3

Boolean #CSP

In this chapter we address the following type of counting problems, called Boolean #CSP. Let $\mathcal{F}$ be a set of functions, where each $f \in \mathcal{F}$ is a function $f : \{0, 1\}^k \to \mathbb{C}$, for some k depending on f, mapping Boolean variables to the complex numbers. In this chapter, the functions in $\mathcal{F}$ are not assumed to be symmetric in general. The Boolean counting constraint satisfaction problem #CSP($\mathcal{F}$) is defined as follows. The input I is a finite sequence of constraints on Boolean variables $x_1, x_2, \ldots, x_n$ of the form $f(x_{i_1}, x_{i_2}, \ldots, x_{i_k})$, where $f \in \mathcal{F}$. The output is called the partition function

$$Z(I) = \sum_{x_1, x_2, \ldots, x_n \in \{0,1\}} \prod f(x_{i_1}, x_{i_2}, \ldots, x_{i_k}),$$

where the product is over all constraints occurring in I. When $\mathcal{F} = \{f\}$, consisting of a single constraint function, we denote #CSP($\{f\}$) simply as #CSP(f).

3.1 The 0-1 Case and Nonnegative Boolean #CSP

In general, functions $f \in \mathcal{F}$ can take arbitrary complex values. If each f takes values 0, 1, then it is identified with a Boolean-valued constraint. In that case, $Z(I)$ counts the number of assignments *satisfying* all the Boolean constraints of I.

Typically $\mathcal{F}$ is a finite set and considered fixed. The complexity is measured in terms of the input size of I. For a finite and fixed $\mathcal{F}$ this input size is equivalent to n, up to a polynomial factor. Occasionally we will also allow $\mathcal{F}$ to be infinite, such as those sets containing all unary functions $\mathcal{U}$, in which case the input size must include the description of the constraints used in the input.

If each $f \in \mathcal{F}$ takes values 0, 1, then we have the following theorem.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Theorem 3.1 (Creignou–Hermann). *For every finite $\mathcal{F}$ consisting of 0-1 valued functions on Boolean variables, #CSP($\mathcal{F}$) is #P-complete, unless each $f \in \mathcal{F}$ is the 0-1 indicator function of some affine linear subspace over $\mathbb{Z}_2$, in which case it is in P.*

In Volume I we present the dichotomy theory only on the Boolean domain. Far reaching generalizations to general domains will be presented in Volume II [DG00, BD07, BG05, Bul13, Thu09, GT11, GGJT10, CCL13, DR13, CCL16, CC12].

A 0-1 indicator function of some affine linear subspace is just a Boolean conjunction of XOR functions on a (possibly empty) subset of variables and possibly a constant $1 \in \mathbb{Z}_2$. This includes the constant 0, 1 functions, the unary functions $\Delta_0 = [1, 0]$ and $\Delta_1 = [0, 1]$, the binary DISEQUALITY $[0, 1, 0]$, and EQUALITY of any arities $(=_k)$. When $\mathcal{F}$ has the form specified in the theorem, one can solve #CSP($\mathcal{F}$) in polynomial time by Gaussian elimination over the field $\mathbb{Z}_2$.

Definition 3.2. A function has *product type* if it can be expressed as a product of unary functions, binary EQUALITY functions ($(=_2) = [1, 0, 1]$), and binary DISEQUALITY functions ($(\neq_2) = [0, 1, 0]$), on not necessarily disjoint subsets of variables. We denote by $\mathscr{P}$ the set of all functions of *product type*.

Exercise: Show that a symmetric signature is in $\mathscr{P}$ iff it is a GEN-EQ, or a (scalar multiple of) binary DISEQUALITY, or a constant function.

Definition 3.3. A function is *pure affine* if it is a constant multiple of a 0-1 indicator function of some affine linear subspace over $\mathbb{Z}_2$.

If all functions in $\mathcal{F}$ take nonnegative values, then Theorem 3.1 can be generalized to the following theorem.

Theorem 3.4 (Dyer–Goldberg–Jerrum). *For every finite $\mathcal{F}$ consisting of nonnegative-valued functions, #CSP($\mathcal{F}$) is solvable in polynomial time in the following two cases, and is #P-hard otherwise:*

1. *Every function in $\mathcal{F}$ is of a* product type*; or*
2. *Every function in $\mathcal{F}$ is a* pure affine *function.*

For either tractable cases, the polynomial time algorithm is quite simple. The main claim is that everything else is #P-hard. Theorem 3.4 is concerned with nonnegative-valued functions. However, the tractability still holds without this restriction. This means in particular that functions in $\mathscr{P}$ can be complex

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

valued, i.e., the unary functions in the definition of a function in $\mathscr{P}$ can be complex valued. Any Gen-Eq is in $\mathscr{P}$. Similarly, the constant multiple in the definition of a pure affine function can be from $\mathbb{C}$.

It is also clear that Theorem 3.1 is a special case of Theorem 3.4.

However, allowing real or complex-valued product type or pure affine functions in Theorem 3.4 turns out to be not sufficient to encompass all tractable cases when functions in $\mathcal{F}$ are no longer restricted to those taking nonnegative values. When the functions $f \in \mathcal{F}$ take possibly negative or complex values, there are more nontrivial tractable #CSP($\mathcal{F}$) problems. This is made possible by algebraic cancelations, and they lead to novel efficient algorithms. The goal of this chapter is to prove an explicit dichotomy theorem for #CSP($\mathcal{F}$), Theorem 3.7, for any complex-valued function set $\mathcal{F}$. Theorem 3.4, and a fortiori Theorem 3.1, are special cases of this dichotomy theorem.

Before we move on to investigate Boolean #CSP in detail, we can restate Theorem 2.12 in terms of what can be transformed to product type $\mathscr{P}$. Note that a symmetric signature is in $\mathscr{P}$ iff it is a Gen-Eq, or a binary Disequality, or a degenerate function.

Theorem 3.5. *Let $\mathcal{F}$ be a set of nondegenerate symmetric signatures over $\mathbb{C}$. Then* Holant$^*(\mathcal{F})$ *is #P-hard, unless $\mathcal{F}$ satisfies the following conditions, in which case it is computable in polynomial time.*

1. *All signatures in $\mathcal{F}$ have arity ≤ 2.*
2. *There exists some $M \in \mathbf{GL}_2(\mathbb{C})$ such that $(=_2)M^{\otimes 2} \in \mathscr{P}$ and $\mathcal{F} \subseteq M\mathscr{P}$.*
3. *There exists $\lambda \in \{2\mathrm{i}, -2\mathrm{i}\}$, such that every signature $f \in \mathcal{F}$ of arity n satisfies the recurrence*

$$f_{k+2} = \lambda f_{k+1} + f_k, \quad \text{for} \quad 0 \leq k \leq n-2.$$

Remark: Case 1 here corresponds to class $\mathfrak{A}$ of Theorem 2.12. Case 2 corresponds to all of class $\mathfrak{B}$ (except type I$(1, \pm 2\mathrm{i})$) as well as class $\mathfrak{C}$. Case 3 corresponds to type I$(1, \pm 2\mathrm{i})$ of class $\mathfrak{B}$. As shown in Section 2.2, type I$(1, \pm 2\mathrm{i})$ is also equivalent to the following. Under the Z transformation, every $f \in \mathcal{F}$ has the form $f = Z[c, d, 0, \ldots, 0]$, or every $f \in \mathcal{F}$ has the form $f = Z[0, \ldots, 0, d, c]$, where $Z = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ \mathrm{i} & -\mathrm{i} \end{bmatrix}$. Recall also that under the covariant transformation by Z, the binary Equality is transformed to the binary Disequality: $(=_2)Z^{\otimes 2} = (\neq_2)$.

Exercise: Show that there exists some $M \in \mathbf{GL}_2(\mathbb{C})$ such that $(=_2)M^{\otimes 2}$ is a Gen-Eq and $\mathcal{F} \subseteq M\mathscr{P}$, iff there exists an orthogonal T that does this transformation.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Exercise: Show that there exists some $M \in \mathbf{GL}_2(\mathbb{C})$ such that $(=_2)M^{\otimes 2}$ is $(\neq_2)$ and $\mathcal{F} \subseteq M\mathscr{P}$, iff we may take $M = Z$ to do this transformation.

Exercise: Prove that Theorem 3.5 is an equivalent form of Theorem 2.12.

3.2 Affine Functions $\mathscr{A}$ and $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$

There is a simple relation between #CSP and Holant problems. We can represent an instance of a #CSP problem by a bipartite graph G where the left-hand side is labeled by variables and the right-hand side is labeled by constraints. We define a signature grid Ω on G by assigning an EQUALITY function to every variable node on the left-hand side, and every constraint node on the right-hand side has the given constraint function. Then Holant_Ω is exactly the same as the #CSP problem. In effect, the EQUALITY function on each variable node forces the incident edges to take the same value; this effectively reduces edge assignments in Holant_Ω to vertex assignments on the left-hand side in the #CSP problem. Thus #CSP problems are precisely the special case of Holant problems on bipartite graphs where every vertex on the left-hand side is assigned an EQUALITY function. Furthermore, the restriction on bipartiteness is not essential; given any constraint function set $\mathcal{F}$, it is easy to show that $\#\text{CSP}(\mathcal{F})$ is exactly the same as $\text{Holant}(\mathcal{F} \cup \mathscr{E})$, where $\mathscr{E} = \{=_k \mid k \geq 1\}$ is the set of all EQUALITY functions of any arity k. In other words, #CSP problems are Holant problems with the implicit assumption that all EQUALITY functions are freely available. This is similar to Holant* problems where we assume all unary functions are freely available. In fact, we will see that the presence of all EQUALITY functions already implies the presence of two special unary functions Δ_0 and Δ_1. Applying Δ_0 or Δ_1 on a variable is called *pinning*.

On the other hand, Holant problems can be considered as #CSP problems where every variable appears twice. We will show that the following three families of functions are tractable for Holant problems. Below we write a function as a vector, listing its values in lexicographic order of the assignments, which we will also call its truth table. Here $\mathrm{i} = \sqrt{-1}$.

$$\mathscr{F}_1 = \{\lambda([1,0]^{\otimes k} + \mathrm{i}^r[0,\ \ 1]^{\otimes k}) \mid \lambda \in \mathbb{C}, k = 1, 2, \ldots, \text{ and } r = 0, 1, 2, 3\};$$

$$\mathscr{F}_2 = \{\lambda([1,1]^{\otimes k} + \mathrm{i}^r[1,-1]^{\otimes k}) \mid \lambda \in \mathbb{C}, k = 1, 2, \ldots, \text{ and } r = 0, 1, 2, 3\};$$

$$\mathscr{F}_3 = \{\lambda([1,\ \mathrm{i}]^{\otimes k} + \mathrm{i}^r[1,\ -\mathrm{i}]^{\otimes k}) \mid \lambda \in \mathbb{C}, k = 1, 2, \ldots, \text{ and } r = 0, 1, 2, 3\}.$$

We explicitly list all the signatures in $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ up to an arbitrary constant multiple from $\mathbb{C}$:

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

1. $[1, 0, \ldots, 0, \pm 1]$; $(\mathscr{F}_1, r = 0, 2)$
2. $[1, 0, \ldots, 0, \pm \mathrm{i}]$; $(\mathscr{F}_1, r = 1, 3)$
3. $[1, 0, 1, 0, \ldots, 0 \text{ or } 1]$; $(\mathscr{F}_2, r = 0)$
4. $[1, -\mathrm{i}, 1, -\mathrm{i}, \ldots, (-\mathrm{i}) \text{ or } 1]$; $(\mathscr{F}_2, r = 1)$
5. $[0, 1, 0, 1, \ldots, 0 \text{ or } 1]$; $(\mathscr{F}_2, r = 2)$
6. $[1, \mathrm{i}, 1, \mathrm{i}, \ldots, \mathrm{i} \text{ or } 1]$; $(\mathscr{F}_2, r = 3)$
7. $[1, 0, -1, 0, 1, 0, -1, 0, \ldots, 0 \text{ or } 1 \text{ or } (-1)]$; $(\mathscr{F}_3, r = 0)$
8. $[1, 1, -1, -1, 1, 1, -1, -1, \ldots, 1 \text{ or } (-1)]$; $(\mathscr{F}_3, r = 1)$
9. $[0, 1, 0, -1, 0, 1, 0, -1, \ldots, 0 \text{ or } 1 \text{ or } (-1)]$; $(\mathscr{F}_3, r = 2)$
10. $[1, -1, -1, 1, 1, -1, -1, 1, \ldots, 1 \text{ or } (-1)]$. $(\mathscr{F}_3, r = 3)$

We can show that Holant_Ω for any $\Omega = (G, \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3)$ is computable in P. These functions are all related to each other by holographic reductions. We will prove this tractability by a more general tractability theorem that covers nonsymmetric functions as well.

We note that expressions in complex numbers appear naturally, even for real-valued functions. The special case where $r = 1$, $k = 2$ and $\lambda = (1 + \mathrm{i})^{-1}$ in $\mathscr{F}_3$ is noteworthy. In this case we get a real-valued binary symmetric function $H = [1, 1, -1]$. In other words, $H(0, 0) = H(0, 1) = H(1, 0) = 1$ and $H(1, 1) = -1$. The matrix form of this function is the Hadamard matrix $H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$. Furthermore, if we take $r = 0$, any k and $\lambda = 1$ in $\mathscr{F}_1$ we get the EQUALITY function on k variables. If $\Omega = (G, \mathcal{F})$ with $\mathcal{F}$ consisting of exactly the function H and all EQUALITY functions, then Holant_Ω is computing the partition function of #CSP(H). This special case where the constraint function set in #CSP consists of a single binary (symmetric) function is also known as graph homomorphism $Z_H(G)$, or a spin system, on (undirected) graph G. What is the value of $Z_H(G)$? If X (respectively Y) is the number of (vertex subset-) induced subgraphs of G with an even (respectively odd) number of edges, then $Z_H(G) = X - Y$. Since trivially $X + Y = 2^n$, we have $X = \frac{2^n + Z_H(G)}{2}$ and $Y = \frac{2^n - Z_H(G)}{2}$; thus the problem Z_H essentially computes the number of induced subgraphs with an even (respectively odd) number of edges.

A function of arity k can be expressed by its truth table of length 2^k. Define

$$\mathscr{D} = \{f \mid f = [a_1, b_1] \otimes [a_2, b_2] \otimes \cdots \otimes [a_k, b_k], \text{ for some } a_i, b_i \in \mathbb{C}\}$$

to be the set of functions that can be expressed as a tensor product of unary functions; that is, a function in $\mathscr{D}$ on k variables is the product of k unary functions applied to its k variables separately. Functions in $\mathscr{D}$ are called *degenerate*. This notion of degeneracy agrees with the definition of degeneracy for symmetric functions. A binary function is in $\mathscr{D}$ iff its corresponding matrix is singular. Note that $\mathscr{D}$ is a subset of $\mathscr{P}$, the signatures of product type.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

The main result in this chapter is a complexity dichotomy theorem, Theorem 3.7, for complex-valued Boolean #CSP. The family $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ can be naturally generalized to include unsymmetric functions, called the affine family $\mathscr{A}$. In fact, the family $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ consists of precisely the unary functions of $\mathscr{A}$, and (for $\lambda \neq 0$ and $k \geq 2$) symmetric nondegenerate functions of $\mathscr{A}$ of arity ≥ 2. Of course unary functions are also symmetric (but are considered degenerate). The dichotomy theorem says that a Boolean #CSP problem #CSP($\mathcal{F}$) is tractable if *either* all its constraint functions $\mathcal{F}$ are of product type $\mathscr{P}$, *or* all are from this affine family $\mathscr{A}$; otherwise it is #P-hard.

Suppose f is a function on input variables $x_1, x_2, \ldots, x_k$. For $1 \leq s \leq k$ and $c = 0, 1$, $f^{x_s=c}$ denotes the function $f^{x_s=c}(x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_k) = f(x_1, \ldots, x_{s-1}, c, x_{s+1}, \ldots, x_k)$, and $f^{x_s=*}$ denotes the function $f^{x_s=*}(x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_k) = \sum_{x_s=0,1} f(x_1, \ldots, x_k)$.

For a constraint function f we define its underlying relation, also called its support , by

$$R_f = \{X \in \{0, 1\}^k \mid f(X) \neq 0\}.$$

We say a relation $R \subseteq \{0, 1\}^k$ is affine if it is an affine subspace over $\mathbb{Z}_2$ (including possibly the empty space). It is composed of solutions to a system of linear equations. Equivalently, it satisfies the property that if $\alpha, \beta, \gamma \in R$, then the bitwise XOR string $\alpha \oplus \beta \oplus \gamma \in R$. To see this, first suppose R is the set of solutions to $AX = r$ for some matrix A and vector r over $\mathbb{Z}_2$. Then clearly for any $\alpha, \beta, \gamma \in R$, $A(\alpha \oplus \beta \oplus \gamma) = r \oplus r \oplus r = r$, and $\alpha \oplus \beta \oplus \gamma \in R$. Now suppose R satisfies this property. Suppose $R \neq \emptyset$. Define $L_R = \{\alpha \oplus \beta \mid \alpha, \beta \in R\}$. Then L_R is a linear subspace of $\mathbb{Z}_2^k$. Take any $\alpha_0 \in R$. Then $R \subseteq L_R + \alpha_0 = \{X \oplus \alpha_0 \mid X \in L_R\}$, since for any $\alpha \in R$, $\alpha = \alpha \oplus \alpha_0 \oplus \alpha_0$. Also, $L_R + \alpha_0 \subseteq R$ by the fact that R satisfies the property.

If R_f is affine, we say f has affine support . We also view relations as functions from $\{0, 1\}^k$ to $\{0, 1\}$.

Now we define the affine family of functions $\mathscr{A}$. Let X denote the $k + 1$ dimensional column vector $(x_1, x_2, \ldots, x_k, 1)^{\mathrm{T}}$ over the field $\mathbb{Z}_2$. Suppose A is a matrix over $\mathbb{Z}_2$. χ_{AX} denotes the affine relation on inputs $x_1, x_2, \ldots, x_k$, whose value is 1 if AX is the zero vector, and 0 otherwise.

Definition 3.6. A function is of *affine type* if it can be expressed as

$$\lambda \cdot \chi_{AX} \cdot \mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)},$$

where $\lambda \in \mathbb{C}$, $\mathrm{i} = \sqrt{-1}$, each L_j is an integer 0-1 indicator function of the form $\langle \alpha_j, X \rangle$, where α_j is a $k + 1$ dimensional vector over $\mathbb{Z}_2$, and the dot product $\langle \cdot, \cdot \rangle$ is computed over $\mathbb{Z}_2$. The set of all functions of *affine type* is denoted by $\mathscr{A}$.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

We may compute the dot product as an ordinary integer dot product, and then take the value mod 2, producing an integer value 0 or 1. The additions among $L_j(X)$ are the usual addition in $\mathbb{Z}$. It can be computed mod 4, but not mod 2. If we solve the affine linear system $AX = 0 \bmod 2$ we can replace dependent variables and assume the expressions $L_j(X)$ are over a set of free variables $Y \subseteq X$. We usually will omit any nonzero constant factor λ, since it does not affect complexity and can be ignored.

3.3 A Dichotomy for Boolean #CSP

Theorem 3.7. *Suppose $\mathcal{F}$ is a set of functions mapping Boolean inputs to complex numbers. If $\mathcal{F} \subseteq \mathscr{A}$ or $\mathcal{F} \subseteq \mathscr{P}$, then #CSP($\mathcal{F}$) is computable in polynomial time. Otherwise, #CSP($\mathcal{F}$) is* #P*-hard.*

Proof Outline: The polynomial time algorithm for #CSP($\mathscr{P}$) is easy. Section 3.4 gives a polynomial time algorithm for #CSP($\mathscr{A}$). Cancelations when summing for functions in $\mathscr{A}$ play an essential role. The starting point of the hardness result is Theorem 3.14, which says that if $\mathcal{F}$ contains only one binary symmetric function and is not in $\mathscr{A} \cup \mathscr{P}$, then the #CSP problem is #P-hard. In Lemma 3.18, we prove that #CSP(f) is #P-hard unless f has affine support. This structure is essential in the proof of Lemmas 3.19 and 3.20, the two key lemmas of the hardness reduction. The common strategy of Lemmas 3.19 and 3.20 is to reduce the arity of a given function. In Lemma 3.19, we prove that given a function f not in $\mathscr{A}$, we can simulate (in polynomial time) a unary function $f' \notin \mathscr{A}$; In Lemma 3.20, we prove that given a function g not in $\mathscr{P}$, we can simulate (in polynomial time) a binary or a ternary function $g' \notin \mathscr{P}$. Then we prove that #CSP($\{f', g'\}$) is #P-hard. To prove this, we show that we can always combine functions f' and g' to realize a binary symmetric function that is not in $\mathscr{A} \cup \mathscr{P}$.

Exercise: The tractability criterion in Theorem 3.7 should not be confused with the statement $\mathcal{F} \subseteq \mathscr{A} \cup \mathscr{P}$. Show that there are $\mathcal{F}$ satisfying $\mathcal{F} \subseteq \mathscr{A} \cup \mathscr{P}$ and yet #CSP($\mathcal{F}$) is #P-hard.

There is an alternative expression of a function in $\mathscr{A}$. We claim that the expressibility as

$$\lambda \chi_{AX} \mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)}$$

is equivalent to an expression of the form $\lambda \chi_{AX} \mathrm{i}^{Q(X)}$, where Q is a homogeneous quadratic polynomial in $\mathbb{Z}[X]$ with the additional requirement that every cross term $x_s x_t$ has an even coefficient, where $s \neq t$. To see this we observe that each

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

$L_j(X)$ as an integer sum mod 2 can be replaced by $(L_j(X))^2$ as an integer sum mod 4, since $L_j(X) = 0, 1 \pmod 2$ iff $(L_j(X))^2 = 0, 1 \pmod 4$. After this, all cross terms have an even coefficient, and i^{x_s} can be replaced by $\mathrm{i}^{x_s^2}$, for $x_s = 0, 1$. Conversely, we can express Q mod 4 as a sum of squares of affine forms of X, using the extra condition that all cross terms have even coefficients.

Exercise: Prove that the class of function $\mathscr{A}$ is not changed by the following modifications in its definition:

- Q is a polynomial of degree ≤ 2 (with possible linear and constant terms, but cross terms are still required have even coefficients).
- Q is a multilinear polynomial of degree ≤ 2, where cross terms have even coefficients.

Exercise: Prove that a function $\{0, 1\}^k \to \mathbb{Z}_4$ has a unique multilinear polynomial expression. In particular, suppose $f = \lambda \chi_{AX} \mathrm{i}^{Q(Y)}$, where $Q(Y)$ is a multilinear polynomial on free variables $Y \subseteq X$ of its support. If Q has degree > 2, or has a cross term with an odd coefficient, then $f \notin \mathscr{A}$.

Exercise: Derive a normal form for any real-valued function $f \in \mathscr{A}$.

The class $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ is the restriction of $\mathscr{A}$ to symmetric functions. More precisely, $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ consists of scalar multiples of all unary or non-degenerate symmetric functions in $\mathscr{A}$. We first show that $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3 \subset \mathscr{A}$. Omitting the nonzero constants, the functions in $\mathscr{F}_1$ can be expressed as $\chi_{x_1=x_2=\ldots=x_k} \mathrm{i}^{rx_1^2}$. The functions in $\mathscr{F}_2$ can be expressed as $\chi_{\sum_j x_j=0}$, or $\chi_{\sum_j x_j=1}$, or $\mathrm{i}^{(\sum_j x_j)^2}$, or $(-\mathrm{i})^{(\sum_j x_j)^2} = \mathrm{i}^{3(\sum_j x_j)^2}$, respectively. The functions in $\mathscr{F}_3$ can be expressed as $\chi_{\sum_j x_j=0} \mathrm{i}^{\sum_j x_j^2}$, or $\chi_{\sum_j x_j=1} \mathrm{i}^{\sum_j x_j^2 - 1} = -\mathrm{i} \chi_{\sum_j x_j=1} \mathrm{i}^{\sum_j x_j^2}$, or $(-1)^{\sum_{s<t} x_s x_t} = \mathrm{i}^{2\sum_{s<t} x_s x_t}$, or $(-1)^{\sum_{s<t} x_s x_t + \sum_j x_j} = \mathrm{i}^{2\sum_j x_j^2 + 2\sum_{s<t} x_s x_t}$, respectively.

Conversely suppose $f(x_1, \ldots, x_k) \in \mathscr{A}$ is symmetric. Let $S = \{x \in \mathbb{Z}_2^k \mid f(x) \neq 0\}$ be its affine support. We claim that S must be of the following form: $\emptyset$, $\mathbb{Z}_2^k$, $\{(0, \ldots, 0)\}$, $\{(1, \ldots, 1)\}$, $\{(0, \ldots, 0), (1, \ldots, 1)\}$, $\{(0, 1), (1, 0)\}$, and $\{x \in \mathbb{Z}_2^k \mid \sum_j x_j = \epsilon \bmod 2\}$, where $\epsilon = 0$ or 1. If f is identically zero, then $S = \emptyset$. We assume f is not identically zero.

Consider all affine linear equations over $\mathbb{Z}_2$ that are valid on S, which must be closed under permutations of variables by the symmetry of f. If the set of equations is empty, then $S = \mathbb{Z}_2^k$. So assume the set of equations is nonempty. By symmetry, if there is an equation involving only one variable $x_i = \epsilon$, for some $1 \leq i \leq k$, then for all $1 \leq j \leq k$, $x_j = \epsilon$. Thus $S = \{(0, \ldots, 0)\}$ or $\{(1, \ldots, 1)\}$. Now we assume no equation has only one variable, but there is an equation of two variables $x_i + x_j = \epsilon$, for some $1 \leq i < j \leq k$, then

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

by symmetry, this holds for all $i \neq j$. If $k > 2$ and $\epsilon = 1$, we would have $x_1 + x_2 = 1$ and $x_1 + x_2 = (x_1 + x_3) + (x_2 + x_3) = 0 \bmod 2$, and S would be empty. Thus for $k > 2$, we must have $\epsilon = 0$ and we have $x_1 = x_2 = \cdots = x_k$. Thus $S = \{(0, \ldots, 0), (1, \ldots, 1)\}$. If $k = 2$, additionally we also have possibly $S = \{(0, 1), (1, 0)\}$. Finally, we may assume all equations involve at least three variables. Then $k \geq 3$, and every equation must involve all k variables. Otherwise, we have $x_1 + \cdots + x_j = \epsilon$ and $x_1 + \cdots + x_{j-1} + x_{j+1} = \epsilon$, for some $j < k$, whose sum gives $x_j + x_{j+1} = 0 \bmod 2$ having only two variables. Thus $S = \{x \in \mathbb{Z}_2^k \mid \sum_j x_j = \epsilon \bmod 2\}$.

Let $f = \chi_S \mathrm{i}^{Q(x_1,\ldots,x_k)} \in \mathscr{A}$ be symmetric, where $Q(x_1, \ldots, x_k) = \sum_j a_j x_j^2 + 2\sum_{s<t} b_{s,t} x_s x_t$. Assume f is either unary or not degenerate if it has arity ≥ 2. The case when the support S has at most two points is easy to prove. Suppose $S = \mathbb{Z}_2^k$. Then it is easy to see that all a_j are the same, and all $b_{s,t}$ are the same, by setting one or two $x_j = 1$ and the other x_j's to 0. Then $Q(x_1, \ldots, x_k)$ takes the form $a\sum_j x_j^2 + 2b\sum_{s<t} x_s x_t = a(\sum_j x_j)^2 + 2(b-a)\sum_{s<t} x_s x_t$. If we denote by $\Sigma = \sum_j x_j$ as an integer sum, and we denote by $(\Sigma \bmod 2)$ the integer value 0 or 1, then $\mathrm{i}^Q = \mathrm{i}^{a(\Sigma \bmod 2)}(-1)^{(b-a)(\binom{\Sigma}{2} \bmod 2)}$. By listing the values according to Σ, we find that either the function f is degenerate, or $f \in \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$.

Now suppose S is specified by $\sum_j x_j = \epsilon \bmod 2$, where $\epsilon = 0, 1$. We first substitute $x_k = \epsilon + \sum_{j=1}^{k-1} x_j \bmod 2$ in the linear forms L_j in the expression $\mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)}$, which can then be turned into the form $\mathrm{i}^{Q'(x_1,\ldots,x_{k-1})}$, where $Q' = \sum_{j=1}^{k-1} a'_j x_j^2 + 2\sum_{1\leq s<t\leq k-1} b'_{s,t} x_s x_t$ is a quadratic form as above. The remaining proof is similar.

Suppose f is a symmetric degenerate function. Then f has the form $[a, b]^{\otimes n}$ for some unary function $[a, b]$ and $n \geq 1$. If either $a = 0$ or $b = 0$, then clearly $f \in \mathscr{A}$, and also $[a, b] \in \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$. Suppose $ab \neq 0$. Then f is a nonzero scalar multiple of $[1, b/a]^{\otimes n}$. Note that the ratio of any two nonzero entries of a function in $\mathscr{A}$ is a power of i; it follows that f belongs to $\mathscr{A}$ iff b/a is a power of i, and also iff $[1, b/a] \in \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$. We conclude that for any $f = [a, b]^{\otimes n}$, $f \in \mathscr{A}$ iff $[a, b] \in \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$.

We summarize the above discussion as the following theorem.

Theorem 3.8. *A function f belongs to $\mathscr{A}$ iff it can be expressed as $\lambda\chi_{AX}\mathrm{i}^{Q(x_1,\ldots,x_k)}$, where Q is a homogeneous quadratic polynomial over $\mathbb{Z}$ with the additional requirement that every cross term $x_s x_t$ has an even coefficient, where $s \neq t$. We may also use all, not necessarily homogeneous, polynomials over $\mathbb{Z}$ of degree ≤ 2, with the same requirement on cross terms.*

The class $\mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$ consists of scalar multiples of all unary and all nondegenerate symmetric functions in $\mathscr{A}$. For any degenerate $f = [a, b]^{\otimes n}$, $f \in \mathscr{A}$ iff $[a, b] \in \mathscr{F}_1 \cup \mathscr{F}_2 \cup \mathscr{F}_3$.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

3.4 Tractable Cases

We first show that #CSP($\mathscr{P}$) is tractable. Each constraint function in an instance I of #CSP($\mathscr{P}$) is a product of unary functions, binary EQUALITY functions ($=_2$), and binary DISEQUALITY functions ($\neq_2$). We may replace each function by its factors as separate constraints. For the new instance of the #CSP, define a graph G: The vertices are variables, and the edges are ($=_2$). On each connected component we may assume no ($\neq_2$) is applied to two variables; otherwise $Z(I) = 0$. Now define another graph G': The vertices are the connected components of G, and there is an edge between two components C_1 and C_2 iff there is a constraint $(\neq_2)(x, y)$ applied on some x in C_1 and y in C_2. Then either G' is bipartite or $Z(I) = 0$. Now suppose G' is bipartite. Take the union of components of G belonging to the same connected component of G'. Then there are exactly two truth assignments to all the variables in this union, propagated by ($=_2$) or ($\neq_2$). We can easily calculate these two values by mutliplying the functional evaluations, taking into account the unary functions. From these we can calculate $Z(I)$. Hence, #CSP($\mathscr{P}$) is computable in polynomial time.

Now we analyze #CSP($\mathscr{A}$). We will ignore the global constant factor λ for each $F \in \mathscr{A}$. Firstly, we show how to get rid of the factor χ_{AX}.

Lemma 3.9. *Let* $F(x_1, x_2, \ldots, x_k) = \chi_{AX}\mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)} \in \mathscr{A}$. *If* $AX = 0$ *is infeasible over* $\mathbb{Z}_2$, *then* $\sum_{x_1,x_2,\ldots,x_k} F = 0$. *Suppose* $AX = 0$ *is feasible. Then in polynomial time, we can construct another function* $H(y_1, y_2, \ldots, y_s) = \mathrm{i}^{L'_1(Y)+L'_2(Y)+\cdots+L'_n(Y)} \in \mathscr{A}$, *such that* $0 \le s \le k$, *and* $\sum_{x_1,x_2,\ldots,x_k} F = \sum_{y_1,y_2,\ldots,y_s} H$.

Proof. In polynomial time we can solve the linear system $AX = 0$ over $\mathbb{Z}_2$, and decide if it is feasible. Suppose $AX = 0$ is feasible. Without loss of generality, we can assume that $\{y_1, y_2, \ldots, y_s\} \subseteq \{x_1, x_2, \ldots, x_k\}$ is a set of independent variables over $\mathbb{Z}_2$ and the others are dependent variables, where $0 \le s \le k$. Each dependent variable can be expressed by an affine linear sum of $y_1, y_2, \ldots, y_s$. For each $L_j(X)$, we can substitute all the dependent variables and get a 0-1 indicator function of $y_1, y_2, \ldots, y_s$, which we denote by $L'_j(Y)$. So we have

$$\sum_{x_1,x_2,\ldots,x_k} \chi_{AX}\mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)} = \sum_{y_1,y_2,\ldots,y_s} \mathrm{i}^{L'_1(Y)+L'_2(Y)+\cdots+L'_n(Y)}.$$

□

The following lemma gives a key property of any function $\mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)} \in \mathscr{A}$. We will also use its alternative form $\mathrm{i}^{Q(X)}$. This property plays an important role in both the tractability proof and the hardness proof.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Lemma 3.10. *Let $F = \mathrm{i}^{Q(X)} \in \mathscr{A}$. Let $G(x_2, \ldots, x_k) = F^{x_1=1}/F^{x_1=0}$. Then exactly one of the following two statements holds:*

1. (Congruity) *There exists a constant $c \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$ such that for all $x_2, x_3, \ldots, x_k \in \{0, 1\}$ we have $G(x_2, x_3, \ldots, x_k) = c$;*
2. (Semicongruity) *There exists a constant $c \in \{1, \mathrm{i}\}$ and an affine subspace S of $\mathbb{Z}_2^{k-1}$ with $\dim S = k - 2$, such that*

$$G(x_2, x_3, \ldots, x_k) = c \text{ on } S, \quad \text{and} \quad G(x_2, x_3, \ldots, x_k) = -c \text{ on } \mathbb{Z}_2^{k-1} - S.$$

Proof. Clearly G has the form

$$G(x_2, \ldots, x_k) = \mathrm{i}^{a_{11}}(-1)^{a_{12}x_2 + \ldots + a_{1k}x_k}.$$

If $a_{12} = \cdots = a_{1k} = 0 \bmod 2$, then we have congruity. Suppose not all $a_{12} = \cdots = a_{1k} = 0 \bmod 2$; then $a_{12}x_2 + \cdots + a_{1k}x_k = 0 \bmod 2$ defines a linear subspace of $\mathbb{Z}_2^{k-1}$ of dimension $k - 2$, and we have semicongruity. □

Theorem 3.11. *#CSP($\mathscr{A}$) is polynomial time computable.*

Proof. We first observe that $\mathscr{A}$ is closed under multiplication. Therefore, given an instance of #CSP($\mathscr{A}$), the value of the output can be expressed as the summation on a single function $F = \chi_{AX}\mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)} \in \mathscr{A}$. We also note that if $F \in \mathscr{A}$, so is $F^{x_s=c}$, for $c = 0, 1$.

In each step of our algorithm, we reduce the number of variables by at least one and still get a summation of this form.

If the linear system $AX = 0$ over $\mathbb{Z}_2$ is infeasible, the function is a totally zero function and we just output 0. Otherwise by Lemma 3.9 we can remove the factor χ_{AX} and possibly decrease the number of variables at the same time, and assume it has the form $F = \mathrm{i}^{L_1(X)+L_2(X)+\cdots+L_n(X)}$. We apply Lemma 3.10 to remove x_1. There are three cases.

Case 1: We have congruity in Lemma 3.10. Then $F^{x_1=1}/F^{x_1=0}$ is a constant c, and

$$\sum_{x_1,x_2,\ldots,x_k} F = (1 + c) \cdot \sum_{x_2,x_3,\ldots,x_k} F^{x_1=0}.$$

So we get a new summation $\sum_{x_2,x_3,\ldots,x_k} F^{x_1=0}$ and have removed a variable x_1.

Case 2: We have semicongruity in Lemma 3.10, and $c = 1$. Then on the affine subspace S, the ratio $F^{x_1=1}/F^{x_1=0} = 1$, and on the complementary subspace $\mathbb{Z}_2^{k-1} - S$ the ratio $F^{x_1=1}/F^{x_1=0} = -1$. For all $(x_2, x_3, \ldots, x_k) \in \mathbb{Z}_2^{k-1} - S$, the terms cancel, $F^{x_1=1}(x_2, x_3, \ldots, x_k) + F^{x_1=0}(x_2, x_3, \ldots, x_k) = 0$. On S,

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

the terms are equal. It follows that

$$\sum_{x_1,x_2,\ldots,x_k} F = 2 \sum_{x_2,x_3,\ldots,x_k} \chi_S F^{x_1=0}.$$

Note that $\chi_S F^{x_1=0}$ is also a function in $\mathscr{A}$, so we get a new summation of this form and have removed a variable x_1.

Case 3: We have semicongruity in Lemma 3.10, and $c = \mathrm{i}$. Then for all $(x_2, x_3, \ldots, x_k)$ in the affine subspace S, we have $F^{x_1=1}/F^{x_1=0} = \mathrm{i}$, and in $\mathbb{Z}_2^{k-1} - S$, we have $F^{x_1=1}/F^{x_1=0} = -\mathrm{i}$. It follows that

$$\sum_{x_1,x_2,\ldots,x_k} F = \sum_{S} (1+\mathrm{i}) F^{x_1=0} + \sum_{\mathbb{Z}_2^{k-1}-S} (1-\mathrm{i}) F^{x_1=0}.$$

Now we make a simple but crucial observation. The ratio of $1+\mathrm{i}$ and $1-\mathrm{i}$ is exactly i, i.e., $\mathrm{i} = \frac{1+\mathrm{i}}{1-\mathrm{i}}$. As a result we can rewrite the two sums as follows:

$$\sum_{x_1,x_2,\ldots,x_k} F = \sum_{S} (1-i) \cdot F^{x_1=0} \cdot \mathrm{i}^{L(X')} + \sum_{\mathbb{Z}_2^{k-1}-S} (1-i) \cdot F^{x_1=0} \cdot \mathrm{i}^{L(X')},$$

where $L(X')$, on $X' = (x_2, x_3, \ldots, x_k, 1)$, is a 0-1 indicator function that takes value 1 on S and 0 on $\mathbb{Z}_2^{k-1} - S$. Thus we can combine the two sums and get

$$\sum_{x_1,x_2,\ldots,x_k} F = (1-i) \cdot \sum_{x_2,x_3,\ldots,x_k} \left(F^{x_1=0} \cdot \mathrm{i}^{L(X')} \right).$$

Note that $F^{x_1=0} \cdot \mathrm{i}^{L(X')}$ is also a function in $\mathscr{A}$. So we get a new summation of this form and have removed a variable x_1.

After at most k steps we can eliminate all the variables and obtain the value of the initial summation. Both k and n are bounded by input size. In each iteration, we either resolve a linear system $AX = 0$ or compute a linear equation by Lemma 3.10 representing the affine subspace S, both of which can be done in polynomial time. After each iteration, the formula inside the summation gets at most one more factor $\mathrm{i}^{L(X')}$ or χ_S, so the whole algorithm is in polynomial time. □

3.5 Hardness

In this section, we prove the hardness part of Theorem 3.7. We start with two useful lemmas that provide tools for constructing reductions.

In #CSP problems, if we have a constraint function F, and we let a variable x_j of F not to occur in any other constraint, then we get $F^{x_j=*}$, which is just $F^{x_j=0} + F^{x_j=1}$. This observation proves the following lemma.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Lemma 3.12. *For any $\mathcal{F}$ and F, and for any variable x_j of F,*

$$\#\mathrm{CSP}(\mathcal{F} \cup \{F^{x_j=*}\}) \leq_{\mathrm{T}} \#\mathrm{CSP}(\mathcal{F} \cup \{F\}).$$

The presence of all EQUALITY gates in #CSP already provides two important unary functions, Δ_0 and Δ_1. Applying Δ_0 or Δ_1 to a variable fixes that variable to the constant 0 or 1 respectively, and thus it is called *pinning*. We can simulate Δ_0 and Δ_1 by the following pinning lemma [DGJ09].

Lemma 3.13 (Dyer–Goldberg–Jerrum). *For any $\mathcal{F}$, $\#\mathrm{CSP}(\mathcal{F} \cup \{\Delta_0, \Delta_1\}) \leq_{\mathrm{T}} \#\mathrm{CSP}(\mathcal{F})$. In particular, for any F and $c \in \{0, 1\}$,*

$$\#\mathrm{CSP}(\mathcal{F} \cup \{F^{x_j=c}\}) \leq_{\mathrm{T}} \#\mathrm{CSP}(\mathcal{F} \cup \{F\}).$$

Proof. Given an instance I of $\#\mathrm{CSP}(\mathcal{F} \cup \{\Delta_0, \Delta_1\})$, let V_0 and V_1 denote the set of variables to which Δ_0 and Δ_1 are applied, respectively. If $V_0 \cap V_1 \neq \emptyset$, then there are no satisfying assignments and the partition function $Z(I) = 0$. We assume $V_0 \cap V_1 = \emptyset$. Let $V_2 = V - (V_0 \cup V_1)$ be the set of other variables. We define an instance I' of $\#\mathrm{CSP}(\mathcal{F})$ that has the variable set $V_2 \cup \{y_0, y_1\}$, where y_0, y_1 are two new variables. The constraints of I' are exactly the same as those of I, except (1) every variable in V_0 (respectively V_1) is replaced by y_0 (respectively y_1), and (2) the constraints of Δ_0 and Δ_1 are not applied. Then $Z(I')$ can be expressed as the sum of four quantities,

$$Z(I') = \sum_{(b_0,b_1)\in\{0,1\}^2} Z(I' \mid b_0 b_1),$$

where $Z(I' \mid b_0 b_1)$ is the sum in $Z(I')$ restricted to assigning (y_0, y_1) to $(b_0, b_1) \in \{0, 1\}^2$.

Now we define a second instance I'' of $\#\mathrm{CSP}(\mathcal{F})$ that has the variable set $V_2 \cup \{y\}$, where y is a new variable. It is the same as I' except we identify y_0 and y_1 by y. Then clearly

$$Z(I'') = \sum_{b_0=b_1} Z(I' \mid b_0 b_1) = Z(I' \mid 00) + Z(I' \mid 11).$$

Hence

$$Z(I') - Z(I'') = Z(I' \mid 01) + Z(I' \mid 10). \tag{3.1}$$

Obviously, $Z(I) = Z(I' \mid 01)$. If every function $F \in \mathcal{F}$ has *complementary invariance*, i.e., if F has arity k, then for every input $(b_1, \ldots, b_k) \in \{0, 1\}^k$, $F(b_1, \ldots, b_k) = F(\overline{b_1}, \ldots, \overline{b_k})$, where $\overline{b_i} = 1 - b_i$, then clearly $Z(I') - Z(I'') = 2Z(I)$, and the lemma is proved.

Suppose for some $F \in \mathcal{F}$ of arity k, and some $(b_1, \ldots, b_k) \in \{0, 1\}^k$, $F(b_1, \ldots, b_k) \neq F(\overline{b_1}, \ldots, \overline{b_k})$, we will define two more instances of

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

#CSP($\mathcal{F}$). The first is I'_b, which is the same as I', except it has one more constraint F applied to the sequence of variables $(y_{b_1}, \ldots, y_{b_k})$. Note that for any assignment in I'_b, if (y_0, y_1) is assigned to $(0, 1)$, then $F(y_{b_1}, \ldots, y_{b_k})$ will evaluate to $F(b_1, \ldots, b_k)$, and if (y_0, y_1) is assigned to $(1, 0)$, then $F(y_{b_1}, \ldots, y_{b_k})$ will evaluate to $F(\overline{b_1}, \ldots, \overline{b_k})$. Of course if (y_0, y_1) is assigned to $(0, 0)$ or $(1, 1)$, then $F(y_{b_1}, \ldots, y_{b_k})$ will evaluate to $F(0, \ldots, 0)$ and $F(1, \ldots, 1)$ respectively.

The second instance we define is I''_b, which is the same as I'', except it has one more constraint F applied to the sequence of variables $(y, \ldots, y)$. It is now clear that

$$Z(I'_b) - Z(I''_b) = Z(I' \mid 01)F(b_1, \ldots, b_k) + Z(I' \mid 10)F(\overline{b_1}, \ldots, \overline{b_k}). \quad (3.2)$$

From (3.1) and (3.2) we can solve $Z(I) = Z(I' \mid 01)$. □

Remark: Before Lemma 3.13 we have used three reduction methods: gadget construction, holographic transformation, and polynomial interpolation. Suppose f is such a reduction from a problem Π to a problem Π'. When the reduction is applied to an instance Ω of Π, f defines a sequence $\{\Omega'_i\}$ of instances of Π', and outputs $\Omega'_1, \Omega'_2, \ldots, \Omega'_m$, where both m and the sizes of Ω'_i, $(1 \leq i \leq m)$, are polynomially bounded in the size of Ω. (That there might be multiple output instances, i.e., $m > 1$, is typical for polynomial interpolations.) The three reduction methods all satisfy the following *locality* property: If the reduction is applied to an instance of Π that consists of d identical copies of Ω, denoted by $d\Omega$, it will output a polynomial number of instances of Π', where each instance consists of d identical copies of Ω'_i, namely $d\Omega'_1, d\Omega'_2, \ldots, d\Omega'_{m'}$ (m' is polynomially bounded in the size of $d\Omega$, possibly a larger polynomial than m). Clearly this property is closed under composition.

However, this locality property is not shared by the pinning reduction in Lemma 3.13. For a technical reason in Chapter 6 (Theorem 6.16, specifically Lemma 6.23), we will need this locality property, and thus ascertain that a reduction chain exists from a basic #P-hard problem such as Vertex Cover that does not use the pinning reduction of Lemma 3.13. For this reason, we will make annotating remarks in the following for #P-hardness results that can be proved without using the pinning reduction. For the convenience of addressing this issue, we denote by #CSP$^c(\mathcal{F})$ the problem #CSP($\mathcal{F} \cup \{\Delta_0, \Delta_1\}$), which is logically equivalent to #CSP($\mathcal{F}$) by Lemma 3.13.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

3.5.1 One Symmetric Binary Function

For hardness, we first establish the following theorem, which says that Theorem 3.7 holds for a single symmetric binary function.

Theorem 3.14. *If* $[a, b, c] \notin \mathscr{A} \cup \mathscr{P}$*, then* #CSP($[a, b, c]$) *is #P-hard. Explicitly, for any binary function* $[a, b, c]$*, where* $a, b, c \in \mathbb{C}$*, if* #CSP($[a, b, c]$) *is not #P-hard, then* $[a, b, c]$ *takes the form:* $[x, 0, y]$*,* $[0, x, 0]$*,* $[x^2, xy, y^2]$*,* $x[1, \pm\mathrm{i}, 1]$ *or* $x[1, \pm 1, -1]$*.*

For #CSP^c($[a, b, c]$) $\equiv$ #CSP($[a, b, c], \Delta_0, \Delta_1$)*, if* $[a, b, c] \notin \mathscr{A} \cup \mathscr{P}$*, then* #CSP^c($[a, b, c]$) *is #P-hard by a reduction that does not use Lemma 3.13.*

Before proving Theorem 3.14, we first note that every one of the five listed exceptional cases are in $\mathscr{A} \cup \mathscr{P}$, and it can be checked directly that all binary symmetric functions in $\mathscr{A} \cup \mathscr{P}$ take one of these five forms.

In this proof we use an important reduction method called *polynomial interpolation*, first introduced by Valiant [Val79b]. The method involves setting up and then solving a system of linear equations. The solvability of these linear systems is usually based on the fact that it is a Vandermonde system.

Here is a simple example. Suppose $\mathcal{F}$ contains some function $F = [1, a, 1]$, where $a \neq 0$ and is not a root of unity. Suppose we want to simulate a function $H = [1, b, 1]$, that is, we want to reduce #CSP($\mathcal{F} \cup \{H\}$) to #CSP($\mathcal{F}$). Given an instance I of #CSP($\mathcal{F} \cup \{H\}$), where there are n occurrences of the constraint H, we construct instances I_j of #CSP($\mathcal{F}$), for each $j \geq 1$, by replacing each occurrence of the constraint $H(x_{i_1}, x_{i_2})$ in I by j many constraints $F(x_{i_1}, x_{i_2})$. Each occurrence of H takes an input of the form $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$. We can write the sum defining $Z(I)$ as a sum over all assignments stratified according to the number of $(1, 0)$ or $(0, 1)$ assigned at the n occurrences of H. If exactly i of n occurrences of H are assigned $(1, 0)$ or $(0, 1)$, then the other $n - i$ are assigned $(0, 0)$ or $(1, 1)$. The value $Z(I)$ can be written as the summation

$$Z(I) = \sum_{i=0}^{n} w_i b^i, \tag{3.3}$$

where w_i is the summation of products of all constraints other than these n occurrences of H, over all assignments such that exactly i of these n occurrences of H receive inputs $(1, 0)$ or $(0, 1)$. Note that for assignments $(1, 0)$ or $(0, 1)$, H evaluates to b; the factor b^i in (3.3) comes from this. The other $n - i$ occurrences of H receiving inputs $(0, 0)$ or $(1, 1)$ contribute a factor 1. Meanwhile, we

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

have

$$Z(I_j) = \sum_{i=0}^{n} w_i a^{ij}. \tag{3.4}$$

Note that the same set of values w_i occur here. We let $j = 1, \ldots, n+1$; then (3.4) becomes a system of linear equations about w_i, whose coefficient matrix is a Vandermonde matrix (a^{ij}). The values $Z(I_j)$ can be obtained by oracle calls to #CSP($\mathcal{F}$). Since a is nonzero and not a root of unity, (a^{ij}) is a nonsingular matrix, and we can solve for all w_i in polynomial time from (3.4), which gives us $Z(I)$ by (3.3). This is essentially how every reduction by polynomial interpolation in this chapter will be done.

To get acquainted with the technique, we start with the following theorem, a special case of Theorem 3.14. We illustrate two features. The first is the use of interpolation in a particularly simple setting, where nonnegativity avoids many complications. The second is to show that the ability to get unary functions is a powerful step to prove #P-hardness. Once we can interpolate all unary functions, we can appeal to the dichotomy theorem for Holant* problems. This theorem is also a very special case of much more general theorems on graph homomorphisms, in particular the Bulatov–Grohe theorem (see Volume II).

Theorem 3.15. *Let $[a, b, c]$ be a symmetric binary function, where a, b, c are nonnegative real numbers. Then* #CSP($[a, b, c]$) *is #P-hard unless $[a, b, c]$ is of one of the following three forms: $[a, 0, c]$, $[0, b, 0]$ or $[x^2, xy, y^2]$.*

For #CSP^c($[a, b, c]$) *the same statement holds, and when* #CSP^c($[a, b, c]$) *is #P-hard a reduction exists that does not use Lemma 3.13.*

Proof. We note that #CSP($[0, 1, 1]$) counts the number of vertex covers in a graph. Given a graph $G = (V, E)$, we may consider it defines a #CSP($[0, 1, 1]$) instance I, where each vertex is a Boolean variable and each edge is given the constraint $[0, 1, 1]$ representing the Boolean Or function. Then the product over all edges evaluates to 1 for an assignment $\sigma : V \to \{0, 1\}$ iff $\sigma^{-1}(1)$ is a vertex cover of G. Hence $Z(I)$ is exactly the number vertex covers in G. This problem is #P-complete.

If we used the constraint $[1, 1, 0]$, then #CSP($[1, 1, 0]$) counts the number of independent sets in a graph, since $\sigma^{-1}(1)$ is an independent set iff the product for σ evaluates to 1. In fact, by exchanging 0 and 1, we see that these two problems always have the same solution.

We will reduce one of these #P-hard problems to #CSP($[a, b, c]$), if $[a, b, c]$ does not have one of the three forms. We assume $b \neq 0$, and not both a and c are 0. Since $b \neq 0$, we may normalize $b = 1$ by replacing $[a, b, c]$ with $[a/b, 1, c/b]$. This only causes $Z(I)$ to change by a factor b^m, where m is the

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

number of occurrences of the constraint $[a, b, c]$ in the input instance I. Thus we may assume the given constraint is $[x, 1, y]$, for some nonnegative x and y, where $(x, y) \neq (0, 0)$, and $xy \neq 1$.

First suppose $y \neq 0, 1$. Using Δ_1 we can get the unary function $[1, y]$, namely

$$\#\mathrm{CSP}([x, 1, y], [1, y]) \leq_{\mathrm{T}} \#\mathrm{CSP}([x, 1, y]).$$

With a positive $y \neq 1$, we can interpolate any unary function using $[1, y]$ as follows. $\#\mathrm{CSP}(\mathcal{F})$ is precisely $\mathrm{Holant}(\mathcal{EQ} \mid \mathcal{F})$ where $\mathcal{EQ} = \{(=_k) \mid k \geq 1\}$ is the set of all EQUALITIES, and $\mathcal{F}$ is any set of constraints. Given any unary function $[u, v]$ and any $\mathcal{F}$, we reduce $\#\mathrm{CSP}(\mathcal{F} \cup \{[u, v]\})$ to $\#\mathrm{CSP}(\mathcal{F} \cup \{[1, y]\})$. Suppose I is an instance for $\#\mathrm{CSP}(\mathcal{F} \cup \{[u, v]\})$, where $[u, v]$ occurs n times. Then $Z(I) = \sum_{i=0}^{n} w_i u^{n-i} v^i$, where w_i is the summation of products of all constraints other than the n occurrences of $[u, v]$, over all assignments such that exactly i of these n occurrences of $[u, v]$ receive input 1. Define an instance I_j for $\#\mathrm{CSP}(\mathcal{F} \cup \{[1, y]\})$ by replacing each occurrence of $[u, v]$ in I by j occurrences of $[1, y]$. Then $Z(I_j) = \sum_{i=0}^{n} w_i y^{ij}$, for all $j \geq 1$. Taking $1 \leq j \leq n + 1$, we obtain a nonsingular Vandermonde system, as the positive $y \neq 1$. We can solve for all w_i in polynomial time, and compute $Z(I)$.

Similarly, if $x \neq 0, 1$, then using Δ_0 we can get the unary function $[x, 1]$, and then interpolate any other unary function. If $x, y \in \{0, 1\}$, since $(x, y) \neq (0, 0)$ and $xy \neq 1$, the only cases left are $(x, y) = (0, 1)$ or $(1, 0)$. These are the #P-hard problems VERTEX COVER and INDEPENDENT SET respectively.

Now suppose we can interpolate any unary function $[u, v]$. By a finite sequence of reductions, we can interpolate any finite set of unary functions. By the dichotomy theorem for Holant* problems, namely Theorem 2.12 (essentially Lemma 2.14), $\mathrm{Holant}^*((=_3) \mid [x, 1, y])$ is #P-hard. This means that for some finite subset of unary functions $\mathcal{U}' \subset \mathcal{U}$, $\mathrm{Holant}(\{(=_3)\} \cup \mathcal{U}' \mid \{[x, 1, y]\} \cup \mathcal{U}')$ is #P-hard. Hence $\mathrm{Holant}((=_3) \mid \{[x, 1, y]\} \cup \mathcal{U}'')$ is #P-hard for some finite $\mathcal{U}'' \subset \mathcal{U}$. The latter is reducible to $\#\mathrm{CSP}([x, 1, y])$; hence it is also #P-hard. This argument will be used often, and for simplicity in what follows we will just say

$$\mathrm{Holant}^*((=_3) \mid [x, 1, y]) \leq_{\mathrm{T}} \#\mathrm{CSP}([x, 1, y]),$$

and since $\mathrm{Holant}^*((=_3) \mid [x, 1, y])$ is #P-hard by Theorem 2.12, we conclude that $\#\mathrm{CSP}([x, 1, y])$ is also #P-hard. □

Now we consider complex valued functions. First we prove two simple lemmas.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Lemma 3.16. *For any symmetric binary function* $[0, b, c]$*, where* $b, c \in \mathbb{C}$ *and* $bc \neq 0$*, the problem* #CSP($[0, b, c]$) *is #P-hard. For* #CSP^c($[0, b, c]$) *a reduction exists that does not use Lemma 3.13.*

Proof. Since $b \neq 0$, we can normalize it and assume $b = 1$. So we rename the constraint function $[0, 1, c]$. First suppose c is a root of unity. Let $c^k = 1$. We can realize $[0, 1^k, c^k] = [0, 1, 1]$ by k repeated copies of $[0, 1, c]$. The problem #CSP($[0, 1, 1]$) is counting vertex covers, hence #P-hard. Now suppose c is not a root of unity. Also by assumption $c \neq 0$. We can interpolate $[0, 1, x]$, for any x, by polynomial interpolation. In particular, we can interpolate $[0, 1, 1]$ and #CSP($[0, 1, 1]$) is #P-hard. □

Lemma 3.17. *Let* $[1, b, c]$ *be a symmetric binary function, where* $bc \neq 0$ *and* $c \neq b^2$*. Suppose either* b *or* c *is not a root of unity. Then* #CSP($[1, b, c]$) *is #P-hard. For* #CSP^c($[1, b, c]$) *the same statement holds, and when it is #P-hard a reduction exists that does not use Lemma 3.13.*

Proof. Suppose b is not a root of unity. Using Δ_0 we can get the unary function $[1, b]$. Then we can interpolate any finitely many unary functions. Then

$$\text{Holant}^*((=_3) \mid [1, b, c]) \leq_{\text{T}} \#\text{CSP}([1, b, c]).$$

By Theorem 2.12 for Holant* problems, #CSP($[1, b, c]$) is #P-hard.

Next suppose c is not a root of unity. Connect two inputs of $(=_3)$ by $[1, b, c]$, we can get the function $[1, c]$, and interpolate any unary function by polynomial interpolation, as in Theorem 3.15. Hence again by Theorem 2.12, #CSP($[1, b, c]$) is #P-hard. □

Now we prove Theorem 3.14: *If* $[a, b, c] \notin \mathscr{A} \cup \mathscr{P}$*, then* #CSP($[a, b, c]$) *is #P-hard.* The basic idea is to interpolate unary functions and then appeal to the Holant* dichotomy. However, in order to interpolate all unary functions we must handle some special cases. For this purpose we also use a gadget that links two copies of the binary function $F = [a, b, c]$. It is realizable in #CSP as $G(x, y) = \sum_z F(x, z)F(z, y)$, using an extra variable z not appearing elsewhere.

Proof. of Theorem 3.14. If $a = 0$, we know $bc \neq 0$, otherwise it is in the second or the third exceptional case of Theorem 3.14. So by Lemma 3.16, #CSP($[a, b, c]$) is #P-hard. The case $c = 0$ is symmetric. Since $[a, b, c] \notin \mathscr{A} \cup \mathscr{P}$, we know $b \neq 0$. Therefore we will assume in the following that $abc \neq 0$, and by normalizing, we can assume $a = 1$.

Now we consider the complexity of #CSP($[1, b, c]$), with $bc \neq 0$. We assume $c \neq b^2$; otherwise $[1, b, c]$ is degenerate, the third exceptional case of

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Theorem 3.14. If either b or c is not a root of unity, then #CSP($[1, b, c]$) is #P-hard by Lemma 3.17.

Suppose both b and c are roots of unity. We can realize $G = \begin{bmatrix} 1 & b_2 \\ b & c \end{bmatrix} = [1 + b^2, b + bc, b^2 + c^2]$, which is the signature of two consecutive copies of $[1, b, c]$. G is also nondegenerate. If #CSP(G) is #P-hard so is #CSP($[1, b, c]$). We consider the following cases.

1. $b = -1$. $G = [2, -1 - c, 1 + c^2]$.
   Since $[1, b, c] = [1, -1, c] \notin \mathscr{A} \cup \mathscr{P}$, we know $c \neq \pm 1$. If $c = \pm \mathrm{i}$, we get $G = [2, -1 \mp \mathrm{i}, 0]$, which is #P-hard by Lemma 3.16 (or rather a version of Lemma 3.16 on a reversed signature, by exchanging the inputs 0 and 1). If $c \notin \{\pm 1 \pm \mathrm{i}\}$, then there are no zero entries in G. Since c is a root of unity, and $c \neq \pm 1$, we have $|1 + c^2| \neq 2$. In particular, $\frac{1+c^2}{2}$ is not a root of unity. Normalizing G, we have $[1, \frac{-1-c}{2}, \frac{1+c^2}{2}]$. Hence #CSP($[1, b, c]$) is #P-hard by Lemma 3.17.
2. $b = -c$. $G = [1 + c^2, -c - c^2, 2c^2]$.
   Since $[1, b, c] = [1, -c, c] \notin \mathscr{A} \cup \mathscr{P}$, we know $c \neq \pm 1$. If $c = \pm \mathrm{i}$, we get $G = [0, 1 \mp \mathrm{i}, -2]$, which is #P-hard by Lemma 3.16. If $c \notin \{\pm 1 \pm \mathrm{i}\}$, then there are no zero entries in G. Normalizing, we get $[1, \frac{-c-c^2}{1+c^2}, \frac{2c^2}{1+c^2}]$. For c a root of unity but $c \neq \pm 1$, we have $|1 + c^2| \neq |2c^2|$. In particular, $\frac{2c^2}{1+c^2}$ is not a root of unity. It follows from Lemma 3.17 that #CSP(G) is #P-hard.
3. $c = 1$. $G = [1 + b^2, 2b, 1 + b^2]$.
   Since $[1, b, c] = [1, b, 1] \notin \mathscr{A} \cup \mathscr{P}$, we know $b \notin \{\pm 1 \pm \mathrm{i}\}$. So $1 + b^2 \neq 0$, and $|\frac{2b}{1+b^2}| = \frac{2}{|1+b^2|} \neq 1$. Hence $\frac{2b}{1+b^2}$ is not a root of unity. Therefore the problem is #P-hard by Lemma 3.17.
4. $b \neq -1, b \neq -c$, and $c \neq 1$. Recall that we are given $c \neq b^2$ since $[1, b, c] \notin \mathscr{A} \cup \mathscr{P}$.
   If $F(x_1, x_2) = [1, b, c]$, then $F^{x_2=*} = [1 + b, b + c]$ with both entries nonzero. By $|b| = |c| = 1$, we claim that $|1 + b| = |b + c|$ if and only if $c = 1$ or $c = b^2$. For now assume the claim holds. Since we assumed $c \neq 1$ and $c \neq b^2$, we have $\frac{|b+c|}{|1+b|} \neq 1$ and thus we can interpolate all unary functions with $[1 + b, b + c]$. Therefore #CSP($[1, b, c]$) is #P-hard by Theorem 2.12 for Holant* problems.
   To prove the claim one direction is obvious. For the reverse direction, suppose $|1 + b| = |b + c|$; then $(1 + b)(1 + \bar{b}) = (b + c)(\bar{b} + \bar{c})$. Expanding this equation, we get $b + \bar{b} = c\bar{b} + \bar{c}b$, i.e., Re b = Re c/b. If $b = e^{i\alpha}$ and $c = e^{i\beta}$, then $\cos\alpha = \cos(\beta - \alpha)$, hence $\beta - \alpha = \pm\alpha \bmod 2\pi$. It follows that $c = 1$ or b^2. □

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Theorem 3.14 is the special case of graph homomorphism on the Boolean domain. We will discuss graph homomorphism on a general domain in much more detail in Volume II of the book.

3.5.2 Affine Support

The following lemma gives a necessary condition for #CSP(F) being *not* #P-hard. This affine support criterion will be much generalized later when we discuss dichotomies for graph homomorphisms and #CSP on general, i.e., not necessarily Boolean, domains. The proof given here is elementary.

Lemma 3.18. *If the relation R_F defined by F is not affine, then #CSP(F) is #P-hard. For #CSP^c(F) the same statement holds, and when R_F is not affine a reduction exists that does not use Lemma 3.13.*

Proof. We prove by induction on the arity of the function F.

All functions of arity one have affine support. So we assume the arity of F is at least two. We first consider a function F of arity two. Suppose F does not have affine support. This implies that exactly one of its four values is 0. The matrix form of F is

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \begin{bmatrix} F(0,0) & F(0,1) \\ F(1,0) & F(1,1) \end{bmatrix};$$

then in particular $\det(F) \neq 0$. By taking two copies of F sharing a free variable z in the appropriate order (x, z) and (z, y), we can realize the binary function $H(x, y) = \sum_z F(x, z)F(z, y)$, whose matrix form is $H = FF^{\mathrm{T}} = \begin{bmatrix} a^2+b^2 & ac+bd \\ ac+bd & c^2+d^2 \end{bmatrix}$. This H is a symmetric binary function, which can also be denoted by $[a^2 + b^2, ac + bd, c^2 + d^2]$. We can apply Theorem 3.14 to H. Because F is nonsingular, so is the matrix for H. Because exactly one entry of F is 0, $ac + bd \neq 0$ and H is not of the form $[x, 0, y]$. Because either $a^2 + b^2 \neq 0$ or $c^2 + d^2 \neq 0$, H is not of the form $[0, x, 0]$. So the only remaining possibilities for $H \in \mathscr{A} \cup \mathscr{P}$ is that H is of the form $x[1, \pm\mathrm{i}, 1]$ or $x[1, \pm 1, -1]$. By symmetry, we only need to consider the cases $a = 0$ and $bcd \neq 0$, or $b = 0$ and $acd \neq 0$. If $a = 0$, we can assume $b = 1$ by normalization, and then the function H is $[1, d, c^2 + d^2]$. If H is of the form $x[1, \pm\mathrm{i}, 1]$, we have $d = \pm\mathrm{i}$ and $c = \pm\sqrt{2}$. Then we can realize another symmetric binary function by $G(x, y) = F(x, y)F(y, x)$. So $G = [a^2, bc, d^2] = [0, \pm\sqrt{2}, -1]$. #CSP($G$) is #P-hard by Theorem 3.14. If H is of the form $x[1, \pm 1, -1]$, we have $d = \pm 1$ and $c = \pm\sqrt{2}\mathrm{i}$. Then $G = [a^2, bc, d^2] = [0, \pm\sqrt{2}\mathrm{i}, 1]$ and #CSP(G) is #P-hard by Theorem 3.14 again. So we have completed for the $a = 0$ case. If

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

$b = 0$, we can assume $a = 1$ and the function H is $[1, c, c^2 + d^2]$. If H is of the form $x[1, \pm\mathrm{i}, 1]$, we have $c = \pm\mathrm{i}$ and $d = \pm\sqrt{2}$. Then we can realize another binary function F' by $F'(x, y) = F(x, y)F(x, y)$. In matrix notation $F' = \begin{bmatrix} a^2 & b^2 \\ c^2 & d^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 2 \end{bmatrix}$. Next we can simulate $H' = F'F'^{\mathrm{T}} = \begin{bmatrix} 1 & -1 \\ -1 & 5 \end{bmatrix}$ from F'. In symmetric notation $H' = [1, -1, 5]$. By Theorem 3.14 #CSP(H') is #P-hard. Finally, if H is of the form $x[1, \pm 1, -1]$, we have $c = \pm 1$ and $d = \pm\sqrt{2}\mathrm{i}$. Then by the same construction, $F' = \begin{bmatrix} 1 & 0 \\ 1 & -2 \end{bmatrix}$ and $H' = \begin{bmatrix} 1 & 1 \\ 1 & 5 \end{bmatrix}$, which in symmetric notation is $H' = [1, 1, 5]$. So again #CSP(H') is #P-hard. We have completed the proof for the case where the function F is of arity 2.

Inductively we assume the lemma has been proved for functions with arity less than k, for some $k \geq 3$, and now assume the function F has arity k. Since R_F is not affine, there exist $\alpha, \beta, \gamma \in R_F$, where $\alpha = (a_1, \ldots, a_k)$, $\beta = (b_1, \ldots, b_k)$, $\gamma = (c_1, \ldots, c_k)$, such that $\delta = \alpha \oplus \beta \oplus \gamma = (d_1, \ldots, d_k) \notin R_F$. We only need to prove that we can use F to simulate a function of smaller arity that does not have affine support.

Divide the index set $[k]$ of input variables of F into four subsets according to the values of α, β, γ as follows:

$$I = \{j \mid a_j = b_j \neq c_j\}, \quad J = \{j \mid a_j = c_j \neq b_j\}, \quad K = \{j \mid b_j = c_j \neq a_j\},$$
$$L = \{j \mid a_j = b_j = c_j\}.$$

Since a_j, b_j and c_j take Boolean values, $\{I, J, K, L\}$ is a partition of $[k]$. We also remark that, if $j, l \in I$, then either $(a_l, b_l, c_l) = (a_j, b_j, c_j)$ or $(\overline{a_j}, \overline{b_j}, \overline{c_j})$. The same is true for J, K and L. Now we have the following four cases, and for each case, we prove our result.

- L is not empty. There exists j such that $a_j = b_j = c_j$.

  We fix the jth input of F to be a_j, and get a function $F^{x_j = a_j}$. Recall that $\delta = \alpha \oplus \beta \oplus \gamma$; then $d_j = a_j = b_j = c_j$. This function $F^{x_j = a_j}$ does not have affine support.

  Now we may assume $L = \emptyset$ and $[k] = I \cup J \cup K$.
- There are indices $l \neq j$, such that $(a_l, b_l, c_l) = (a_j, b_j, c_j)$.

  Without loss of generality we assume $l = 1$ and $j = 2$. Define a function of arity $k - 1$ by $H(x_1, x_3 \ldots, x_k) = F(x_1, x_1, x_3 \ldots, x_k)$. H can be simulated by F, and by the property that $\alpha, \beta, \gamma \in R_F$ and yet $\delta \notin R_F$, H does not have affine support.
- There are indices $l \neq j$, such that $(a_l, b_l, c_l) = (\overline{a_j}, \overline{b_j}, \overline{c_j})$.

  Clearly both l and j belong to the same set I or J or K. Without loss of generality we assume $l = 1 \in I$ and $j = 2 \in I$. The proofs for J and K are

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

the same. Then we have

$$\alpha = (a, \overline{a}, \alpha'), \quad \beta = (a, \overline{a}, \beta'), \quad \gamma = (\overline{a}, a, \gamma'), \quad \text{and} \quad \delta = (\overline{a}, a, \delta'), \tag{3.5}$$

where $a \in \mathbb{Z}_2$, $\alpha', \beta', \gamma' \in \mathbb{Z}_2^{k-2}$, and $\delta' = \alpha' \oplus \beta' \oplus \gamma' \in \mathbb{Z}_2^{k-2}$. Assume for a contradiction that all functions of the forms $F^{x_i=b}$ and $F^{x_i=*}$ have affine support ($1 \le i \le k$, $b = 0, 1$).

Consider $F^{x_1=a}$, whose underlying relation $R_{F^{x_1=a}}$ is affine. Because $\alpha, \beta \in R_F$, $(\overline{a}, \alpha'), (\overline{a}, \beta') \in R_{F^{x_1=a}}$. The summation of $(\overline{a}, \alpha')$, $(\overline{a}, \beta')$, (a, γ'), (a, δ') is the zero vector in $\mathbb{Z}_2^{k-1}$, so $(a, \gamma') \in R_{F^{x_1=a}}$ iff $(a, \delta') \in R_{F^{x_1=a}}$. This implies that $(a, a, \gamma') \in R_F$ iff $(a, a, \delta') \in R_F$.

Next consider $F^{x_2=a}$. Because $\gamma \in R_F$ and $\delta \notin R_F$, we have $(\overline{a}, \gamma') \in R_{F^{x_2=a}}$, and $(\overline{a}, \delta') \notin R_{F^{x_2=a}}$. As we just proved, $R_F(a, a, \gamma') = R_F(a, a, \delta')$. If this value is 1, then $(a, \gamma'), (a, \delta') \in R_{F^{x_2=a}}$, but this is impossible for the affine relation $R_{F^{x_2=a}}$, in view of the four inputs $(\overline{a}, \gamma')$, $(\overline{a}, \delta')$, (a, γ'), and (a, δ'). So we must have $(a, a, \gamma') \notin R_F$ and $(a, a, \delta') \notin R_F$.

Similarly, we can prove that both $(\overline{a}, \overline{a}, \alpha')$ and $(\overline{a}, \overline{a}, \beta') \notin R_F$. More explicitly, first consider $F^{x_2=\overline{a}}$. Because $\alpha, \beta \in R_F$, $(a, \alpha'), (a, \beta') \in R_{F^{x_2=\overline{a}}}$. Having an affine support, $R_{F^{x_2=\overline{a}}}$ evaluates to the same value on $(\overline{a}, \alpha')$ and $(\overline{a}, \beta')$. Thus R_F evaluates to the same value on $(\overline{a}, \overline{a}, \alpha')$ and $(\overline{a}, \overline{a}, \beta')$. Next consider $F^{x_1=\overline{a}}$. It also has an affine support. Since $\gamma \in R_F$ and $\delta \notin R_F$, we have $(a, \gamma') \in R_{F^{x_1=\overline{a}}}$ and $(a, \delta') \notin R_{F^{x_1=\overline{a}}}$. If $(\overline{a}, \overline{a}, \alpha'), (\overline{a}, \overline{a}, \beta') \in R_F$, then $(\overline{a}, \alpha'), (\overline{a}, \beta') \in R_{F^{x_1=\overline{a}}}$. This is impossible for an affine relation $R_{F^{x_1=\overline{a}}}$, in view of the four inputs (a, γ'), (a, δ'), $(\overline{a}, \alpha')$, and $(\overline{a}, \beta')$. Thus it follows that $(\overline{a}, \overline{a}, \alpha') \notin R_F$ and $(\overline{a}, \overline{a}, \beta') \notin R_F$.

To summarize, we have all (a, a, γ'), (a, a, δ'), $(\overline{a}, \overline{a}, \alpha')$, $(\overline{a}, \overline{a}, \beta') \notin R_F$.

Finally, we consider $F^{x_1=*}$, and calculate as follows:

$$\begin{aligned}
F^{x_1=*}(\overline{a}, \alpha') &= F(\alpha) + F(\overline{a}, \overline{a}, \alpha') = F(\alpha) \neq 0, \\
F^{x_1=*}(\overline{a}, \beta') &= F(\beta) + F(\overline{a}, \overline{a}, \beta') = F(\beta) \neq 0, \\
F^{x_1=*}(a, \gamma') &= F(\gamma) + F(a, a, \gamma') = F(\gamma) \neq 0, \\
F^{x_1=*}(a, \delta') &= F(\delta) + F(a, a, \delta') = F(\delta) = 0.
\end{aligned}$$

This is a contradiction, since we made the assumption that $R_{F^{x_1=*}}$ is affine.

- If there is more than one element in the set I, or in the set J, or in the set K, it is included in the previous two cases. The remaining case is that the sizes of I, J, K are all at most one and L is empty. Because $k > 2$, the sizes of I, J, K are exactly one, and $k = 3$. Without loss of generality, let $I = \{1\}$, $J = \{2\}$ and $K = \{3\}$.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

A moment's reflection shows that we can write α, β, γ and δ as follows:

$$\alpha = (p, q, \overline{r}),$$
$$\beta = (p, \overline{q}, r),$$
$$\gamma = (\overline{p}, q, r),$$
$$\delta = (\overline{p}, \overline{q}, \overline{r}),$$

where $p, q, r \in \mathbb{Z}_2$.

First we consider $F^{x_1=p}$, which has an affine support by our induction hypothesis on arity. Let $u = (p, q, r)$, and suppose $u \in R_F$. Then $(q, r) \in R_{F^{x_1=p}}$. Because $\alpha, \beta \in R_F$, then $(q, \overline{r})$ and $(\overline{q}, r)$ both belong to $R_{F^{x_1=p}}$. Then being affine, $(\overline{q}, \overline{r}) \in R_{F^{x_1=p}}$. Let $v = (p, \overline{q}, \overline{r})$, then $v \in R_F$.

Next we consider $R_{F^{x_2=q}}$. By $\alpha, \gamma \in R_F$, we get $(p, \overline{r}), (\overline{p}, r) \in R_{F^{x_2=q}}$. By assumption $u \in R_F$, then $(p, r) \in R_{F^{x_2=q}}$. By $R_{F^{x_2=q}}$ being affine, we get $(\overline{p}, \overline{r}) \in R_{F^{x_2=q}}$. Let $w = (\overline{p}, q, \overline{r})$; then $w \in R_F$.

Now $\alpha, v, w \in R_F$. This gives us $(p, q), (p, \overline{q}), (\overline{p}, q) \in R_{F^{x_3=\overline{r}}}$. Since $R_{F^{x_3=\overline{r}}}$ is affine, $(\overline{p}, \overline{q}) \in R_{F^{x_3=\overline{r}}}$. This means that $\delta = (\overline{p}, \overline{q}, \overline{r}) \in R_F$, which is a contradiction.

We conclude that in fact $u \notin R_F$.

By tracing the above steps, under the new condition $u \notin R_F$, we get $v \notin R_F$, and also $w \notin R_F$.

Finally we consider $F^{x_3=r}$. By $\beta, \gamma \in R_F$, we get $(p, \overline{q}), (\overline{p}, q) \in R_{F^{x_3=r}}$. By $u \notin R_F$, we have $(p, q) \notin R_{F^{x_3=r}}$. By $R_{F^{x_3=r}}$ being affine, we get $(\overline{p}, \overline{q}) \notin R_{F^{x_3=r}}$. i.e., $(\overline{p}, \overline{q}, r) \notin R_F$.

We have now accounted for all eight points of $\mathbb{Z}_2^3$. Exactly three of them α, β, γ belong to R_F and the other five do not. It can be directly verified that $R_{F^{x_1=*}}$ has exactly three members $(q, r), (q, \overline{r}), (\overline{q}, r)$, but not $(\overline{q}, \overline{r})$, which is a contradiction to $R_{F^{x_1=*}}$ being affine. This contradiction completes our proof. □

3.5.3 Arity Reduction

Now we come to the two key lemmas for the hardness proof. Both proofs inductively reduce the arity of a function. Suppose $\mathcal{F} \not\subseteq \mathscr{A}$ and $\mathcal{F} \not\subseteq \mathscr{P}$. Let $F \notin \mathscr{A}$ and $G \notin \mathscr{P}$, where $F, G \in \mathcal{F}$. (It is possible that $F = G$). From F and from G, we recursively simulate functions with smaller arities, keeping the property of being not in $\mathscr{A}$ and not in $\mathscr{P}$ respectively. After the two lemmas we handle the base case of the induction.

Lemma 3.19. *If $F \notin \mathscr{A}$, then either #CSP(F) is #P-hard, or we can simulate a unary function $H \notin \mathscr{A}$, that is, there is a reduction from #CSP($\{F, H\}$) to #CSP(F).*

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

The same statement holds for #CSP^c, and a reduction exists that does not use Lemma 3.13.

Proof. We prove by induction on the arity of the function F. If F has arity 1, then we may let $H = F$.

Inductively we assume the lemma has been proved for functions with arity less than k, for some $k \geq 2$. Now let F have arity k. In the following proof, for each case, we construct some functions that can be simulated in #CSP(F), but have arity less than k, and then *assume* they are in $\mathscr{A}$ (otherwise, it is proved by induction). Finally we prove that either the problem is #P-hard, or we get a unary function $H \notin \mathscr{A}$ or reach a contradiction.

Since the constant function 0 is in $\mathscr{A}$, F has a nonempty support R_F. We first assume R_F is not the whole space $\mathbb{Z}_2^k$. By Lemma 3.18, either #CSP(F) is #P-hard, or R_F is affine. Suppose $R_F = \chi_{AX}$, and $x_1, x_2, \ldots, x_s$ ($0 \leq s < k$) are free variables of $AX = 0$. The function $F^{x_{s+1}=*,x_{s+2}=*,\ldots,x_k=*}$ can be simulated by F and has an arity less than k. Thus by our assumption, $F^{x_{s+1}=*,x_{s+2}=*,\ldots,x_k=*} \in \mathscr{A}$. Then obviously F is equal to $\chi_{AX} F^{x_{s+1}=*,x_{s+2}=*,\ldots,x_k=*}$ and thus $F \in \mathscr{A}$. This is a contradiction to the assumption that $F \notin \mathscr{A}$.

So we may assume $R_F = \mathbb{Z}_2^k$. We normalize F by dividing F by the nonzero constant $F(0, 0, \ldots, 0)$ so that we may assume $F(0, 0, \ldots, 0) = 1$. By our assumption, $F^{x_1=0} \in \mathscr{A}$ so all values of $F(0, x_2, \ldots, x_k)$ are powers of i. Also, $F^{x_2=0}, F^{x_2=1} \in \mathscr{A}$, and since all values of $F^{x_1=0,x_2=0}$ and $F^{x_1=0,x_2=1}$ are powers of i, so are all values of $F^{x_2=0}$ and $F^{x_2=1}$. In particular, so are all values of $F^{x_1=1,x_2=1}$.

Now we apply Lemma 3.10 to $F^{x_2=0} \in \mathscr{A}$ and $F^{x_2=1} \in \mathscr{A}$. Accordingly, we have the following three cases.

1. Both $F^{x_2=0}$ and $F^{x_2=1}$ have congruity. We will denote the function $F^{x_1=a,x_2=b}$ by F^{ab}. Let c and $c' \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$ be the two constants in the definition of congruity for $F^{x_2=0}$ and $F^{x_2=1}$ respectively, in Lemma 3.10. Thus $F^{10}/F^{00}(x_3, \ldots, x_k) = c$ and $F^{11}/F^{01}(x_3, \ldots, x_k) = c'$.
   (a) $c = c'$.
   This means $F^{x_1=1}/F^{x_1=0}$ is a constant c in $\{1, -1, \mathrm{i}, -\mathrm{i}\}$. Suppose $c = \mathrm{i}^r$. Then $F = (\mathrm{i}^{x_1})^r F^{x_1=0} = \mathrm{i}^{rx_1^2} F^{x_1=0}$. Since $F^{x_1=0}$ is in $\mathscr{A}$ by arity, this shows that F is also in $\mathscr{A}$, a contradiction.
   (b) $c = -c'$.
   We will use the notation $[\alpha(X)]$ to denote the 0-1 indicator function for an affine linear function $\alpha(X)$ over $\mathbb{Z}_2$. For any input X, $[\alpha(X)]$ takes value $0 \in \mathbb{Z}$ if $\alpha(X) = 0$ in $\mathbb{Z}_2$, and it takes value $1 \in \mathbb{Z}$ if $\alpha(X) = 1$ in $\mathbb{Z}_2$.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Since $c \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$, there exists an r such that $\mathrm{i}^r = c/\mathrm{i}$. Then we claim that

$$F = \left(\mathrm{i}^{[x_1]}\right)^r \cdot \mathrm{i}^{[x_1 \oplus x_2]+[x_2]+[x_2]+[x_2]} \cdot F^{x_1=0}.$$

To verify this, first suppose $x_1 = 0$, then the right-hand side is $\mathrm{i}^{4[x_2]} \cdot F^{x_1=0} = F^{x_1=0}$. Now let $x_1 = 1$; then the right-hand side is $\mathrm{i}^r \cdot \mathrm{i}^{1-[x_2]+3[x_2]} \cdot F^{x_1=0} = c(-1)^{[x_2]} F^{x_1=0}$. This is $cF^{00} = F^{10}$, if $x_2 = 0$. For $x_2 = 1$, the expression is $-cF^{01} = c'F^{01} = F^{11}$. The claim is proved. Now since $F^{x_1=0}$ has arity less than k, $F^{x_1=0} \in \mathscr{A}$. But then the claim implies that $F \in \mathscr{A}$ as well, a contradiction.

(c) $c = \mathrm{i}c'$ or $c = -\mathrm{i}c'$.

Assign an arbitrary assignment for $x_3, \ldots, x_k$. Let P be the resulting function on x_1, x_2. In matrix form, where the rows are indexed by $x_1 = 0, 1$ and columns are indexed by $x_2 = 0, 1$, we have $P = \begin{bmatrix} u & v \\ \pm\mathrm{i}c'u & c'v \end{bmatrix}$, for some u, v taking values in $\{\pm 1, \pm\mathrm{i}\}$. Let $Q(x_1, x_2) = P^3(0, x_2)P(x_1, x_2)$, a product of functions, realizable by pinning Δ_0. In matrix form, this is $Q = \begin{bmatrix} u^4 & v^4 \\ \pm\mathrm{i}c'u^4 & c'v^4 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ \pm\mathrm{i}c' & c' \end{bmatrix}$. Here we used the fact that the values of P are powers of i. Now $Q^{x_2=*}$ is a unary function $[2, (1 \pm \mathrm{i})c']$ that has unequal and nonzero norms $2 \neq |(1 \pm \mathrm{i})c'| = \sqrt{2}$ and hence not in $\mathscr{A}$.

2. One of $F^{x_2=0}$ and $F^{x_2=1}$ has congruity and the other has semicongruity. Without loss of generality, assume $F^{x_2=0}$ has congruity and $F^{x_2=1}$ has semicongruity. The other case is similar.

By congruity, there is a constant $c \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$, such that $F^{10}/F^{00}(x_3, \ldots, x_k) = c$ for all $x_3, \ldots, x_k \in \mathbb{Z}_2^{k-2}$. By semicongruity there is a constant $c' \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$, and a $(k-3)$-dimensional affine linear subspace $S \subset \mathbb{Z}_2^{k-2}$, represented by an affine linear function $\alpha(x_3, \ldots, x_k) = 0$, such that on S, $F^{11}/F^{01}(x_3, \ldots, x_k) = c'$ and on $\mathbb{Z}_2^{k-2} - S$, $F^{11}/F^{01}(x_3, \ldots, x_k) = -c'$. We note that to have semicongruity, it must be the case that $k \geq 3$, and one of the coefficients of $x_3, \ldots, x_k$ in $\alpha(x_3, \ldots, x_k)$ must be nonzero. Without loss of generality, let it be the coefficient of x_3.

Fix an arbitrary assignment to $x_4, \ldots, x_k$ (if $k = 3$ this step is vacuous), this gives a function $P(x_1, x_2, x_3)$. By changing the constant term in α and c' to $-c'$ if necessary we may assume $x_3 = 0$ gives a point with $\alpha(x_3, \ldots, x_k) = 0$.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Now we will use a special notation to represent $P(x_1, x_2, x_3)$.

$$P = \begin{array}{cccc} z & & & cz \\ & x & cx & \\ & y & c'y & \\ w & & & -c'w \end{array}.$$

This symbol is intended to suggest a cube and is to be read as follows: The left (right) four entries are function values with $x_1 = 0$ ($x_1 = 1$); the top (bottom) four entries are function values with $x_2 = 0$ ($x_2 = 1$); finally the inner (outer) four entries are values with $x_3 = 0$ ($x_3 = 1$).

Let $Q(x_1, x_2, x_3) = P(x_1, x_2, x_3)(P(0, x_2, x_3))^3$. This corresponds to taking the third power of each of left four nodes (x, y, z, w) and multiplying to itself and the node to its right. We get

$$Q = \begin{array}{cccc} 1 & & & c \\ & 1 & c & \\ & 1 & c' & \\ 1 & & & -c' \end{array},$$

since $x^4 = y^4 = z^4 = w^4 = 1$. Next let $R(x_1, x_2, x_3) = Q(x_1, x_2, x_3)(Q(x_1, x_2, 0))^3$. This gives

$$R = \begin{array}{cccc} 1 & & & 1 \\ & 1 & 1 & \\ & 1 & 1 & \\ 1 & & & -1 \end{array},$$

since $c^4 = c'^4 = 1$. Then $R^{x_1=0} = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$, $R^{x_1=1} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$. It follows that $R^{x_1=*} = \begin{bmatrix} 2 & 2 \\ 2 & 0 \end{bmatrix}$, and $R^{x_1=*,x_2=*} = [4, 2]$. It has unequal nonzero norms, hence this unary function $R^{x_1=*,x_2=*} \notin \mathscr{A}$.

3. Both $F^{x_2=0}$ and $F^{x_2=1}$ have semicongruity. Let $F^{10}/F^{00} = c$ on $\alpha(x_3, \ldots, x_k) = 0$ and $-c$ on $\alpha(x_3, \ldots, x_k) = 1$. Similarly $F^{11}/F^{01} = c'$ on $\beta(x_3, \ldots, x_k) = 0$ and $-c'$ on $\beta(x_3, \ldots, x_k) = 1$. Here $c, c' \in \{1, -1, \mathrm{i}, -\mathrm{i}\}$, and α, β are two nontrivial affine linear functions.
   (a) $c \neq \pm c'$. Since β is nontrivial, we may assume the coefficient of x_3 in β is nonzero. Fix any assignment to $x_4, \ldots, x_k$, we may assume without loss of generality $x_3 = 0$ satisfies $\beta = 0$. We have the following function

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

$P(x_1, x_2, x_3)$, which in our symbolic notation is

$$P = \begin{array}{cccc} & z & & \epsilon cz \\ & x & \delta cx & \\ & y & c'y & \\ w & & & -c'w \end{array},$$

where $\epsilon, \delta \in \{\pm 1\}$ depending on the assignment of $x_4, \ldots, x_k$. If $\epsilon = \delta$, the two entries $\pm cz$ and $\pm cx$ both take the same $+c$ or $-c$ multiplier, then we have obtained a ternary function of the same form in case 2 and therefore we can continue in exactly the same way. So assume $\epsilon = -\delta$. By renaming c as $-c$, we may assume the two entries are in fact $-cz$ and $+cx$ respectively. Now we take $Q(x_1, x_2, x_3) = P(0, x_2, x_3)^3 P(x_1, x_2, x_3)$. Finally, let $R(x_1, x_2, x_3) = Q(x_1, 0, x_3)^3 Q(x_1, x_2, x_3)$. Then we have

$$Q = \begin{array}{ccc} 1 & & -c \\ 1 & c & \\ 1 & c' & \\ 1 & & -c' \end{array} \quad \text{and} \quad R = \begin{array}{ccc} 1 & & 1 \\ 1 & 1 & \\ 1 & c'/c & \\ 1 & & c'/c \end{array}.$$

Note that $c^3 c' = c'/c$, since $c^4 = 1$. It is easy to see that $R^{x_1=*, x_3=0} = [2, 1 + c'/c]$. Since $c'/c \neq \pm 1$ we have $c'/c = \pm \mathrm{i}$. Then this unary function $[2, 1 \pm \mathrm{i}] \notin \mathscr{A}$ since it has unequal nonzero norms $2 \neq |1 \pm \mathrm{i}|$.

(b) $c = \pm c' \in \{1, -1\}$. In this case $F^{x_1=1}/F^{x_1=0}$ takes only values ± 1. Then $R_{F^{x_1=*}}$ is precisely where $F^{x_1=1}/F^{x_1=0} = +1$. If $R_{F^{x_1=*}}$ is not affine, we have #P-hardness by Lemma 3.18. So let $R_{F^{x_1=*}}$ be defined by an affine linear $\gamma(x_2, \ldots, x_k) = 0$. It can be directly verified that

$$F = F^{x_1=0} \cdot \mathrm{i}^{[x_1]+[x_1]+[x_1]+[x_1 \oplus \gamma]+[\gamma]+[\gamma]+[\gamma]}.$$

We have $F^{x_1=0} \in \mathscr{A}$ by arity, which implies that $F \in \mathscr{A}$, a contradiction.

(c) $c = \pm c' \in \{\mathrm{i}, -\mathrm{i}\}$. In this case $F^{x_1=1}/F^{x_1=0}$ takes only values $\pm \mathrm{i}$. We may assume $c = c' = \mathrm{i}$ by changing α to $\alpha \oplus 1$ and/or β to $\beta \oplus 1$ if necessary. Consider the subset

$$\begin{aligned} S &= \{(x_2, x_3, \ldots, x_k) \mid F^{x_1=1}/F^{x_1=0} = \mathrm{i}\} \\ &= \{(0, x_3, \ldots, x_k) \mid \alpha(x_3, \ldots, x_k) = 0\} \cup \{(1, x_3, \ldots, x_k) \mid \\ &\qquad \beta(x_3, \ldots, x_k) = 0\}. \end{aligned}$$

First suppose all coefficients of $x_3, \ldots, x_k$ in α and β are the same. If $\alpha(x_3, \ldots, x_k) = \sum_{i=3}^k \alpha_i x_i + a$ and $\beta(x_3, \ldots, x_k) = \sum_{i=3}^k \alpha_i x_i + b$ over $\mathbb{Z}_2$, then $(a \oplus b)x_2 + \sum_{i=3}^k \alpha_i x_i + a = 0$ over $\mathbb{Z}_2$ defines the set S.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Denote this affine linear function by γ, then it can be verified that

$$F = F^{x_1=0} \cdot \mathrm{i}^{[x_1 \oplus \gamma]+[\gamma]+[\gamma]+[\gamma]}.$$

We have $F^{x_1=0} \in \mathscr{A}$ by arity, which implies that $F \in \mathscr{A}$, a contradiction.

Now suppose some coefficients of $x_3, \ldots, x_k$ in α and β differ. Without loss of generality, suppose the coefficient of x_3 is 0 in α, and is 1 in β respectively. Fix any assignment to $x_4, \ldots, x_k$; then the value of α is fixed, and yet by setting x_3 to 0 or 1, the value of β flips. Then we get a function

$$P(x_1, x_2, x_3) = \begin{array}{ccc} z & & \epsilon z \\ x & \epsilon x & \\ y & \delta y & \\ w & & -\delta w \end{array}$$

for some $\epsilon, \delta = \pm\mathrm{i}$. From here the proof is completed as in case 2. □

We observe that for any $a, b \neq 0$, the ternary functions $H = [a, 0, b, 0]$ and $[0, a, 0, b] \notin \mathscr{P}$. In fact, if H were to be in $\mathscr{P}$, any expression as a member of $\mathscr{P}$ must not use any binary DISEQUALITY ($\neq_2$), since there is an assignment having a nonzero value $H \neq 0$, with $x_i = x_j = 0$, for any $i \neq j \in \{1, 2, 3\}$, and the third variable set accordingly. Similarly it cannot use any binary EQUALITY ($=_2$) since there is also an assignment having a nonzero value $H \neq 0$, with $x_i \neq x_j$. But unary functions alone cannot define these functions either.

Lemma 3.20. *For any function $F \notin \mathscr{P}$, either #CSP(F) is #P-hard, or we can simulate, using F, a function $[a, 0, 1, 0]$ (or $[0, 1, 0, a]$), where $a \neq 0$, or a binary function $H \notin \mathscr{P}$ having no zero values.*

The same statement holds for #CSP^c, and a reduction exists that does not use Lemma 3.13.

Proof. Suppose F has arity k. Then $k \geq 2$ since $F \notin \mathscr{P}$ and $\mathscr{P}$ contains all unary functions. Define a $\{0, 1\}$-matrix by listing all elements of R_F in some order. The rows are indexed by R_F and the columns are indexed by $1, \ldots, k$ corresponding to $x_1, \ldots, x_k$.

We first remove any column that is all-0 or all-1. If we remove an all-0 column corresponding to x_i, then any $X \in R_F$ has $x_i = 0$. The updated matrix corresponds to $R_{F^{x_i=0}}$. Similarly, if we remove an all-1 column corresponding to x_i, then any $X \in R_F$ has $x_i = 1$. The updated matrix corresponds to $R_{F^{x_i=1}}$. If two columns are identical or are complementary in every bit, we remove one of them. If the columns at x_i and x_j are identical, then any $X \in R_F$ has $x_i = x_j$. Similarly, for a pair of complementary columns at x_i and x_j, any $X \in R_F$

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

implies that $x_i \oplus x_j = 1$. In both cases, the updated matrix by removing column j corresponds to $R_{F^{x_j=*}}$.

We remove columns as long as possible. We claim that this removal process maintains the property of not belonging to $\mathscr{P}$. Suppose we removed an all-0 column at x_i, to get $G = F^{x_i=0}$. Since any $X \in R_F$ has $x_i = 0$, we have $F = \Delta_0(x_i) \cdot G$, where $\Delta_0(x_i)$ is the unary function $[1, 0]$ applied to x_i. This expression shows that if $G \in \mathscr{P}$ then $F \in \mathscr{P}$, a contradiction. The case with removing an all-1 column is similar, where we use the unary function $\Delta_1(x_i)$ instead. If we removed the column at x_j identical to the column at x_i, then $G = F^{x_j=*}$ and $F = \chi_{x_i=x_j} \cdot G$. Finally, for the removal of a complementary column at x_j we have $G = F^{x_j=*}$ and $F = \chi_{x_i \oplus x_j \oplus 1=0} \cdot G$. In every step, we maintain $G \notin \mathscr{P}$.

Now we suppose there is some $G \notin \mathscr{P}$ where no more columns can be removed by the above process. There must be some columns left in the matrix; otherwise the function just before the last column removal is a unary function, hence in $\mathscr{P}$. In fact since $G \notin \mathscr{P}$, the arity of G is ≥ 2. For simplicity we still denote it by k. Thus $k \geq 2$. We have two cases:

Case 1: $|R_G| < 2^k$. By Lemma 3.18, we may assume R_G is affine, given by a linear system on $x_1, \ldots, x_k$ over $\mathbb{Z}_2$. We have that $|R_G| = |R_F| \neq 0$. This is because we never deleted any rows, and distinct rows produce distinct rows in the column removal process; so the number of rows remains the same. Clearly $|R_G| > 1$ since otherwise every column (of length 1) is all-0 or all-1. Without loss of generality, assume $x_1, \ldots, x_s$ are free variables in the linear system defining R_G, and $x_{s+1}, \ldots, x_k$ are dependent variables. $|R_G| = 2^s$ is a power of 2. We have shown that $s \geq 1$. By $|R_G| < 2^k$, we have $s < k$. We claim $s \geq 2$. If instead $s = 1$, then every $x_2, \ldots, x_k$ is dependent on x_1 on R_G, so the column at x_2 must be an all-0 or all-1 column, or be identical or complementary to x_1, and the removal process would have continued. The expression of x_k in terms of $x_1, \ldots, x_s$ must involve at least two nonzero coefficients; otherwise the column at x_k must be an all-0 or all-1 column, or be identical or complementary to another column. Without loss of generality, say the coefficients of x_1, x_2 are nonzero.

Let $P(x_1, x_2, x_k) = G^{x_3=0,\ldots,x_s=0,x_{s+1}=*,\ldots,x_{k-1}=*}$ (the two sets of variables $\{x_3, \ldots, x_s\}$ and $\{x_{s+1}, \ldots, x_{k-1}\}$ could be empty). Note that the sum over $x_{s+1}, \ldots, x_{k-1}$ does not introduce any cancelations because for any assignment $x_1, \ldots, x_s$ there is a unique nonzero value for each variable $x_{s+1}, \ldots, x_k$. It follows that $R_P = \chi_{x_1 \oplus x_2 \oplus x_k = c}$ for some $c \in \mathbb{Z}_2$.

The affine linear equation $x_1 \oplus x_2 \oplus x_k = c$ is symmetric. Now we define a *symmetrized* function $H(x_1, x_2, x_k) = \prod_{\sigma \in S_3} P(x_{\sigma(1)}, x_{\sigma(2)}, x_{\sigma(k)})$, where S_3 is

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

the symmetry group on three letters $\{1, 2, k\}$. This H is a symmetric function on (x_1, x_2, x_k) and has support $R_H = R_P$. Thus, after normalizing, $H = [a, 0, 1, 0]$ or $[0, 1, 0, a]$, where $a \neq 0$. This ternary function $H \notin \mathscr{P}$.

Case 2: $|R_G| = 2^k$. If for all $1 \leq j \leq k$, the ratio $G^{x_j=1}/G^{x_j=0}$ is a constant function c_j, (since $|R_G| = 2^k$ there are no divisions by zeros), then $G = c_0 \cdot [1, c_1] \otimes \cdots \otimes [1, c_k]$, where the constant $c_0 = G^{x_1=0,\ldots,x_k=0}$, and the unary function $[1, c_j]$ is on x_j. This gives $G \in \mathscr{P}$, a contradiction.

Now suppose for some j, $G^{x_j=1}/G^{x_j=0}$ is not a constant function. Without loss of generality, we assume $j = 1$. The Boolean hypercube on $(x_2, \ldots, x_k) \in \{0, 1\}^{k-1}$ is connected by edges that flip just one bit. Without loss of generality, along some flip of x_2 we have two different values of $G^{x_1=1}/G^{x_1=0}$. Suppose $G^{x_1=1}/G^{x_1=0}(0, a_3, \ldots, a_k) \neq G^{x_1=1}/G^{x_1=0}(1, a_3, \ldots, a_k)$, for some $a_3, \ldots, a_k$. Set $x_3 = a_3, \ldots, x_k = a_k$, we get a binary function $H(x_1, x_2) = G(x_1, x_2, a_3, \ldots, a_k)$. We have $H(1, 0)/H(0, 0) \neq H(1, 1)/H(0, 1)$, hence the rank of $H = \begin{bmatrix} H(0,0) & H(0,1) \\ H(1,0) & H(1,1) \end{bmatrix}$ is 2.

Suppose $H \in \mathscr{P}$. If a defining expression of H uses binary EQUALITY or DISEQUALITY functions, then we can set x_1, x_2 so that $H = 0$. But H is never zero. So H is defined by unary functions alone. But such a function has a matrix form of rank ≤ 1. This contradiction proves that $H \notin \mathscr{P}$ and completes our proof. □

3.5.4 Proof of Boolean #CSP Dichotomy

Now we are ready to complete the proof for the main theorem of this chapter: Theorem 3.7.

Proof of Theorem 3.7: In Section 3.4 we have shown that #CSP($\mathscr{A}$) and #CSP($\mathscr{P}$) are computable in polynomial time.

If $\mathcal{F} \not\subseteq \mathscr{A}$, by Lemma 3.19, either #CSP($\mathcal{F}$) is #P-hard, or we can simulate a unary function not in $\mathscr{A}$. After dividing by a nonzero factor, such a function takes the form $F = [1, \lambda]$, where $\lambda \notin \{0, \pm 1, \pm \mathrm{i}\}$. If $\mathcal{F} \not\subseteq \mathscr{P}$, by Lemma 3.20, either #CSP($\mathcal{F}$) is #P-hard, or we can simulate a function $P = [a, 0, 1, 0]$, or $P' = [0, 1, 0, a]$, where $a \neq 0$, or a binary function $H \notin \mathscr{P}$ having no zero values.

First, we prove #CSP($\{F, P\}$) is #P-hard. Clearly $P^{x_1=*} = [a, 1, 1]$. If $a \notin \{1, -1\}$, #CSP($[a, 1, 1]$) is #P-hard by Theorem 3.14. If $a \in \{1, -1\}$, we can construct $Q(x_1, x_2) = \sum_{x_3} P(x_1, x_2, x_3)F(x_3) = [a, \lambda, 1]$, which is $[\pm 1, \lambda, 1]$. Both problems #CSP($[\pm 1, \lambda, 1]$) are #P-hard by Theorem 3.14. The proof for #CSP($\{F, P'\}$) is the same.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

Secondly, we prove #CSP($\{F, H\}$) is #P-hard. After normalizing, we may suppose $H = \begin{bmatrix} 1 & x \\ y & z \end{bmatrix}$, where $xyz \neq 0$. Since $\mathscr{P}$ is a superset of $\mathscr{D}$, H is nondegenerate. Hence $z \neq xy$.

There are two cases, depending on whether $z = -xy$. For the case $z \neq -xy$, we construct a symmetric function $H(x_1, x_2)H(x_2, x_1) = [1, xy, z^2]$. By the conditions $xyz \neq 0, z \neq \pm xy$, it is impossible to be the first three tractable cases in Theorem 3.14. If it is in the last two tractable cases, then xy is a power of i. Now we can form the function $H(x_1, x_2)H(x_2, x_1)F(x_1)F(x_2)$, which is $[1, \lambda xy, \lambda^2 z^2]$. This function has no zero entry and its matrix form has rank 2, so it is not of the first three tractable cases in Theorem 3.14. If it were in the last two tractable cases, then λxy is a power of i, which implies that $\lambda = (\lambda xy)/(xy)$ itself is a power of i. However, since $[1, \lambda] \notin \mathscr{A}$, λ is not a power of i.

For the case $z = -xy$, We construct some binary functions with a nonnegative integer parameter s as follows:

$$\sum_{x_3} H(x_1, x_3)H(x_2, x_3)(F(x_3))^s = [1 + \lambda^s x^2, (y + \lambda^s xz), (y^2 + \lambda^s z^2)]$$
$$= [1 + \lambda^s x^2, y(1 - \lambda^s x^2), y^2(1 + \lambda^s x^2)].$$

As λ is not a power of i, at most one of the two values x^2 and λx^2 can be a power of i. Now we choose $s = 0$ or $s = 1$ so that $\lambda^s x^2 \notin \{\pm 1, \pm \mathrm{i}\}$.

After normalizing, we may write the function $[1 + \lambda^s x^2, y(1 - \lambda^s x^2), y^2(1 + \lambda^s x^2)]$ as $[1, y(1 - \lambda^s x^2)/(1 + \lambda^s x^2), y^2]$, noticing that $1 + \lambda^s x^2 \neq 0$. We claim that this function is not one of the five tractable cases from Theorem 3.14. Since there are no zero entries, clearly it is not the first two cases. Its matrix form has rank 2; therefore it is not the third case. If it were the fourth tractable case $[1, \pm \mathrm{i}, 1]$, then $y = \pm 1$, and $(1 - \lambda^s x^2)/(1 + \lambda^s x^2) = \pm \mathrm{i}$. This implies that $\lambda^s x^2 = \pm \mathrm{i}$, which is impossible. If $[1, y(1 - \lambda^s x^2)/(1 + \lambda^s x^2), y^2] = [1, \pm 1, -1]$, the fifth tractable case, then $y = \pm \mathrm{i}$, and again $(1 - \lambda^s x^2)/(1 + \lambda^s x^2) = \pm \mathrm{i}$, also impossible.

The proof of Theorem 3.7 is complete. □

For #CSP$^c(\mathcal{F})$ we have the following theorem. Notice that #CSP$^c(\mathcal{F})$ and #CSP($\mathcal{F}$) are really equivalent by Lemma 3.13. And since $\Delta_0, \Delta_1 \in \mathscr{A} \cap \mathscr{P}$, the tractability criteria of Theorems 3.7 and 3.21 are the same, as they should be. The point of Theorem 3.21 is that a reduction exists for #CSP$^c(\mathcal{F})$ when it is #P-hard that does not use pinning.

Theorem 3.21. *Suppose $\mathcal{F}$ is a set of functions mapping Boolean inputs to complex numbers. If $\mathcal{F} \subseteq \mathscr{A}$ or $\mathcal{F} \subseteq \mathscr{P}$, then* #CSP$^c(\mathcal{F})$ *is computable in*

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press

polynomial time. Otherwise, #CSP$^c(\mathcal{F})$ is #P-hard, and a reduction exists that does not use Lemma 3.13.

If we restrict to real (algebraic)-valued constraint functions in Theorem 3.7, we have the following theorem due to Bulatov, Dyer, Goldberg, Jalsenius, and Richerby.

Theorem 3.22 (Bulatov et al.). *Suppose $\mathcal{F}$ is a set of functions mapping Boolean inputs to (algebraic) real numbers. #CSP$(\mathcal{F})$ is #P-hard, unless*

- *Either every function in $\mathcal{F}$ takes the form $\lambda \cdot \chi_{AX} \cdot (-1)^{Q(X)}$, where $\lambda \in \mathbb{R}$, χ_{AX} is a 0-1 indicator function of an affine linear subspace over $\mathbb{Z}_2$, and $Q(X)$ is a quadratic polynomial on X over $\mathbb{Z}$;*
- *Or every function in $\mathcal{F}$ is a product of $(=_2)$, $(\neq_2)$ and real-valued unary functions.*

In these two cases, #CSP$(\mathcal{F})$ is computable in polynomial time.

Exercise: Prove that Theorem 3.22 follows from Theorem 3.7. Similarly show that Theorems 3.1 and 3.4 can be derived from Theorem 3.22 or Theorem 3.7.

https://doi.org/10.1017/9781107477063.004 Published online by Cambridge University Press