always evaluates to 0, thus is degenerate. If there are no contradictions, then the set of variables is divided into several subsets, within which variables are restricted to take the same value, and variables in different subsets *may* be required to take different values.

In fact, if f is symmetric and has size at least 3, it must be the case that all variables are in the same set which takes equal values. Otherwise, swapping the values of any pair of variables in $y_1, \ldots, y_k$ that are taking different values (and hence are not in the same set) always makes f zero. We thus conclude that f must have the form $[a, 0, \ldots, 0, b]$ for some $a, b \in \mathbb{R}$.

Now consider $f \in \mathscr{A} \backslash \mathscr{P}$. It follows from definition that there exists some $c > 0$, such that $|f_i| \in \{0, c\}$. Now we focus on the $\chi_{AX=0}$ factor, since we are only interested in whether the values are zero or not.

We first prove that if $f_2 \neq 0$, then $f_i \neq 0$ for all even i's. Let

$$X_1 = (1, 1, 0, \ldots, 0, 1)\,,$$
$$X_2 = (0, 1, 1, 0, \ldots, 0, 1)\,,$$
$$X_3 = (1, 0, 1, 0, \ldots, 0, 1)$$

be $(k+1)$-dimensional Boolean vectors, and they all correspond to inputs of weight 2. Note that $f_2 \neq 0$ implies that $AX_i = 0$ for $i = 1, 2, 3$. This implies that $A(X_1 + X_2 + X_3) = 0$, and since $X_1 + X_2 + X_3 = (0, 0, \ldots, 0, 1)$ corresponds exactly to the input of weight 0, we have that $f_0 \neq 0$. If $k \geq 4$, we can define $Y_1 = (1, 1, 0, \ldots, 0, 1)$, $Y_2 = (0, 1, 1, 0, \ldots, 0, 1)$, and $Y_3 = (0, 1, 0, 1, 0, \ldots, 0, 1)$. $Y_4 \triangleq Y_1 + Y_2 + Y_3 = (1, 1, 1, 1, 0, \ldots, 0, 1)$ is a vector corresponding to some input of weight 4. Then we also have $AY_4 = 0$ and hence $f_4 \neq 0$. Similarly, we can prove that $f_{2t} \neq 0$ for all $2t \leq k$. The proof that $f_1 \neq 0$ implies $f_i \neq 0$ for all odd i's is essentially the same. □

The following theorem is very useful as a way to normalize the given signature set $\mathcal{F}$.

THEOREM 2.3. *Let $\mathcal{F}$ be a set of signatures and M be a 2×2 orthogonal matrix. For any signature grid $\Omega = (G, \mathcal{F}, \pi)$, replacing every signature $F \in \mathcal{F}$ by $M^{\otimes n}F$, where n is the arity of F, we can get a new signature grid Ω'. Then*

$$\text{Holant}_\Omega = \text{Holant}_{\Omega'}.$$

PROOF. First we use the standard technique to reformulate the signature grid $\Omega = (G, \mathcal{F}, \pi)$. We insert a new vertex at each edge of G with signature $=_2$. This will not change the value of the signature grid. Then for the new bipartite signature grid where we have $\mathcal{F}$ on one side and $=_2$ on the other, we apply a holographic reduction with the basis M. This will map a signature $F \in \mathcal{F}$ to $M^{\otimes n}F$, where n is the arity of F. It is an algebraic fact that the $=_2$ will map to itself. Then we view these (new) $=_2$ as an edge and ignore these vertices. This gives the signature grid Ω' as required. Due to the Holant theorem, its value is the same as Ω. □

Prior to this work, dichotomy results were proved in Cai *et al.* (2009) and Cai *et al.* (2012) for $\text{Holant}^*(\mathcal{F})$ and $\text{Holant}^c(\mathcal{F})$ where $\mathcal{F}$ is a set of complex symmetric Boolean signatures. We list the dichotomy for $\text{Holant}^c(\mathcal{F})$ here as it would be used in this paper.

THEOREM 2.4 [Cai *et al.* (2012, 2009)]. *Let $\mathcal{F}$ be a set of complex symmetric signatures.* $\text{Holant}^c(\mathcal{F})$ *is #P-hard unless $\mathcal{F}$ satisfies one of the following conditions, in which case it is tractable:*

(i) Every signature in $\mathcal{F}$ is of arity no more than two;

(ii) There exist two constants a and b (not both zero, depending only on $\mathcal{F}$), such that for every signature $[x_0, x_1, \ldots, x_n] \in \mathcal{F}$ one of the two conditions is satisfied: (1) for every $k = 0, 1, \ldots, n-2$, we have $ax_k + bx_{k+1} - ax_{k+2} = 0$; (2) $n = 2$ and the signature $[x_0, x_1, x_2]$ is of form $[2a\lambda, b\lambda, -2a\lambda]$.

(iii) *For every signature* $[x_0, x_1, \ldots, x_n] \in \mathcal{F}$*, one of the two conditions is satisfied: (1) For every* $k = 0, 1, \ldots, n-2$*, we have* $x_k + x_{k+2} = 0$*; (2)* $n = 2$ *and the signature* $[x_0, x_1, x_2]$ *is of form* $[\lambda, 0, \lambda]$*.*

(iv) *There exists a* $T \in \mathcal{T}$ *such that* $\mathcal{F} \subseteq T\mathscr{A}$*, where* $\mathcal{T} \triangleq \{T \mid [1,0,1]T^{\otimes 2}, [1,0]T, [0,1]T \in \mathscr{A}\}$*.*

Regarding models of computation for real numbers, strictly speaking we should restrict it to computable numbers (Blum *et al.* 1998; Ko 1991), or algebraic numbers. However, this issue seems not essential for our result, and we will state our theorems assuming that we can compute $+$, $\times$ and solve linear equations in polynomial time for all real numbers used. If restricted to algebraic numbers, our proof in Section 5 can be simplified. But we do not restrict our result by exploiting the special properties of algebraic numbers.

3. Main Dichotomy and Proof Outline

For the simplicity of statement, we define the following property for function sets.

Definition 3.1. *A set of signatures* $\mathcal{F}$ *is called* $\mathscr{A}\&\mathscr{P}$-compatible *if there exists a* 2×2 *non-singular complex matrix* T *such that one of the following conditions is satisfied:*

$$\mathcal{F}T^{-1} \subseteq \mathscr{A} \quad \text{and} \quad T^{\otimes 2}[1,0,1]^T \in \mathscr{A}; \text{ or}$$
$$\mathcal{F}T^{-1} \subseteq \mathscr{P} \quad \text{and} \quad T^{\otimes 2}[1,0,1]^T \in \mathscr{P}.$$

Our main theorem is the following.

Theorem 3.2. *Let* $\mathcal{F}$ *be a set of symmetric signatures on Boolean variables with real values. Then* Holant$(\mathcal{F})$ *is #P-hard unless the arity of any non-degenerate signature in* $\mathcal{F}$ *is no more than two or* $\mathcal{F}$ *is* $\mathscr{A}\&\mathscr{P}$*-compatible, in which case it is computable in polynomial time.*

Proof Outline. If the arity of any non-degenerate signature in $\mathcal{F}$ is no more than two, then Holant$(\mathcal{F})$ is obviously tractable.

Otherwise, the tractability of $\mathscr{A}\&\mathscr{P}$-compatible $\mathcal{F}$ follows directly from the tractability of #CSP($\mathscr{A}$) and #CSP($\mathscr{P}$) after applying transformation under T as given in Definition 3.1. Therefore, we only need to prove the hardness part and we can assume that $\mathcal{F}$ contains a non-degenerate signature whose arity is at least 3.

Our starting point is Theorem 4.9, which states that the dichotomy holds if $\mathcal{F}$ contains a non-degenerate ternary function. To prove this, we use the relationship between Holant problems and #CSP. In some cases, we need a dichotomy for special #CSP where variables appear a multiple of 3 times. A general dichotomy for such #CSP is proved in Section 4.

The idea then is to realize a non-degenerate ternary function. In the previous dichotomy for Holant* or Holant^c problems, this step is easy because the freely available functions such as Is-Zero and Is-One enable us to realize sub-signatures with smaller arities. In our case, however, there are no longer freely available unary signatures. We can only use signatures from the given set. Probably the simplest gadget one can construct is by adding self-loops. For a signature with arity k, we can construct a signature with arity $k-2$ by adding a self-loop. If the new signature is degenerate, then it has some very special structure and we can deal with that separately. Otherwise, we have constructed a smaller signature which is still non-degenerate. Repeat this process of adding self-loops, and we will finally have a non-degenerate signature of arity 3 or 4, depending on the parity of k. The ternary case is proved in Theorem 4.9. It is not directly applicable for arity 4 case since we would not be able to construct any signature of odd arity from signatures of arity 4. We handle this in Theorem 7.3.

The idea of proving Theorem 7.3 is to realize degenerate binary signatures. A degenerate binary signature can be viewed as two unary signatures, and in this sense, we can realize a ternary function with the help of this "unary" signature. As stated in Lemma 7.1, we can show that the dichotomy holds if we have a non-degenerate 4-ary signature and one nonzero unary function. Similar to the ternary case, the proof makes use of the relation between Holant and #CSP.

The main remaining work is to realize a nonzero degenerate binary signature. We generalize the polynomial interpolation technique to achieve this. There are cases when this approach fails, and for those cases, we use our new reduction tool of approximating. This is done in Section 5. There is still one exceptional case, namely $[1, 0, 1/3, 0, 1]$. We prove its hardness in Section 6. By holographic reduction, this problem is equivalent to the counting Eulerian orientations problem in 4-regular graphs, which can be proved to be #P-hard. □

REMARK 3.3. *We note that our main dichotomy for Holant is only for real-valued functions. However, the dichotomy for the #CSP where variables appear a multiple of d times is for complex numbers. This is necessary to make it useful in the proof of our main dichotomy. Even starting from real Holant problem, we may come to the field of complex number after some holographic transformation. As mentioned at the end of Section 1, the dichotomy for complex symmetric Boolean Holant problems in Cai et al. (2013b) made important use of the #CSP^d results.*

4. #CSP where variables appear a multiple of d times

In this section, we consider a special family of the #CSP, where the number of occurrences of each variable must be a multiple of d times (d is a given constant). We use #CSP$^d(\mathcal{F})$ to denote this problem. For example, #CSP$^2(\mathcal{F})$ is the #CSP instance where each variable appears an even number of times. Clearly, if #CSP$(\mathcal{F})$ is polynomial time computable, then so is #CSP$^d(\mathcal{F})$. However, the reverse is not necessarily true. We use $\mathcal{T}_d$ to denote the set $\left\{ \begin{bmatrix} 1 & 0 \\ 0 & \omega \end{bmatrix} \mid \omega^d = 1 \right\}$. Then applying any $T \in \mathcal{T}_d$ to $\mathcal{F}$ will not change the value of a #CSP$^d(\mathcal{F})$ instance and as a result will not change the complexity of #CSP$^d(\mathcal{F})$. For example, #CSP$^3([1, \omega_3, -\omega_3^2])$, where ω_3 is the primitive third root of unity, is computable in polynomial time since #CSP$^3([1, 1, -1])$ is. On the other hand, note that by Theorem 2.1, #CSP$([1, \omega_3, -\omega_3^2])$

is #P-hard without the additional constraints on the number of occurrences of variables. For a symmetric function set $\mathcal{F}$, we prove a dichotomy for $\#\mathrm{CSP}^d(\mathcal{F})$ which shows that these are essentially the only new tractable cases.

THEOREM 4.1. *Let $d \geq 1$ be an integer and $\mathcal{F}$ be a set of symmetric functions taking complex values. Then $\#\mathrm{CSP}^d(\mathcal{F})$ is #P-hard unless there exists $T \in \mathcal{T}_{4d}$ such that $(T\mathcal{F}) \subset \mathscr{P}$ or $(T\mathcal{F}) \subset \mathscr{A}$, in which case the problem is in P.*

Note that when $d = 1$, this is just $\#\mathrm{CSP}(\mathcal{F})$, and the above theorem follows from Theorem 2.1 and the fact that the families $\mathscr{A}$ and $\mathscr{P}$ are invariant under transformations in $\mathcal{T}_4$.

Below we assume that $d > 1$.

The following Theorem in Cai *et al.* (2012) gives a reduction between #CSP and Holant, which will be used here as a starting point.

THEOREM 4.2. *Consider the bipartite Holant instance*

$$\mathrm{Holant}\left([1,0,0,1] \cup \mathcal{G}_1 | \mathcal{G}_2\right).$$

We assume that $\mathcal{G}_2$ contains a non-degenerate binary signature $[y_0, y_1, y_2]$. And in the case of $y_0 = y_2 = 0$, we further assume that $\mathcal{G}_2$ contains a unary signature $[a, b]$, where $ab \neq 0$. Then $\mathrm{Holant}([1,0,0,1] \cup \mathcal{G}_1|\mathcal{G}_2)$ is #P-hard unless there exists a $T \in \mathcal{T}_3$ such that $\mathcal{G}_1 T \cup T^{-1}\mathcal{G}_2 \subset \mathscr{P}$ or $\mathcal{G}_1 T \cup T^{-1}\mathcal{G}_2 \subset \mathscr{A}$, in which cases the problem is in P.

Before proving Theorem 4.1, we prove in Lemma 4.3 that the conclusion holds if we have IS-ZERO ($[1,0]$) and IS-ONE ($[0,1]$) in addition. For general #CSP, one can assume freely available $[1,0]$ and $[0,1]$ by the nice pinning lemma from Dyer *et al.* (2009). This is not obviously true for $\#\mathrm{CSP}^d$. We prove in Lemma 4.7 that we can still effectively realize the idea of pinning by a similar idea used in Cai *et al.* (2010). Then Theorem 4.1 follows directly from the following lemma.

LEMMA 4.3. *Let $d > 1$ be an integer and $\mathcal{F}$ be a set of symmetric functions taking complex values. Then $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0],[0,1]\})$*

is #P-hard unless there exists $T \in \mathcal{T}_{4d}$ such that $(T\mathcal{F}) \subset \mathscr{P}$ or $(T\mathcal{F}) \subset \mathscr{A}$, in which case the problem is in P.

PROOF. Before the main part of the proof, we list some simple linear algebra facts which will be useful in the proof. All of them can be verified easily by definition. Let $M = \begin{bmatrix} 1 & 0 \\ 0 & x \end{bmatrix}$ be a non-degenerate *diagonal* matrix. Then, (1) both $[0, 1], [1, 0]$ remain unchanged (up to a scale) after a holographic reduction under M; (2) the property that a signature F is in $\mathscr{P}$ or not remains unchanged after a holographic reduction under M; (3) for any $d \geq 1$, $M \in \mathcal{T}_d \Leftrightarrow M^{-1} \in \mathcal{T}_d$; (4) for any $d \geq 1$, $=_d$ remains unchanged after a holographic reduction under M iff $M \in \mathcal{T}_d$; and (5) for any $d \geq 1$, $M^{\otimes d}(=_d) \in \mathscr{A}$ iff $M \in \mathcal{T}_{4d}$.

We use the following bipartite Holant problem to represent this #CSP$^d(\mathcal{F} \cup \{[1, 0], [0, 1]\})$

$$\operatorname{Holant}\left(\{=_d, =_{2d}, \ldots, \}|\mathcal{F} \cup \{[1, 0], [0, 1]\}\right).$$

We first show the tractability part. Let $T \in \mathcal{T}_{4d}$ be the matrix such that $(T\mathcal{F}) \subset \mathscr{P}$ or $(T\mathcal{F}) \subset \mathscr{A}$. Applying a holographic reduction on the above problem under basis T^{-1}, we have

$$\begin{aligned} &\operatorname{Holant}\left(\{=_d, =_{2d}, \ldots, \}\,|\mathcal{F} \cup \{[1, 0], [0, 1]\}\right) \\ &\quad \equiv_T \ \operatorname{Holant}\left(\{=_d, =_{2d}, \ldots, \}\,T^{-1}|\,(T\mathcal{F}) \cup \{[1, 0], [0, 1]\}\right). \end{aligned}$$

Since $\{=_d, =_{2d}, \ldots, \}T^{-1} \subset \mathscr{P} \cap \mathscr{A}$, we have that either all the signatures involved in the above Holant problem are in $\mathscr{P}$ or all the signatures involved in the above Holant problem are in $\mathscr{A}$. The polynomial-time algorithm follows directly from that.

Now we prove the hardness part. We can actually also realize $\{[1, 0], [0, 1]\}$ on the LHS by connecting the $[1, 0]$'s and $[0, 1]$'s on the RHS to the equality signatures, and by this we can realize any sub-signature on the RHS. If all the binary sub-signatures of signatures in $\mathcal{F}$ are degenerate, then $\mathcal{F} \subset \mathscr{P}$ and we are done. Now we assume that we can realize a non-degenerate binary $[y_0, y_1, y_2]$ on the RHS.

Let $f := [f_0, f_1, \ldots, f_r]$ be a function in $\mathcal{F}$. If there exists some $i \in \{0, \ldots, r-1\}$ such that $f_i f_{i+1} \neq 0$, then we use $[1, 0]$ and

$[0,1]$ on the LHS to realize $[f_i, f_{i+1}]$. After a scale, we can write it as $[1,a]$, where $a \neq 0$. By connecting $3d-3$ copies of $[1,a]$ to a $(=_{3d})$ (note that we have $3d > 3$), we can realize $[1,0,0,a^{3d-3}]$ on the LHS. Then we can apply a holographic reduction under $M = \begin{bmatrix} 1 & 0 \\ 0 & a^{-(d-1)} \end{bmatrix}$, which transforms $[1,0,0,a^{3d-3}]$ into $[1,0,0,1]$. Note that we have a non-degenerate binary signature on the RHS and also a unary $[x,y] = M[1,a]$ with $xy \neq 0$. We conclude by Theorem 4.2 that the problem is #P-hard unless there exists an $N(= MT)$, where $T \in \mathcal{T}_3$, such that $(\{=_d, =_{2d}, \ldots, \}N) \cup (N^{-1}\mathcal{F}) \subset \mathscr{P}$ or $(\{=_d, =_{2d}, \ldots, \}N) \cup (N^{-1}\mathcal{F}) \subset \mathscr{A}$. We note that N is a diagonal matrix. If $N^{-1}\mathcal{F} \subset \mathscr{P}$, then $\mathcal{F} \subset \mathscr{P}$ and we are done. Otherwise, $\{=_d, =_{2d}, \ldots, \}N \cup N^{-1}\mathcal{F} \subset \mathscr{A}$. The fact that $(=_d)N \in \mathscr{A}$ directly implies that $N \in \mathcal{T}_{4d}$ and the proof for this case is also complete.

Consider now all signatures and sub-signatures of $\mathcal{F}$. By the argument above, we assume that all unary signatures and sub-signatures are of form either $[1,0]$, $[0,1]$, or $[0,0]$. This also rules out degenerate signatures of the form $[a, ar, ar^2, \ldots]$ where $a, r \neq 0$. For a binary signature $[a,b,c]$, by $f_i f_{i+1} = 0$, we have that if $b \neq 0$, then $a = c = 0$, so binary signatures must have form either $[a,0,c]$ or $[0,b,0]$. Now we turn to signatures and sub-signatures of arity at least 3. If all of them are of the form $[a,0,\ldots,0,b]$, then $\mathcal{F} \subseteq \mathscr{P}$ and we are done. Otherwise, we can find a signature or sub-signature of arity at least 3 which is not of the form $[a,0,\ldots,0,b]$. Denote it as $f = [f_0, f_1, \ldots, f_r]$. This means that there exists $0 < i < r$, $f_i \neq 0$. We also have that $f_{i-1} = f_{i+1} = 0$. Since f has arity at least 3, either $i-2 \geq 0$ or $i+2 \leq r$. So we can find a (sub-)signature of form either $[b,0,a,0]$ or $[0,a,0,b]$ where $a \neq 0$.

So now we have $f_1 \neq 0$ or $f_2 \neq 0$. By symmetry, we can assume that $f_1 \neq 0$. Then $f_0 = f_2 = 0$ and the signature is of form $[0,1,0,a]$ after scaling. If $a = 0$, we can prove #P-hardness as follows. We use one copy of $=_d$ on the LHS to connect d different copies of $[0,1,0,0]$. Then we group the other $2d$ inputs into two groups as in Figure 4.1. We claim that we can effectively reduce the #P-hard problem #CSP($[1,1,0]$) to it— given a #CSP($[1,1,0]$) instance, we construct an instance of #CSP^d($[0,1,0,0]$) by replacing $=_k$ with $=_{dk}$, $[1,1,0]$ with the gadget in Figure 4.1. Each

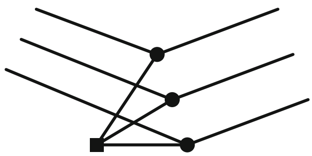

Figure 4.1: The circle vertices has signature $[0,1,0,0]$, and the square vertex is an equality function (we use $=_3$ as example here; in general, it is the corresponding $=_d$).

group of d inputs of Figure 4.1 would be connected to d inputs of the $=_{dk}$ at the corresponding vertex. This construction forces the d edges in the same group in Figure 4.1 to take the same value. Observe that if all $2d$ inputs take value 0, then all edges connected to the $=_d$ in Figure 4.1 need to be 1 to make the overall value of the assignment nonzero. Similarly, if one group of inputs take value 0 and the other take value 1, then the $=_d$ edges need to be 0. And if all $2d$ inputs take value 1, the overall value of the assignment would always be 0 no matter what value the $=_d$ edges take. Thus given an assignment to the #CSP($[1,1,0]$) instance that has nonzero value, we assign the same value to the variables $=_{dk}$ in #CSP^d($[0,1,0,0]$), and there is exactly one way to complete this assignment to get nonzero value, and vice versa. This gives a bijection between edge assignments that have nonzero value on the #CSP($[1,1,0]$) instance and the #CSP^d($[0,1,0,0]$) instance. This completes the reduction.

If $a \neq 0$, then we can realize $[1,0,a]$ ($a \neq 0$) on the RHS. If d is odd, we can connect $\frac{3d-3}{2}$ copies of $[1,0,a]$ to $=_{3d}$ to realize $[1,0,0,a^{\frac{3d-3}{2}}]$ on the LHS. We apply a suitable *diagonal* holographic transformation to make it into $[1,0,0,1]$ and apply Theorem 4.2. We note that this time we may not have a suitable unary function on the RHS. But since we are applying a *diagonal* transformation, $[1,0,a]$ becomes $[1,0,a']$ for some $a' \in \mathbb{R}$, $a' \neq 0$, so we can still apply Theorem 4.2. This completes the proof for the case where d is odd.

Now assume that d is even. Suppose further that there exists a non-degenerate signature of form $[a,0,0,0,\ldots,0,b]$, $a,b \neq 0$, with odd arity on the RHS. Let r be the arity. Then by connecting all its inputs to an equality $=_{dk}$ on the LHS where $dk > r$, we can realize a signature of odd arity $dk - r$ on the LHS. Connecting $[1, 0, a]$'s to it, again we realize a signature of the form $[c, 0, 0, d]$ on the LHS and we complete the proof similarly.

Now we consider the case that d is even, and that all non-degenerate signatures of form $[a, 0, \ldots, 0, b]$, $a, b \neq 0$ are of even arity. We reduce #CSP$([0, 1, 0, a^d])$ to #CSP$^d([0, 1, 0, a])$. Given an instance of #CSP$([0, 1, 0, a^d])$, we construct a #CSP$^d([0, 1, 0, a])$ instance by replacing $=_k$ on the LHS with $=_{dk}$, and for each vertex on the RHS with $[0, 1, 0, a^d]$, we introduce d vertices with $[0, 1, 0, a]$ and connect them to the respective LHS neighbors. Therefore, #CSP$^d([0, 1, 0, a])$ is #P-hard unless $[0, 1, 0, a^d] \in \mathscr{P} \cup \mathscr{A}$. Since $[0, 1, 0, a^d] \notin \mathscr{P}$, we conclude that $[0, 1, 0, a^d] \in \mathscr{A}$, which implies that $a^d = \pm 1$. By applying a holographic reduction under basis $T = \begin{bmatrix} 1 & 0 \\ 0 & a^{\frac{1}{2}} \end{bmatrix}$, we can transform $[0, 1, 0, a]$ on the RHS to $[0, 1, 0, 1]$. We note that $T \in \mathcal{T}_{4d}$ since $(a^{\frac{1}{2}})^{4d} = (\pm 1)^{2d} = 1$. All the $=_{4kd}$ in the LHS will remain unchanged. We realize $[1, 0, 1]$ from $[0, 1, 0, 1]$ and then connect it to $=_{4kd}$s to realize all of $\{=_2, =_4, =_6 \ldots, \}$ on the LHS. Since we have $=_2$ in both sides now, we can reduce the following non-bipartite Holant problem to the original problem

$$\text{(4.4)} \quad \mathrm{Holant}(\{=_2, =_4, =_6 \ldots, \} \cup T\mathcal{F} \cup \{[1, 0], [0, 1], [0, 1, 0, 1]\}).$$

So it is enough to show that this problem is #P-hard unless we have $(T\mathcal{F}) \subset \mathscr{A}$.

Before we proceed, we highlight the two assumptions we have at this point: 1) If there is a non-degenerate signature f of form $[a, 0, 0, \ldots, 0, b]$, $a, b \neq 0$ in $\mathcal{F}$, then f has even arity; 2) all ternary non-degenerate (sub-)signatures in $\mathcal{F}$ have form $[b, 0, a, 0]$ or $[0, a, 0, b]$ where $a \neq 0$.

We first show that if there is a signature of form $[a, 0, \ldots, 0, b]$ which is not in $\mathscr{A}$, then the problem is #P-hard. After a scale, we can write it as $[1, 0, 0, 0, \ldots, 0, b]$, where $b \neq 0$. Let its arity be $2k$, and we can realize $[1, 0, b]$ by connecting it to an equality function $=_{2k-2}$. Similar to the above, we group signatures together to achieve reduction from general #CSP. The difference is that here we group different signatures with the same arity. Specifically, given an instance of $\#\mathrm{CSP}([0,1,0,1],[1,0,b])$, we construct an instance of (4.4) by replacing $=_k$ with $=_{2k}$, and for each vertex on the RHS, we make two copies of it. If the vertex is labeled $[0,1,0,1]$, then the new vertices are both labeled $[0,1,0,1]$. If the vertex is labeled $[1,0,b]$, then one of the new vertices is labeled $[1,0,b]$ and the other is labeled $[1,0,1]$. In this way, we reduce the problem $\#\mathrm{CSP}([0,1,0,1],[1,0,b])$ to (4.4), and since $[0,1,0,1] \in \mathscr{A}\backslash\mathscr{P}$, we conclude that (4.4) is #P-hard if $b^4 \neq 1$. Now we prove the same result for signatures of form $[b,0,a,0]$ or $[0,a,0,b]$. We can scale $[0,a,0,b]$ to $[0,1,0,b]$. By grouping one copy of $[0,1,0,b]$ and one copy of $[0,1,0,1]$, we conclude that the problem is #P-hard unless $[0,1,0,b] \in \mathscr{A}$, which implies that $b = \pm 1$. Since we can construct $[0,1,0]$ from $[0,1,0,1]$ and $[1,0]$, the result also holds for $[b,0,a,0]$ because connecting $[0,1,0]$'s to all inputs of $[b,0,a,0]$ gives a $[0,a,0,b]$. Extending this result for ternary sub-signatures to general signatures, we conclude that either the whole signature is in $\mathscr{A}$ or we can construct a longer sub-signature that are multiples of $[1,0,1,0,-1]$ or $[1,0,-1,0,-1]$ after scale. These two cases are also symmetric, so it remains to prove that

$$\text{Holant}\left([1,0,1,0,-1], =_2, =_4, =_6, \ldots\right) \tag{4.5}$$

is #P-hard. We define the following matrix notation for the signature $[1,0,1,0,-1]$: $A = \begin{bmatrix} 1&0&0&1\\ 0&1&1&0\\ 0&1&1&0\\ 1&0&0&-1 \end{bmatrix}$. Now we connect two inputs of two $[1,0,1,0,-1]$s together to form a chain. We can calculate that A^3 is $\begin{bmatrix} 2&0&0&2\\ 0&4&4&0\\ 0&4&4&0\\ 2&0&0&-2 \end{bmatrix}$. Let $A' = \begin{bmatrix} 4&0&0&4\\ 0&16&16&0\\ 0&16&16&0\\ 4&0&0&4 \end{bmatrix}$ be the Hadamard product of A^3 with itself. Given a $\#\mathrm{CSP}(A')$ instance, we can simulate it with $\#\mathrm{CSP}^2(A^3)$ (and hence (4.5)) via grouping as the following: For each variable in $\#\mathrm{CSP}(A')$, double the number of appearances in $\#\mathrm{CSP}^2(A^3)$, and for each constraint in $\#\mathrm{CSP}(A')$, add two constraints of A^3 connecting to the same variable. We see that A' is not in $\mathscr{A}$ because the nonzero entries are not of the same norm, and it is not in $\mathscr{P}$ because no two inputs are always equal or always different, and it is clearly not a product of only unary signatures. Therefore the problem is #P-hard and so is (4.5). This completes the proof. □

We now prove the pinning lemma for $\#\mathrm{CSP}^d$. To remove $[0,1], [1,0]$ in Lemma 4.3, we start with the following special pinning lemma. Similar to Dyer *et al.* (2009), we have the following claim.

CLAIM 4.6. $\#\mathrm{CSP}^d(\mathcal{F}) \equiv_T \#\mathrm{CSP}^d\left(\mathcal{F} \cup \left\{[1,0]^{\otimes d}, [0,1]^{\otimes d}\right\}\right)$.

The proof is exactly the same as in Dyer *et al.* (2009) so we omit it here. The only thing one need to notice is that when adding auxiliary variables, it is important that it appears a multiple of d times, and in our case this is guaranteed by $[1,0]^{\otimes d}$ and $[0,1]^{\otimes d}$. The following lemma shows that we can effectively realize pinning. A similar idea was used in Cai *et al.* (2010).

LEMMA 4.7. *The problem* $\#\mathrm{CSP}^d(\mathcal{F})$ *is #P-hard (or in P) if and only if the problem* $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0],[0,1]\})$ *is #P-hard (or in P).*

PROOF. Obviously, the first one can be reduced to the second one. Hence if the second problem is in P, so is the first. We have already proved a dichotomy theorem for the second one Lemma 4.3. So now we may assume the second problem is #P-hard and show that the first problem is also #P-hard.

We observe that in all the proofs in this paper and in Cai *et al.* (2013c), when we prove the second problem to be #P-hard for any signature set, we reduce one of the following three problems to it by a chain of reductions: (a) $\mathrm{Holant}([1,0,0,1]|[1,1,0])$, (b) $\mathrm{Holant}([1,1,0,0])$, or (c) $\mathrm{Holant}([0,1,0,0])$ (VERTEX COVER or MATCHING or PERFECT MATCHING for 3-regular graph). There are only three reduction methods in this reduction chain: direct gadget construction, polynomial interpolation, and holographic reduction.

Given an instance G of either of the three Holant problems listed above, we consider the graph $G^{\otimes d}$, which denotes the disjoint union of d copies of G.

Notice that the value of the above three Holant problems on the instance G is a nonnegative integer. Denote that value by $val(G)$. Then the value on $G^{\otimes d}$ is exactly $val(G^{\otimes d}) = val(G)^d$. So we can compute the value on G uniquely from its dth power. Suppose the reduction chain on the instance G produced instances $G_1, G_2, \ldots, G_m$ of the second problem. The same reduction applied to $G^{\otimes d}$ produces instances of the form $G_1^{\otimes d}, G_2^{\otimes d}, \ldots, G_{m'}^{\otimes d}$ (we note that in the case of polynomial interpolation, although $val(G^{\otimes d})$ is a polynomial is of higher degree, the number of unknown coefficients we need to solve is the same and thus the number of instances we need for the reduction remains unchanged).

For each $G_i^{\otimes d}$ as an instance of $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0],[0,1]\})$, the number of occurrences of $[0,1]$ or $[1,0]$ is a multiple of d. Hence, we can view it as an instance of $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0]^{\otimes d}, [0,1]^{\otimes d}\})$. By the assumption that $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0],[0,1]\})$ is hard, we conclude that $\#\mathrm{CSP}^d(\mathcal{F} \cup \{[1,0]^{\otimes d}, [0,1]^{\otimes d}\})$ is #P-hard. By Claim 4.6, we have that $\#\mathrm{CSP}^d(\mathcal{F})$ is #P-hard. □

This completes the proof of Theorem 4.1.

In Cai *et al.* (2012), we proved the following dichotomy for single ternary signature. Note that we omitted one additional tractable case here since it cannot happen for real-valued signatures.

THEOREM 4.8. *Let $[x_0, x_1, x_2, x_3]$ be a real non-degenerate signature. Then* Holant$([x_0, x_1, x_2, x_3])$ *is #P-hard unless there exists a 2×2 matrix T such that $[x_0, x_1, x_2, x_3] = T^{\otimes 3}[1,0,0,1]$ and $[1,0,1]T^{\otimes 2}$ is in $\mathscr{A} \cup \mathscr{P}$.*

We now prove that our main dichotomy result (Theorem 3.2) holds if $\mathcal{F}$ contains a non-degenerate ternary signature.

THEOREM 4.9. *Let $\mathcal{F}$ be a set of real symmetric signatures and X be a real symmetric non-degenerate ternary signature. Then* Holant$(X, \mathcal{F})$ *is #P-hard unless $\mathcal{F} \cup \{X\}$ is $\mathscr{A}$&$\mathscr{P}$-compatible, for which case there is a polynomial-time algorithm.*

PROOF. If Holant(X) is #P-hard according to Theorem 4.8, then we are done. Otherwise, we take T as guaranteed in Theorem 4.8, and we have the following by applying holographic reduction

$$\text{Holant}(X, \mathcal{F}) \equiv_T \text{Holant}([1,0,0,1], T^{-1}\mathcal{F} | [1,0,1]T^{\otimes 2}).$$

We also have that $[1,0,1]T^{\otimes 2}$ is non-degenerate. If $[1,0,1]T^{\otimes 2}$ is not of form $[0,\lambda,0]$, we are done by Theorem 4.2. Otherwise, we have $[0,1,0]$ on the RHS. This enables us to realize $=_k$ on the RHS whenever $=_k$ is available on the LHS: Simply connect each of the k inputs of a $=_k$ on the LHS to one of the inputs of k different copies of $[0,1,0]$'s on the RHS. We can realize all equalities of arity $3k$ as the following. First, construct $[1,0,0,1]$ on the RHS as described above. Once we have a $=_{3k}$ on the RHS, we can connect $3k$ copies of $[1,0,0,1]$ to it and realize a $=_{6k}$ on the LHS. For $6k-3(k+1) = 3(k-1)$ of the inputs of $=_{6k}$, partition them arbitrarily into $(k-1)$ groups of 3 inputs and connect the inputs within the same group to a $[1,0,0,1]$ on the RHS. This gives us $=_{3(k+1)}$ on the LHS. Repeating this procedure, we can realize $=_{3k}$ for any k. Then we can view it as a #CSP^3 problem and we are done by Theorem 4.1. □

For a signature with arity larger than 3, it is not necessarily true that we can transform it to $[1,0,0,0,1]$ by holographic reduction. But for some special signatures, we can. Formally, we have the following two corollaries.

COROLLARY 4.10. *Let X be a real non-degenerate generalized Fibonacci signature of arity no less than 3 and $\mathcal{F}$ be a set of symmetric signatures. Then* Holant$(\mathcal{F}, X)$ *is #P-hard unless $\mathcal{F} \cup \{X\}$ is $\mathscr{A}$&$\mathscr{P}$-compatible, for which case there is a polynomial-time algorithm.*

PROOF. Let k be the arity of signature X. For a real non-degenerate generalized Fibonacci signature, there is an orthogonal holographic reduction that transforms it into the form of $[1,0,0,\ldots,0,a]$ where $a \neq 0$ after scale. By Theorem 2.3, we assume that this is already the case with X. If the arity of X is odd, we realize $[1,0,0,a]$ by adding some self-loops and then apply Theorem 4.9. If the arity is even, we connect two copies of the signature using half of their dangling edges to realize $[1, 0, 0, \ldots, 0, a^2]$ with same arity. Keep doing this, and we can realize $[1, 0, 0, \ldots, 0, a^t]$ for any t. If a is a pth root of unity, we get $[1, 0, 0, \ldots, 0, 1]$ of arity k by choosing $t = p$. Otherwise, we get $[1, 0, 0, \ldots, 0, 1]$ by interpolation. Having $[1, 0, 0, \ldots, 0, 1]$ of even arity k, we can realize all equality functions with an even arity and the result follows from Theorem 4.1. □

COROLLARY 4.11. *Let $x, y \in \mathbb{C}$, and*

$$X = [x, y, -x, -y, x, y, -x, -y \ldots]$$

be a non-degenerate signature of arity $k \geq 3$ and $\mathcal{F}$ be a set of symmetric signatures. Then Holant$(\mathcal{F}, X)$ *is #P-hard unless $\mathcal{F} \cup \{X\}$ is $\mathscr{A}$&$\mathscr{P}$-compatible, for which case there is a polynomial-time algorithm.*

PROOF. Rewrite Holant$(\mathcal{F}, X)$ as Holant$(\mathcal{F}, X | =_2)$. Applying a holographic reduction under basis $Z = \begin{bmatrix} A & Bi \\ Ai & B \end{bmatrix}$ with suitable A and B, we can make X into $=_k$ where k is the arity of X and $=_2$ on the RHS into $[0, 1, 0]$. With $[0, 1, 0]$ on the RHS, we can realize all the equality functions whose arities are multiples of k on the LHS. Then we can apply Theorem 4.1 for #CSP^k. □

5. Realizing a signature by approximating it

In this section, we study Holant$([1, a, b, -a, 1] \cup \mathcal{F})$, an important case in the proof of Theorem 7.3. We first normalize the signature $[1, a, b, -a, 1]$ by some orthogonal transformation.

CLAIM 5.1. *There exists a real orthogonal 2×2 matrix Q, such that either $[1, a, b, -a, 1]Q^{\otimes 4} = c[1, 0, b', 0, 1]$ for some $b', c \in \mathbb{R}$, or $[1, a, b, -a, 1]Q^{\otimes 4} = [0, 0, b', 0, 0]$ for some $b' \in \mathbb{R}$.*

PROOF. Consider the following orthogonal matrix

$$Q = \begin{bmatrix} r & \sqrt{1-r^2} \\ \sqrt{1-r^2} & -r \end{bmatrix}, \text{ for some } r, |r| \leq 1,$$

and the 4-ary signature $X = [1, a, b, -a, 1]$ as a vector in $\mathbb{R}^{2^4}$

$$X = (1, a, a, b, a, b, b, -a, a, b, b, -a, b, -a, -a, 1)\,.$$

Consider transforming X under Q, and we have

$$XQ^{\otimes 4} = \begin{matrix} (d', & a', & a', b', a', & b', \\ b', & -a', a', b', b', & -a', b', -a', -a', d'). \end{matrix}$$

where

$$\begin{aligned} d' &= -4ar\sqrt{1-r^2} + 8ar^3\sqrt{1-r^2} - 2br^4 + 2br^2 \\ &\quad -4br^4 + 4br^2 + 2r^4 - 2r^2 + 1 \\ &= 4ar\sqrt{1-r^2}(2r^2-1) - 2r^2(r^2-1)(3b-1) + 1 \\ a' &= -a(8r^4 - 8r^2 + 1) - r\sqrt{1-r^2}(2r^2-1)(3b-1) \\ b' &= 4ar\sqrt{1-r^2} - 8ar^3\sqrt{1-r^2} + 2br^4 - 2br^2 + b \\ &\quad +4br^4 - 4br^2 - 2r^4 + 2r^2 \\ &= -d' + 1 + b. \end{aligned}$$

We have that $a' = -a$ when $r = 0$ or $r = 1$. Also, when $r = 1/\sqrt{2}$, $a' = a$. Thus by continuity of the expression of a', there must exist some r such that $a' = 0$. If for that r we also have $d' \neq 0$, then we are in the first case; otherwise, we get the second case. □

Claim 5.1 converts an instance of Holant$([1, a, b, -a, 1] \cup \mathcal{F})$ to one of either Holant$([1, 0, b', 0, 1] \cup (\mathcal{F}Q))$ or Holant$([0, 0, 1, 0, 0] \cup (\mathcal{F}Q))$ with the same underlying graph. We prove in Claim 5.16 that the latter is #P-hard. In the following, we simply assume that we are given Holant$([1, 0, b, 0, 1] \cup \mathcal{F})$. If $b \in \{0, 1, -1\}$, we are done by Corollary 4.10 and Corollary 4.11. For $b \notin \{0, 1, -1\}$, we will prove that Holant$([1, 0, b, 0, 1])$ is #P-hard, and these together give the following main lemma of this section.

LEMMA 5.2. *Let $X = [1, a, b, -a, 1]$ be a non-degenerate signature. Then* Holant$(\mathcal{F}, X)$ *is #P-hard unless $\mathcal{F} \cup \{X\}$ is $\mathscr{A}$&$\mathscr{P}$-compatible, for which case there is a polynomial-time algorithm.*

In the remaining part of this section, we prove the following result on the hardness of Holant$([1, 0, b, 0, 1])$.

Lemma 5.3. *If $b \notin \{0, 1, -1\}$, then Holant$([1, 0, b, 0, 1])$ is #P-hard.*

To prove this lemma, observe that if we can realize $[1, 0, 0, 0, 1]$, then we can use it to simulate $\#CSP^2([1, 0, b, 0, 1])$, which is #P-hard by Theorem 4.1 and the fact $b \notin \{0, 1, -1\}$. If we can realize $[1, 0, 1, 0, 1]$, then we can apply orthogonal transformation under $\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ to convert $[1, 0, 1, 0, 1]$ to $[1, 0, 0, 0, 1]$ and $[1, 0, b, 0, 1]$ to $[2 + 6b, 0, 2 - 2b, 0, 2 + 6b]$. To see that $[2 + 6b, 0, 2 - 2b, 0, 2 + 6b]$ is among the hard cases in Theorem 4.1, note that $|2 + 6b| \neq |2 - 2b|$ if $b \notin \{0, 1, -1\}$, so it cannot be transformed into $\mathscr{A}$ by $T \in \mathcal{T}_4$. It is also not hard to verify that it cannot be transformed into any signature in $\mathscr{P}$ by $T \in \mathcal{T}_4$.

In the following, we introduce two new techniques for realizing special signatures $[1, 0, 0, 0, 1]$ or $[1, 0, 1, 0, 1]$. First, we generalize the widely used interpolation technique to enable us to interpolate 4-ary signatures instead of unary signatures in the traditional setting. This generalization is already powerful enough for almost all b. The failed b are roots of some non-trivial integer coefficient polynomials and thus must be algebraic numbers. For transcendental numbers, we have the following lemma.

Lemma 5.4. *For any transcendental real number b, we have*

$$\text{Holant}\left([1, 0, b, 0, 1]\right) \equiv_T \text{Holant}\left([1, 0, b, 0, 1], (=_4)\right).$$

Proof. The idea of the proof is similar to the interpolation of unary signatures. We first show how to use polynomial interpolation in a slightly more general setting and then apply the result to Holant$([1, 0, b, 0, 1])$.

Consider the following matrix

$$Q = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix}.$$

Note that we can express a 4-ary signature as a 4×4 matrix. Consider signature $B = Q^{-1} diag(1, x, y, 0) Q$, where $diag(\cdot, \cdot, \cdot, \cdot)$

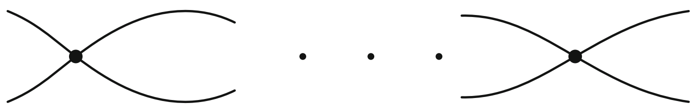

Figure 5.1: Recursive gadget. Nodes are assigned signature represented by B.

denotes the diagonal matrix of size 4 with corresponding elements on the diagonal. We have

$$B = \frac{1}{2}\begin{bmatrix} 1+x & & & 1-x \\ & y & y & \\ & y & y & \\ 1-x & & & 1+x \end{bmatrix}.$$

We construct 4-ary gadgets as in Figure 5.1. Then the signature of the construction with k copies of B is

$$\begin{aligned} B_k &= Q^{-1} diag(1, x^k, y^k, 0) Q \\ &= \frac{1}{2}\begin{bmatrix} 1+x^k & & & 1-x^k \\ & y^k & y^k & \\ & y^k & y^k & \\ 1-x^k & & & 1+x^k \end{bmatrix}. \end{aligned}$$

Consider an instance Ω with n copies of B. Then there exists $\{u_{ij}\}_{i,j\in\{0,\dots,n\}}$, $\{v_{ij}\}_{i,j\in\{0,\dots,n\}}$, such that we can write the Holant value as

$$\begin{aligned} \text{Holant}_\Omega &= \sum_{\substack{i,j\in\{0,\dots,n\} \\ i+j\le n}} u_{ij}(1+x)^i(1-x)^j y^{n-i-j} \\ &= \sum_{i,j\in\{0,\dots,n\}} v_{ij} x^i y^j. \end{aligned}$$

By replacing B with B_k, we get a series of new instances Ω_k with Holant value

$$\text{Holant}_{\Omega_k} = \sum_{i,j\in\{0,\dots,n\}} v_{ij} x^{ik} y^{jk}.$$

Let β be the column vector

$$\beta = (v_{00}, v_{01}, v_{02}, \ldots, v_{0n}, v_{10}, \ldots, v_{nn})^T ,$$

and let α_k be the row vector

$$\alpha_k = \left(1, y^k, y^{2k}, \ldots, y^{kn}, x^k, x^k y^k, \ldots, x^{kn} y^{kn}\right),$$

and we have that $\alpha_k \beta = \mathrm{Holant}_{\Omega_k}$. Similar to interpolation of unary signatures, we view v_{ij} as variables and get the following

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{(n+1)^2} \end{bmatrix} \beta = \begin{bmatrix} \mathrm{Holant}_{\Omega_1} \\ \mathrm{Holant}_{\Omega_2} \\ \vdots \\ \mathrm{Holant}_{\Omega_{(n+1)^2}} \end{bmatrix}. \tag{5.5}$$

We denote A the matrix on the left-hand side. If A is non-degenerate, then we can reconstruct β for any given instance. This would enable us to interpolate Holant_Ω for all instances Ω obtained by replacing all appearances of function B with function $Q^{-1} diag(1, a, b, 0) Q$ for any a and b.

To study the condition under which A is non-degenerate, we calculate the determinant of A.

$$\det(A) = \det\left(\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{(n+1)^2} \end{bmatrix}\right) = (xy)^{\frac{n(n+1)^2}{2}} \det\left(\begin{bmatrix} 1 \\ \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_{(n+1)^2-1} \end{bmatrix}\right)$$

$$= (xy)^{\frac{n(n+1)^2}{2}} \prod_{\substack{i_1 > i_2 \text{ or} \\ i_1 = i_2, j_1 > j_2}} \left(x^{i_1} y^{j_1} - x^{i_2} y^{j_2}\right).$$

The final equality is due to the fact that the resulting matrix is a Vandermonde matrix. Therefore, $\det(A) = 0$ only if $xy = 0$ or $x^{i_1} y^{j_1} = x^{i_2} y^{j_2}$ for some nonnegative integers i_1, i_2, j_1, j_2 such that either $i_1 > i_2$, or $i_1 = i_2$ and $j_1 > j_2$.

We now try to construct $[1, 0, 0, 0, 1]$ using polynomial interpolation discussed above. Consider the signature $X = [1, 0, b, 0, 1]$. If we write X as a 4×4 matrix, then it is easy to verify that $X = Q^{-1} diag(1+b, 1-b, 2b, 0)Q$, since b is transcendental; we further have $(1+b)(1-b)(2b) \neq 0$, and $X = (1+b)Q^{-1} diag(1, \frac{1-b}{1+b}, \frac{2b}{1+b}, 0)Q$. Since $(1+b)$ is only a multiplicative constant, we can view X as $X = Q^{-1} diag(1, \frac{1-b}{1+b}, \frac{2b}{1+b}, 0)Q$ without loss of generality. We can use X to construct gadgets as in Figure 5.1, and since $1 - b \neq 0$, $b \neq 0$, this interpolation fails only when

$$(1-b)^{i_1}(2b)^{j_1}(1+b)^{i_2+j_2} = (1-b)^{i_2}(2b)^{j_2}(1+b)^{i_1+j_1}. \tag{5.6}$$

We first argue that the above equation is non-trivial. The coefficient of the leading term of the left-hand side is $(-1)^{i_1}2^{j_1}$ and that of the right-hand side is $(-1)^{i_2}2^{j_2}$. Hence if $i_1 = i_2$, then $j_1 > j_2$ and those terms would be different. The same argument holds if $j_1 \neq j_2$, or $j_1 = j_2$, but i_1 and i_2 have different parity. Now we consider the case when $j_1 = j_2$, $i_1 > i_2$, and that i_1 and i_2 have the same parity. Then Equation (5.6) becomes trivial implies that the following equation is also trivial

$$(1-b)^{i_1-i_2} = (1+b)^{i_1-i_2}. \tag{5.7}$$

Since the above equation is clearly non-trivial, neither is Equation (5.6).

This is an integer coefficient equation on variable b, and therefore all roots must be algebraic numbers. Since we are considering transcendental number here, we know that Equation (5.6) would never hold, and thus we can always use polynomial interpolation to realize $Q^{-1} diag(1, a, b, 0)Q$ for any a, b. Specifically, by setting $a = 1$ and $b = 0$, we can realize the 4-ary equality signature. This completes the proof. □

For the cases when b is an algebraic real number, we use our second new technique—realizing a signature by approximating it. Here is the formal statement.

THEOREM 5.8. *Let $f = [x_0, \ldots, x_k]$ and $v = [v_0, \ldots, v_l]$ be symmetric Boolean signatures of arity k and l, and $\{g_m\}$ be a sequence of signatures of arity k. We assume that all the signature values are real algebraic numbers of degree at most d, and there exists a* *constant $C > 1$ such that for all m, we have $|f - g_m|_\infty < C^{-m}$. If we can compute Holant(g_m, v) in time poly(n, m), where n is the number of the vertices, then we can compute Holant(f, v) in polynomial time.*

To complete the proof of Theorem 5.8, we need the following technical lemma about real algebraic numbers.

LEMMA 5.9 (Corollary 3.12 of Pollard & Diamond 1998). *Let a and b be algebraic integers, and $\{\alpha_i\}_{i\in\{1\ldots n\}}$, $\{\beta_i\}_{i\in\{1\ldots m\}}$ be conjugates of a and b (including a and b) of the corresponding minimal integer polynomials that define a and b, respectively. Then*

$$\prod_{i=1}^{n}\prod_{j=1}^{m}(x - \alpha_i - \beta_j)$$

is an integer polynomial in x with $(a + b)$ as one of its roots.

The following lemma says that linear combinations of algebraic numbers cannot be too close to each other. This is an important property we use to recover the true values from approximate values.

LEMMA 5.10. *Let $x_i \in \{-D^t, \ldots, D^t\}$ for $i = 1, \ldots, k$ be a set of integer variables and $\{v_i\}_{1\le i\le k}$'s be a set of algebraic numbers. Let*

$$S = \left\{ \sum_{i=1}^{k} x_i v_i | x_i \in \{-D^t, \ldots, D^t\} \right\}.$$

Then there exists a constant $C > 1$ depending only on D and $\{v_i\}$, such that for all t and all distinct $r_1, r_2 \in S$, $|r_1 - r_2| > C^{-t}$.

PROOF. We relax the range of $x_i's$ to $\{-2D^t, \ldots, 2D^t\}$, so we have a new set S' which contains all $|r_1 - r_2|$'s from the original S. Then we only need to show a lower bound for norms of nonzero elements in S'.

Let d be the maximum degree of v_i's. For any set of x_i's, $\sum_{i=1}^{k} x_i v_i$ is also an algebraic number, and the norm of the product of all its conjugates is a positive rational, which is the constant term in the minimal polynomial. This value is lower-bounded by L^{-t} for some constant L depending on D and v_i's. On the other hand, the complex norm of these conjugates are upper-bounded by L'^t, where L' is also a constant depending on D and v_i's. The number of conjugates of $\sum_{i=1}^k x_i v_i$ is upper-bounded by d^k. Therefore, we have that the norm of $\sum_{i=1}^k x_i v_i$ is lower-bounded by $L^{-t}/(L'^t)^{d^k} = 1/(LL'^{d^k})^t =: C^{-t}$. Note that LL'^{d^k} only depends on D and v_i's. □

We also need an algorithm for integer programming when the number of variables is fixed. More specifically, consider the following problem:

(ILP$_\mathbb{R}$) Given a matrix $A \in \mathbb{R}^{m\times n}$ and vectors $\mathbf{b} \in \mathbb{R}^m$, $\mathbf{d} \in \mathbb{R}^n$, decide whether there is $\mathbf{x} \in \mathbb{Z}^n$ such that $A\mathbf{x} \le b$, where $\mathbf{0} \le \mathbf{x} \le \mathbf{d}$.

Theorem 5.11. (Lenstra (1983), Brimkov & Dantchev (2000)) *There is an $O(m \log \|d\|)$ algorithm for ILP$_\mathbb{R}$ of fixed dimension n.*

Proof (Theorem 5.8). Given a Holant instance with n vertices labeled either f or v, we consider the error introduced by replacing f with g_m. We first consider the error of Holant value for a fixed edge assignment. Suppose the vertices take values f_1, f_2, $\dots$, f_n. Then the value of the signature of F with g_m replacing f is within $[\min\prod_{t=1}^n (f_t \pm C^{-m}), \max\prod_{t=1}^n (f_t \pm C^{-m})]$, where minimum and maximum is taken over different choices of plus and minus signs. Let $M = \max\{1, |x_0|, \dots, |x_k|, |v_0|, \dots, |v_l|\}$ be the maximum absolute value appearing in function f and v. Then the maximum error in the value of a single assignment is

$$\left(M + C^{-m}\right)^n - M^n = \sum_{s=1}^{n} \binom{n}{s} C^{-ms} M^{n-s} \le nC^{-m}(Mn)^n.$$

To get the last inequality, note that for each of the n terms in the summation, $C^{-ms} \le C^{-m}$, $M^{n-s} \le M^n$, $\binom{n}{s} \le n^n$. There are at most $2^{n\max\{k,l\}}$ possible edge assignments, and summing over all of them gives us a corresponding multiplicative factor as an upper bound. Therefore, the total error is upper-bounded by

$$nC^{-m}(Mn)^n 2^{n\max\{k,l\}}, \tag{5.12}$$

where M and C are some constants greater than 1 and m is a parameter we could choose. Now we need to choose a proper m such that given the hypothesis of the theorem, we can compute $\mathrm{Holant}(f, v)$ in polynomial time.

Let $S = \{0, 1, 2, \ldots, k\}$, $T = \{0, 1, \ldots, l\}$. Given an edge assignment of a Holant instance, the corresponding product for this assignment is a product of powers of these x_i and v_i. Hence the Holant of a given instance can be written as

$$\sum_{\substack{\mathbf{y}\in\{0,\ldots,n\}^S\\ \mathbf{z}\in\{0,\ldots,n\}^T}} c_{\mathbf{y},\mathbf{z}} \prod_{i=0}^{k} x_i^{\mathbf{y}_i} \prod_{i=0}^{l} v_i^{\mathbf{z}_i},$$

where $c_{\mathbf{y},\mathbf{z}} \in \{0, \ldots, 2^{n \max\{k,l\}}\}$ is some integer indexed by $\mathbf{y}$ and $\mathbf{z}$, corresponding to the number of edge assignments that evaluates to $\prod_{i=0}^{k} x_i^{\mathbf{y}_i} \prod_{i=0}^{l} v_i^{\mathbf{z}_i}$. Since x_i and v_i are algebraic numbers of degree at most d, we can replace x_i^r $(r \geq d)$ with $\sum_{j=0}^{d-1} c'_j x_i^j$, and similarly for v_i^r $(r \geq d)$. It is easy to see that the new coefficients are rational numbers, and if we denote c'_i as p_i/q_i, where $(p_i, q_i) = 1$, then $|p_i|, |q_i| \in \{0, \ldots, C^n\}$, where C is a constant depending only on x_i and v_i. We do this for all the x_i and v_i, and we have

$$\begin{aligned}\sum_{\substack{\mathbf{y}\in\{0,\ldots,n\}^S\\ \mathbf{z}\in\{0,\ldots,n\}^T}} c_{\mathbf{y},\mathbf{z}} \prod_{i=0}^{k} x_i^{\mathbf{y}_i} \prod_{i=0}^{l} v_i^{\mathbf{z}_i} &= \sum_{\substack{\mathbf{y}'\in\{0,\ldots,d-1\}^S\\ \mathbf{z}'\in\{0,\ldots,d-1\}^T}} c''_{\mathbf{y}',\mathbf{z}'} \prod_{i=0}^{k} x_i^{\mathbf{y}'_i} \prod_{i=0}^{l} v_i^{\mathbf{z}'_i}\\ &\triangleq \frac{1}{C_H} \sum_{i=1}^{d^{k+l}} w_i u_i. \end{aligned} \tag{5.13}$$

Here $c''_{\mathbf{y}',\mathbf{z}'}$ are rational coefficients depending on x_i, v_i and $c_{\mathbf{y},\mathbf{z}}$, and if we denote it as $p_{\mathbf{y}',\mathbf{z}'}/q_{\mathbf{y}',\mathbf{z}'}$, and $(p_{\mathbf{y}',\mathbf{z}'}, q_{\mathbf{y}',\mathbf{z}'}) = 1$, then $p_{\mathbf{y}',\mathbf{z}'}, q_{\mathbf{y}',\mathbf{z}'} \in \{-C'^{n^2}, \ldots, C'^{n^2}\}$ where C' is a constant depending on S, T, d, x_i, v_i. In the last step in Equation (5.13), we take C_H as the least common multiple of the $c''_{\mathbf{y}',\mathbf{z}'}$; thus, w_i's are integers, and C_H and w_i are in $\{-D^{n^2}, \ldots, D^{n^2}\}$ where D is a constant depending on S, T, d, x_i and v_i, while u_i's are constants that correspond to $\prod_{i=1}^{k} x_i^{\mathbf{y}'_i} \prod_{i=1}^{l} v_i^{\mathbf{z}'_i}$. If we can find a group of integer coefficients w_i

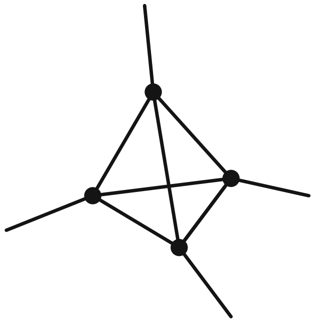

Figure 5.2: The tetrahedron gadget.

such that the last summation equals the original Holant instance, then we are done.

Let C_0 be the constant guaranteed by Lemma 5.10. The idea is to choose m in Equation (5.12), such that the error (5.12) is less than $\frac{1}{3}C_0^{-n^2}$. Such an m is still a polynomial of n, and therefore we can approximate the Holant value in polynomial time. Also, different values of the form in Equation (5.13) are at least $C_0^{-n^2}$ away, so the $(-\frac{1}{3}C_0^{-n^2}, \frac{1}{3}C_0^{-n^2})$ interval around these values are disjoint, and therefore the true Holant value is in exactly one of them. We can now form an $\mathrm{ILP}_{\mathbb{R}}$, which has the coefficients w_i's in Equation (5.13) as integer variables and states that the $(-\frac{1}{3}C_0^{-n^2}, \frac{1}{3}C_0^{-n^2})$ neighbor of the approximated value contains the RHS of (5.13). By solving this $\mathrm{ILP}_{\mathbb{R}}$, we can find out a set of coefficients w_i's which gives the true Holant value as the RHS of (5.13).

Remark 5.14. *The set of solutions may not be unique, but the above argument guarantees that the resulting sum is unique.*

□

In the following, we use the above reduction to study the complexity of Holant$([1, 0, b, 0, 1])$. For $[1, 0, b, 0, 1]$, using the tetrahedron gadget in Figure 5.2, we realize a new symmetric signature: $[(b+1)^2(3b^2-2b+1), 0, 2b^2(b+1)^2, 0, (b+1)^2(3b^2-2b+1)]$. The signature is not all-zero due to the assumption that $b \neq -1$. On keep doing this recursive construction, we can realize a signature $[1, 0, b_r, 0, 1]$ with $b_r = \frac{2b_{r-1}^2}{3b_{r-1}^2-2b_{r-1}+1}$. The following lemma shows that this recursive construction converges to a fixed point very fast.

LEMMA 5.15. *Let b be a real algebraic number, $b \neq 0, b \neq \pm 1, b \neq \frac{1}{3}$. Let $[1, 0, b_r, 0, 1]$ be the signature realized by the rth recursive tetrahedron gadget starting from $[1, 0, b, 0, 1]$. Let $\beta = 0$ if $b < \frac{1}{3}$, and $\beta = 1$ otherwise. Then $|b_r - \beta| < C^{-2^r}$, where $C > 1$ is some constant. In other words, the recursive construction converges to either $[1, 0, 0, 0, 1]$ or $[1, 0, 1, 0, 1]$, depending on whether b is smaller than $\frac{1}{3}$ or not.*

PROOF. We can rewrite the recursion

$$b_r = \frac{2b_{r-1}^2}{3b_{r-1}^2 - 2b_{r-1} + 1}$$

as

$$b_r = \frac{2}{2 + \left(\frac{1}{b_{r-1}} - 1\right)^2}.$$

And we further have

$$\frac{1}{2}\left(\frac{1}{b_r} - 1\right) = \left(\frac{1}{2}\left(\frac{1}{b_{r-1}} - 1\right)\right)^2,$$

and finally

$$\frac{1}{2}\left(\frac{1}{b_r} - 1\right) = \left(\frac{1}{2}\left(\frac{1}{b} - 1\right)\right)^{2^r}.$$

Depending on whether $\frac{1}{2}(\frac{1}{b} - 1) > 1$ or $\frac{1}{2}(\frac{1}{b} - 1) < 1$, we have that the recursion b_r converges to 0 or 1 exponentially fast, which is exactly what we need. The only exceptional case $\frac{1}{2}(\frac{1}{b} - 1) = 1$ corresponds to $b = \frac{1}{3}$, which was excluded in the statement of the lemma. □

The rth gadget contains 4^r nodes. We do the recursive gadget $O(\log n)$ levels so it is still of polynomial size. We note that this is the reason why we cannot use the tetrahedron gadget to interpolate since we would need polynomial many levels. The speed of convergence in Lemma 5.15 is so fast that we can approximate the target gadget to within $C^{-poly(n)}$ by a gadget with $O(\log n)$ levels of recursive construction. Then by Theorem 5.8 and the above analysis, we get the hardness for Holant$([1, 0, b, 0, 1])$ when $b \notin \{0, 1, -1, \frac{1}{3}\}$. This also gives us hardness of Holant$([0, 0, 1, 0, 0])$.

Claim 5.16. *The problem* Holant$([0,0,1,0,0])$ *is #P-hard.*

Proof. We first apply an orthogonal holographic transformation using the following matrix

$$Q = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix},$$

and we have

$$\text{Holant}([0,0,1,0,0]) \equiv_T \text{Holant}([1,0,-\frac{1}{3},0,1]).$$

It follows from the argument above that the problem is #P-hard. □

To complete the proof for Lemma 5.3, the only remaining case is $[1,0,\frac{1}{3},0,1]$. This is done in Lemma 6.9 by a reduction from counting number of Eulerian orientations on 4-regular graphs.

6. Hardness of counting Eulerian orientations in regular undirected graphs

In this section, we prove a hardness result for a rather independent problem: counting Eulerian orientations in regular undirected graphs. We show hardness for this problem here in terms of the Holant framework.

First we define Eulerian orientations.

Definition 6.1. *Given a graph $G = (V,E)$ of which all vertices have even degree. Let σ be an orientation of its edges E. Then σ is an Eulerian orientation iff for each vertex $v \in V$, the number of incoming edges and outgoing edges of v are the same.*

To prove the hardness of counting Eulerian orientations, we show how to use it to calculate a certain hard-to-compute weighted sum of orientations on the medial graph of planar graphs. We recall the definition of medial graphs.

Definition 6.2 (Las Vergnas 1988). *Let G be a plane graph. For simplicity, we assume that G is connected and that every edge of G is contained in exactly two different faces. Define its medial graph $H = (V_H, E_H)$, where V_H consists of the middle points of edges in G, and for each face in G, connect the middle points on the border of a face of G to get a cycle, and E_H consists of all edges on all cycles.*

Note that medial graphs are 4-regular graphs.

The following theorem shows the relation between Eulerian orientations, medial graphs, and Tutte polynomials. For the definition of Tutte polynomial, we refer to Bollobás (1998).

Theorem 6.3 (Las Vergnas 1988). *Let G be a connected planar graph, and let $\mathscr{O}(H)$ be the set of all Eulerian orientations of the medial graph $H = H(G)$. Then*

$$\sum_{O \in \mathscr{O}(H)} 2^{\beta(O)} = 2 \cdot T(G; 3, 3), \tag{6.4}$$

where $\beta(O)$ is the number of saddle vertices in orientation O, i.e., vertices in which the edges are oriented "in, out, in, out" in cyclic order.

It is known that calculating the right-hand side of the above is #P-hard.

Theorem 6.5 (Jaeger *et al.* 1990, Vertigan 2005). *For $x, y \in \mathbb{R}$, if $(x, y) \in \{(1, 1), (-1, -1), (0, -1), (-1, 0)\}$ or satisfies $(x - 1)(y - 1) = 1$, the Tutte polynomial is computable in polynomial time. Otherwise, it is #P-hard. If the problem is restricted to the class of planar graphs, the points on the hyperbola defined by $(x-1)(y-1) = 2$ become polynomial-time computable, but all other points remain #P-hard.*

Before we prove the main theorem of this section, first observe that $\text{Holant}([0, 0, 1, 0, 0] | [0, 1, 0])$ is exactly the number of Eulerian orientations in a 4-regular graph.

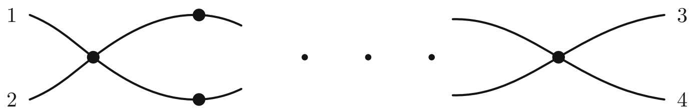

Figure 6.1: Recursive gadget. 4-ary signatures are $[0, 0, 1, 0, 0]$, and binary ones are $[0, 1, 0]$.

CLAIM 6.6. *Let $G = (V, E)$ be a 4-regular graph. Define bipartite graph $G' = (X, Y, E')$ as the following:*

$X = \{v_x | x \in V\}$, $Y = \{v_e | e \in E\}$, $E' = \{(v_x, v_e) | x \in e\}$. *Then the number of Eulerian orientations of G is equal to $\text{Holant}_{G'}([0, 0, 1, 0, 0] | [0, 1, 0])$.*

Now we prove the main theorem of this section. We show how to calculate the LHS in Theorem 6.3 given an oracle of counting Eulerian orientations.

THEOREM 6.7. *Counting Eulerian orientations is #P-hard for 4-regular graphs.*

PROOF. We reduce calculating the LHS of Equation (6.4) to $\text{Holant}([0, 0, 1, 0, 0] | [0, 1, 0])$. Then since it is known that calculating the Tutte polynomial on graphs at $(3, 3)$ is #P-hard, we conclude that $\text{Holant}([0, 0, 1, 0, 0] | [0, 1, 0])$ is #P-hard.

Suppose we have $[0, 0, 1, 0, 0]$ on the left and $[0, 1, 0]$ on the right. Consider the recursive gadget in Figure 6.1. Let

$$P = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & -1 & 0 \end{bmatrix},$$

$$G_0 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

For a 4-ary gadget, we can represent it as a 4×4 matrix, where the rows indicate the two inputs on the left side, and the columns indicate the two inputs on the right side, and the inputs are ordered in lexicographical order. Then the signature of the gadget in Figure 6.1 is actually

$$G_0 G_1^k = G_0 P^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 2^k & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} P.$$

We realize the following signature via interpolation

$$G_0 P^{-1} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} P = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}.$$

We call this signature G_x. Now we show that

$$\text{Holant}_{G_H}(G_x | [0, 1, 0]) = \sum_{O \in \mathscr{O}(H)} 2^{\beta(O)},$$

for a suitably constructed bipartite graph G_H. The vertices on the left side of G_H correspond to vertices in H, and the vertices on the right correspond to edges of H, and they are connected in the natural way. Note that the resulting graph is not planar any more, since we need to change the layout of the constructed gadgets to match the weights. More specifically, according to the current layout, the signature evaluates to 1 when both input bits 1 and 2 are 1, or both input bits 3 and 4 are 1, and all other inputs of weight 2 evaluate to 1/2. Thus, when replacing it in the medial graph, the order of edges should be 1-3-2-4, so that assignments with two 1's that give 1 to non-neighboring inputs evaluate to 1, and assignments with two 1's that give 1 to neighboring inputs evaluate to 1/2.

Clearly, there is a 1-1 correspondence between edge assignments with nonzero values and Eulerian orientations of H: For each edge, its orientation in H corresponds to an assignment of 0 and 1 on RHS vertices of G_H corresponding to that edge. Also, in H, saddle vertices contribute a factor of 2 to the value of the assignment, while other vertices contribute 1, and correspondingly in

G_H, assignments that produce saddle vertices cause the function at that vertex to evaluate to 1, while the others evaluate to $\frac{1}{2}$, differing by a factor of 2. Therefore, the weight of this assignment is exactly $2^{\beta(O)-n}$, where n is the number of nodes, and we are done with the reduction. □

Next we show that counting Eulerian orientations in all $2k$-regular graphs are also hard.

COROLLARY 6.8. *For all $2k$-regular graphs, $k \geq 2$, counting Eulerian orientations is #P-hard.*

PROOF. Let X be a $2k$-ary signature which evaluates to 0 unless the weight of the input is k, when it becomes 1. Similar to Claim 6.6, observe that $\text{Holant}(X|[0,1,0])$ characterizes exactly the problem of counting Eulerian orientations in a $2k$-regular graph. Connecting $\frac{2k-4}{2}$ $[0,1,0]$'s to a single X gives a $[0,0,1,0,0]$ on the LHS; thus, $\text{Holant}(X|[0,1,0]) \geq_T \text{Holant}([0,0,1,0,0]|[0,1,0])$ and is thus #P-hard. □

LEMMA 6.9. $\text{Holant}([1,0,\frac{1}{3},0,1])$ *is #P-hard.*

PROOF. By applying holographic transformation $\begin{bmatrix} 1 & 1 \\ i & -i \end{bmatrix}$, we have that

$$\text{Holant}\left(\left[1,0,\frac{1}{3},0,1\right]\right) \equiv_T \text{Holant}([0,0,1,0,0]|[0,1,0]).$$

This is exactly the counting Eulerian orientations problem on 4-regular graph. By Theorem 6.7, it is #P-hard. □

7. Dichotomy for Real Holant

In this section, we prove our main result. The idea of the proof is to use induction on the arity of the signatures. We apply dichotomy theorems for signatures with smaller arities for the induction step. The base step would be dichotomy theorems for signatures of arity three and four. The ternary case is proved in Theorem 4.9 in Section 4. In this section, we go on to analyze complexity of signatures of arity four. We start with the following lemma in which we have an additional unary signature.

LEMMA 7.1. *Let X be a non-degenerate real 4-ary signature and $a, b \in \mathbb{R}$ such that they are not both zero. Then Holant$(\mathcal{F}, X, [a, b])$ is #P-hard unless $\mathcal{F} \cup \{X, [a, b]\}$ is $\mathscr{A}\&\mathscr{P}$-compatible.*

PROOF. Since a, b are not both zero, we can apply a real orthogonal transformation Q, so that $Q[a, b] = [1, 0]$. Let $Y = Q^{\otimes 4} X = [y_0, y_1, y_2, y_3, y_4]$. Note that Y is still a real signature. Since Holant$(Q\mathcal{F}, Y, [1, 0])$ has the same value as the original problem, it is equivalent to just considering the problem after transformation. Using $[1, 0]$ in this transformed instance, we realize $Y' = [y_0, y_1, y_2, y_3]$. If Y' is non-degenerate, we apply Theorem 4.9. Now consider the case that Y' is degenerate.

If Y' is an all-zero signature, then Y is degenerate, which means that X is degenerate, contradicting our hypothesis. If $Y' = [1, 0]^{\otimes 3}$, then $Y = [1, 0, 0, 0, *]$ is a non-degenerate generalized Fibonacci signature and we apply Corollary 4.10. If $Y' = [0, 1]^{\otimes 3}$, by adding a self-loop, we can realize $[0, 1]$. Since we have both $[1, 0]$ and $[0, 1]$, we apply Theorem 2.4, the dichotomy theorem for Holant^c. Otherwise, by a scaling, we can assume that $Y' = [1, t]^{\otimes 3}$, where $t \in \mathbb{R}\backslash\{0\}$, and $Y = [1, t, t^2, t^3, y]$, where $y \neq t^4$. Connecting three copies of $[1, 0]$ to Y, we can realize $[1, t]$. Connecting one copy of $[1, t]$ to Y, we have $Y'' = [1 + t^2, t + t^3, t^2 + t^4, t^3 + yt]$. This is a non-degenerate ternary function for any real $t \neq 0$ and $y \neq t^4$. We now apply Theorem 4.9 to finish the proof. □

By a similar argument as in Lemma 4.7, we can replace $[a, b]$ with $[a, b]^{\otimes 2}$.

LEMMA 7.2. *Let X be a non-degenerate real 4-ary signature, $a, b \in \mathbb{R}$ such that they are not both zero. Then Holant$(\mathcal{F}, X, [a, b]^{\otimes 2})$ is #P-hard unless $\mathcal{F} \cup \{X, [a, b]^{\otimes 2}\}$ is $\mathscr{A}\&\mathscr{P}$-compatible.*

We now prove a theorem for Holant problems when we have a non-degenerate 4-ary function.

THEOREM 7.3. *Let $X = [x_0, x_1, x_2, x_3, x_4]$ be a non-degenerate real-valued 4-ary signature and $\mathcal{F}$ be a set of symmetric real-valued* *signatures. Then Holant$(X, \mathcal{F})$ is #P-hard unless $\mathcal{F} \cup \{X\}$ is $\mathscr{A}\&\mathscr{P}$-compatible, for which there is a polynomial-time algorithm.*

PROOF. As usual, the tractability part follows from algorithms for #CSP. We prove the hardness part. The main idea is to realize a degenerate binary function and make use of Lemma 7.2.

By adding a self-loop to X, we have $X' = [x_0 + x_2, x_1 + x_3, x_2 + x_4]$. If X' is all-zero, then we have $X = [x_0, x_1, -x_0, -x_1, x_0]$ and we apply Corollary 4.11. If $X' = [x_0 + x_2, x_1 + x_3, x_2 + x_4]$ is degenerate and not all-zero, then we apply Lemma 7.2 directly.

Now we assume that X' is non-degenerate. We make a polynomial interpolation by a chain of k copies of signature X'. The eigenvalues of $X' = \begin{bmatrix} x_0 + x_2 & x_1 + x_3 \\ x_1 + x_3 & x_2 + x_4 \end{bmatrix}$ are $\lambda_{1,2} = \frac{(x_0+2x_2+x_4)\pm\sqrt{\Delta}}{2}$ where $\Delta = (x_4 - x_0)^2 + 4(x_1 + x_3)^2$. We can realize X'_k by a chain of k signature X'. Since X' is real and symmetric, we have $X'_k = P \begin{bmatrix} \lambda_1^k & 0 \\ 0 & \lambda_2^k \end{bmatrix} P^{-1}$, where P is the orthonormal basis formed by its eigenvectors. We already know that $\lambda_1\lambda_2 \neq 0$ since X' is non-degenerate. If we further have that the ratio $\frac{\lambda_1}{\lambda_2}$ is not a root of unity, we can interpolate all the binary signatures expressible as $P \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix} P^{-1}$. In particular, we can interpolate $P \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} P^{-1}$, which is a degenerate nonzero binary signature. We are done by Lemma 7.2.

The exceptional case is that the ratio $\frac{\lambda_1}{\lambda_2}$ is a root of unity. Since X' is a real symmetric function, both λ_1 and λ_2 are real. So the only possible roots of unity are ± 1. We have that $\lambda_1 = \lambda_2$ iff $\Delta = 0$ iff $x_4 = x_0$ and $x_1 = -x_3$. Also, $\lambda_1 = -\lambda_2$ iff $(x_0 + x_2) = -(x_2 + x_4)$. We deal with these exceptional cases separately as follows.

Case 1: $x_4 = x_0 \neq 0$ and $x_1 = -x_3$. This is of form $[1, a, b, -a, 1]$, and we apply Lemma 5.2.

Case 2: $x_4 = x_0 = 0$ and $x_1 = -x_3$. If we further have $x_2 = 0$, then this is a signature of form $[x, y, -x, -y, x]$ and we apply Corollary 4.11. Otherwise, it is of form $[0, a, 1, -a, 0]$. By the tetrahedron gadget, we can realize a signature of $[6a^2 + 3, a, 2a^2 + 2, -a, 6a^2 + 3]$. Since $6a^2 + 3 \neq 0$, this case is proved in case 1.

Case 3: $(x_0 + x_2) = -(x_2 + x_4)$, and it does not belong to either case 1 or case 2. If $x_2 \neq 0$, then after scaling we can assume that X has form $[a, b, 1, c, -2-a]$ for some $a, b, c \in \mathbb{R}$. If $x_2 = 0$, then we can assume that X has form $[a, b, 0, c, -a]$ for some $a, b, c \in \mathbb{R}$. We apply the tetrahedron gadget with X, and let the resulting signature be $Y = [y_0, y_1, y_2, y_3, y_4]$. We study the condition under which $(y_0 + y_2) = -(y_2 + y_4)$, or $y_0 + 2y_2 + y_4 = 0$.

Using the tetrahedron gadget with $[a, b, 1, c, -2-a]$, we realize a signature of $[y_0, y_1, y_2, y_3, y_4]$ where

$$\begin{aligned}
y_0 &= c^4 + 6c^2 + 4b^3c + 12bc + 3b^4 + 6a^2b^2 + 12ab^2 + 12b^2 \\
&\quad + a^4 + 4a + 3, \\
y_1 &= -ac^3 + c^3 + 6bc^2 + 3ab^2c + 9b^2c + 2ab^3 + 4b^3 + a^3b \\
&\quad + 3a^2b + 3ab + b, \\
y_2 &= c^4 + 2bc^3 + 2b^2c^2 + a^2c^2 + 2ac^2 + 3c^2 + 2b^3c + 4bc \\
&\quad + b^4 + a^2b^2 + 2ab^2 + 3b^2 + 2a^2 + 4a + 2, \\
y_3 &= -2ac^3 - 3abc^2 + 3bc^2 + 6b^2c - a^3c - 3a^2c - 3ac - c \\
&\quad + ab^3 + 3b^3, \\
y_4 &= 3c^4 + 4bc^3 + 6a^2c^2 + 12ac^2 + 12c^2 + 12bc + b^4 + 6b^2 \\
&\quad + a^4 + 8a^3 + 24a^2 + 28a + 11.
\end{aligned}$$

In this case, $y_0 + 2y_2 + y_4 = 2((a+1)^2 + (b+c)^2)((a+1)^2 + (b-c)^2 + 2b^2 + 2c^2 + 8)$; thus, the only real solution to $y_0 + 2y_2 + y_4 = 0$ is $a = -1$, $b = -c$. In this case, the signature X can be written as $X = [-1, b, 1, -b, -1]$ and this belongs to case 1.

Using the tetrahedron gadget with $[a, b, 0, c, -a]$ gives a signature of $[y_0, y_1, y_2, y_3, y_4]$ where

$$\begin{aligned}
y_0 &= c^4 + 4b^3c + 3b^4 + 6a^2b^2 + a^4, \\
y_1 &= -ac^3 + 3ab^2c + 2ab^3 + a^3b, \\
y_2 &= c^4 + 2bc^3 + 2b^2c^2 + a^2c^2 + 2b^3c + b^4 + a^2b^2, \\
y_3 &= -2ac^3 - 3abc^2 - a^3c + ab^3, \\
y_4 &= 3c^4 + 4bc^3 + 6a^2c^2 + b^4 + a^4.
\end{aligned}$$

In this case, $y_0 + 2y_2 + y_4 = 2(a^2 + (b+c)^2)(a^2 + 2b^2 + 2c^2 + (b-c)^2)$, and the only real solution to $y_0 + 2y_2 + y_4 = 0$ is $a = 0, b = -c$. In this case, $X = [0, b, 0, -b, 0]$ and belongs to case 2.

To summarize, if X is of case 3 but not case 1 or case 2, then Y is not of case 3. In particular, Y cannot be all-zero.

Now we consider whether Y can be degenerate. Assume that $y_0 \neq 0$, and by scaling assume that $y_0 = 1$. Signature Y being degenerate means that $y_2^2 = y_4$. Combining this with $1+2y_2+y_4 = 0$, we have that $y_2 = -1$, $y_4 = 1$. This means that $y_1, y_3 \in \{-i, i\}$. This is not possible because X is a real-valued signature and so must Y.

Therefore, whenever X is of case 3 but not of case 1 or case 2, we can construct Y that is non-degenerate and not of case 3. Therefore Y can be handled in a setting that has already been proved. This completes the proof. □

Now we are ready to prove our main result.

Proof (Theorem 3.2). As stated in the outline in Section 3, we prove this theorem by showing that for any non-degenerate signature X with arity at least three, $\text{Holant}(X, \mathcal{F})$ is tractable iff there exists a 2×2 matrix satisfying the conditions. We proceed by induction on the arity k of X.

The cases of $k = 3$ and $k = 4$ are proved in Theorem 4.9 and Theorem 7.3.

Suppose for arity $k < n$, we have proved our claim. Now we have a non-degenerate signature X of arity $n \geq 5$. We obtain an $(n-2)$-ary signature X' by adding a self-loop to X. If X' is non-degenerate, then we are done by induction hypothesis. If X' is all-zero, then X is of form $[x, y, -x, -y, x, y, -x, -y \ldots]$ and we apply Corollary 4.11. The only remaining case is that X' is degenerate but not all-zero, and we assume that $X' = [a, b]^{\otimes(n-2)}$ for some real a and b that are not both zero. By applying an appropriate *real* orthogonal transformation, we transform X' into $[1, 0]^{\otimes(n-2)}$, and X into $Y = XQ^{\otimes n} \triangleq [y_0, y_1, \ldots, y_n]$. By Theorem 2.3, we may just assume that we actually have Y in the place of X. The fact that X' is transformed into $Y' = [1, 0]^{\otimes(n-2)}$ implies that $Y = [y_0, y_1, y_2, -y_1, -y_2, \ldots]$. After adding enough self-loops to Y' we can get either $[1, 0]$ or $[1, 0, 0]$ depending on the parity of n. Then connecting some copies of $[1, 0]$ or $[1, 0, 0]$ to Y, we can get either $Y'' = [y_0, y_1, y_2, -y_1]$ or $Y'' = [y_0, y_1, y_2, -y_1, -y_2]$. We argue that

Y'' is not degenerate. If $y_1 = 0$, then we must have that $y_2 \neq 0$ since otherwise Y is degenerate, contradicting the assumption that X (and hence Y) is non-degenerate. Otherwise, $y_1 \neq 0$. If Y'' is degenerate, then $y_0, y_1, y_2, -y_1, (-y_2)$ form a geometric sequence, and denote the ratio $r = y_2/y_1$. It must be that $\pm i$, and $y_0 = -y_2$. This implies that Y' is an all-zero signature, a contradiction. Now we know that such a Y'' is not degenerate, and we can complete the proof by Theorem 4.9 or Theorem 7.3. □

References

LENORE BLUM, FELIPE CUCKER, MICHAEL SHUB & STEVE SMALE (1998). *Complexity and real computation.* Springer-Verlag.

BÉLA BOLLOBÁS (1998). *Modern graph theory*, volume 184 of *Graduate Texts in Mathematics.* Springer-Verlag.

GRAHAM BRIGHTWELL & PETER WINKLER (2005). Counting Eulerian circuits is #P-Complete. In *Proceedings of the Seventh Workshop on Algorithm Engineering and Experiments and the Second Workshop on Analytic Algorithmics and Combinatorics,* Vancouver, BC, Canada, 259–262. Society for Industrial and Applied Mathematics.

VALENTIN E. BRIMKOV & STEFAN S. DANTCHEV (2000). On the complexity of integer programming in the Blum-Shub-Smale computational model. In *Proceedings of Theoretical Computer Science, Exploring New Frontiers of Theoretical Informatics, International Conference 2000,* Sendai, Japan, 286–300. Springer-Verlag.

ANDREI BULATOV, MARTIN DYER, LESLIE ANN GOLDBERG, MARKUS JALSENIUS & DAVID RICHERBY (2009). The complexity of weighted boolean #CSP with mixed signs. *Theoretical Computer Science* **410**, 3949–3961.

ANDREI A. BULATOV (2006). A dichotomy theorem for constraint satisfaction problems on a 3-element set. *Journal of the ACM* **53**(1), 66–120.

ANDREI A. BULATOV (2013). The complexity of the counting constraint satisfaction problem. *Journal of the ACM* **60**(5), 34.

Andrei A. Bulatov & Víctor Dalmau (2007). Towards a dichotomy theorem for the counting constraint satisfaction problem. *Information and Computation* **205**(5), 651–678.

Andrei A. Bulatov & Martin Grohe (2004). The complexity of partition functions. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming ICALP 2004,* Turku, Finland, 294–306. Springer-Verlag.

Andrei A. Bulatov & Martin Grohe (2005). The complexity of partition functions. *Theoretical Computer Science* **348**(2–3), 148–186.

Jin-Yi Cai, Xi Chen & Pinyan Lu (2011a). Non-negatively weighted #CSP: an effective complexity dichotomy. In *Proceedings of the 26th IEEE Conference on Computational Complexity,* San Jose, CA, USA, 45–54. IEEE Computer Society Press.

Jin-Yi Cai, Xi Chen & Pinyan Lu (2013a). Graph homomorphisms with complex values: a dichotomy theorem. *SIAM Journal on Computing* **42**(3), 924–1029.

Jin-Yi Cai, Heng Guo & Tyson Williams (2013b). A complete dichotomy rises From The capture Of vanishing signatures. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing,* Palo Alto, CA, USA, 635–644. ACM Press.

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

Jin-Yi Cai & Michael Kowalczyk (2010). A dichotomy for k-regular graphs with $\{0, 1\}$-vertex assignments And real edge functions. In *Proceedings of the 7th Annual Conference on Theory and Applications of Models of Computation,* Prague, Czech Republic, 328–339. Springer-Verlag.

Jin-Yi Cai, Pinyan Lu & Mingji Xia (2008). Holographic algorithms by Fibonacci gates And holographic reductions for hardness. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science,* Philadelphia, PA, 644–653. IEEE Computer Society Press.

Jin-Yi Cai, Pinyan Lu & Mingji Xia (2009). Holant problems and counting CSP. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing,* Bethesda, Maryland, USA, 715–724. ACM Press.

Jin-Yi Cai, Pinyan Lu & Mingji Xia (2010). Holographic algorithms with matchgates capture precisely tractable planar #CSP. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science,* Las Vegas, NV, 427–436. IEEE Computer Society Press.

Jin-Yi Cai, Pinyan Lu & Mingji Xia (2011b). A computational proof of complexity of some restricted counting problems. *Theoretical Computer Science* **412**(23), 2468–2485.

Jin-Yi Cai, Pinyan Lu & Mingji Xia (2013c). Holographic algorithms by Fibonacci gates. *Linear Algebra and its Applications* **438**(2), 690–707.

Nadia Creignou, Sanjeev Khanna & Madhu Sudan (2001). *Complexity classifications of boolean constraint satisfaction problems.* Society for Industrial and Applied Mathematics. ISBN 0-89871-479-6.

C. T. J. Dodson & T. Poston (1991). *Tensor geometry.* Springer-Verlag.

Martin E. Dyer, Leslie Ann Goldberg & Mark Jerrum (2009). The complexity of weighted boolean #CSP. *SIAM Journal on Computing* **38**(5), 1970–1986.

Martin E. Dyer, Leslie Ann Goldberg & Mike Paterson (2007). On counting homomorphisms to directed acyclic graphs. *Journal of the ACM* **54**(6).

Martin E. Dyer & Catherine S. Greenhill (2000). The complexity of counting graph homomorphisms (extended abstract). In *Proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms,* San Francisco, CA, USA, 246–255. Society for Industrial and Applied Mathematics.

Martin E. Dyer & Catherine S. Greenhill (2004). Corrigendum: The complexity of counting graph homomorphisms. *Random Structures and Algorithms* **25**(3), 346–352.

Martin E. Dyer & David Richerby (2010). On the complexity of #CSP. In *Proceedings of the 42nd Annual ACM Symposium on Theory of Computing,* Cambridge, MA, USA, 725–734. ACM Press.

Martin E. Dyer & David Richerby (2011). The #CSP dichotomy is decidable. In *Proceedings of the 28th International Symposium on Theoretical Aspects of Computer Science,* Dortmund, Germany, 261–272. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

John Faben (2008). The complexity of counting solutions to generalised satisfiability problems modulo k. *CoRR* **abs/0809.1836**.

Tomás Feder & Moshe Y. Vardi (1998). The computational structure of monotone monadic SNP and constraint satisfaction: a study through datalog and group theory. *SIAM Journal on Computing* **28**(1), 57–104.

Qi Ge & Daniel Stefankovic (2012). The complexity of counting Eulerian tours in 4-regular graphs. *Algorithmica* **63**(3), 588–601.

Leslie Ann Goldberg, Martin Grohe, Mark Jerrum & Marc Thurley (2010). A complexity dichotomy for partition functions with mixed signs. *SIAM Journal on Computing* **39**(7), 3336–3402.

Heng Guo, Sangxia Huang, Pinyan Lu & Mingji Xia (2011). The complexity of weighted boolean #CSP modulo k. In *Proceedings of the 28th International Symposium on Theoretical Aspects of Computer Science,* Dortmund, Germany, 249–260. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

Heng Guo, Pinyan Lu & Leslie G. Valiant (2013). The complexity of symmetric boolean parity Holant problems. *SIAM Journal on Computing* **42**(1), 324–356.

Pavol Hell & Jaroslav Nešetřil (1990). On the complexity of H-coloring. *Journal of Combinatorial Theory, Series B* **48**(1), 92–110.

Sangxia Huang & Pinyan Lu (2012). A dichotomy for real weighted Holant problems. In *Proceedings of the 27th IEEE Conference on Computational Complexity,* Porto, Portugal, 96–106. IEEE Computer Society Press.

E. Ising (1925). Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei* **31**(1), 253–258.

F. Jaeger, D. L. Vertigan & D. J. A. Welsh (1990). On the computational complexity of the Jones and Tutte polynomials. *Proceedings of the Cambridge Philosophical Society* **108**, 35–53.

Ker-I Ko (1991). *Complexity theory of real functions*. Birkhauser Boston Inc., Cambridge, MA, USA.

Michael Kowalczyk & Jin-Yi Cai (2010). Holant problems for regular graphs with complex edge functions. In *Proceedings of the 27th International Symposium on Theoretical Aspects of Computer Science,* Nancy, France, 525–536. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik.

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

Michel Las Vergnas (1988). On the evaluation at (3, 3) of the Tutte polynomial of a graph. *Journal of Combinatorial Theory, Series B* **45**(3), 367–372.

H. W. Jr. Lenstra (1983). Integer programming with a fixed number of variables. *Mathematics of Operations Research* **8**(4).

Milena Mihail & Peter Winkler (1996). On the number of Eulerian orientations of a graph. *Algorithmica* **16**(4/5), 402–414.

Christos H. Papadimitriou & Stathis Zachos (1982). Two remarks on the power of counting. In *Proceedings of the 6th GI-Conference on Theoretical Computer Science*, 269–276.

H. Pollard & H.G. Diamond (1998). *The theory of algebraic numbers*. Phoenix Edition Series. Dover Publications, Inc.

Leslie G. Valiant (1979). The complexity of computing the permanent. *Theoretical Computer Science* **8**, 189–201.

Leslie G. Valiant (2006). Accidental algorithms. In *FOCS*, 509–517.

Leslie G. Valiant (2008). Holographic algorithms. *SIAM Journal on Computing* **37**(5), 1565–1594.

Leslie G. Valiant (2010). Some observations on holographic algorithms. In *Proceedings of 9th Latin American Theoretical Informatics Symposium,* Oaxaca, Mexico, 577–590. Springer-Verlag.

Dirk Vertigan (2005). The computational complexity of Tutte invariants for planar graphs. *SIAM Journal on Computing* **35**(3), 690–712.

Manuscript received 20 September 2012

Sangxia Huang
École polytechnique fédérale de Lausanne
Lausanne, Switzerland
huang.sangxia@gmail.com
people.epfl.ch/sangxia.huang

Pinyan Lu
Microsoft Research Asia
Beijing, China
pinyanl@microsoft.com
http://research.microsoft.com/en-us/people/pinyanl/