The complexity of weighted and unweighted #CSP ☆

Andrei Bulatov[a], Martin Dyer[b,*], Leslie Ann Goldberg[c], Markus Jalsenius[d], Mark Jerrum[e], and David Richerby[b]

[a] *School of Computing Science, Simon Fraser University, University Drive, Burnaby, Canada, V5A 1S6.*
[b] *School of Computing, University of Leeds, Leeds, LS2 9JT, UK.*
[c] *Department of Computer Science, University of Liverpool, Liverpool, L69 3BX, UK.*
[d] *Department of Computer Science, University of Bristol, Bristol, BS8 1UB, UK.*
[e] *School of Mathematical Sciences, Queen Mary, University of London, Mile End Road, London E1 4NS, UK.*

Abstract

We give some reductions among problems in (nonnegative) weighted #CSP which restrict the class of functions that needs to be considered in computational complexity studies. Our reductions can be applied to both exact and approximate computation. In particular, we show that a recent dichotomy for unweighted #CSP can be extended to rational-weighted #CSP.

Keywords: counting, constraint satisfaction, complexity theory

1. Introduction

The counting complexity of the weighted constraint satisfaction problem, for both exact and approximate computation, has been an active research area for several years. See, for example, [1–18]. The objective is to give a precise categorisation of the computational complexity of problems in a given class. Easily the most significant development in this stream of research was a recent result of Bulatov [1]. This establishes a dichotomy for exact counting in the whole of (unweighted) #CSP. The dichotomy is between problems in FP and problems which are #P-complete. Dyer and Richerby [16] have given an easier proof of this theorem, and have shown it to be decidable [17].

☆Research supported by an EPSRC grant "The complexity of counting in constraint satisfaction problems" (Dyer, Goldberg, Jalsenius, Jerrum and Richerby), and by an NSERC Discovery Grant (Bulatov).

*Corresponding author. Email: `M.E.Dyer@leeds.ac.uk`, Phone: `+44 113 343 5442`, Fax: `+44 113 343 5468`.

In this paper, we study equivalences among problems in weighted #CSP. These equivalences can greatly simplify the classes of problems which need to be considered in studies of computational complexity. A particular consequence of these results is that the dichotomy for unweighted #CSP can be extended to nonnegative rational-weighted #CSP. In the results we present here, the weights will usually lie in some subset of the nonnegative algebraic numbers, since the proofs do not appear to extend to negative weights [18] or complex weights [5]. Neither do we consider general real numbers, since we want our results to apply to standard models of computation and their complexity classes. An extension to a suitable model of real number computation may be possible, though statements about complexity would need to be modified appropriately.

The plan of the paper is as follows. In Section 1.1 we define the weighted constraint satisfaction problem and establish some notation. In Section 1.2, we define a notion of reducibility, which we call *weighted reduction*, that is used in all our proofs. Its advantage is that the same reductions apply to both exact and approximate computation. Section 2 proves the equivalence of unweighted and rational-weighted #CSP. Section 3 shows that a weighted #CSP problem can be assumed to have only one function, while retaining several useful restrictions on instances. Finally, in Section 4, we show that any rational-weighted problem is computationally equivalent to an unweighted problem with only binary constraints. Thus any #CSP problem is equivalent to a canonical digraph-labelling problem. This gives another proof of the equivalence of unweighted and rational-weighted #CSP.

1.1. Weighted constraint satisfaction

Let $\mathbb{Z}$, $\mathbb{Q}$, $\overline{\mathbb{Q}}$ and $\mathbb{A}$ denote the integers, rational numbers, real algebraic numbers, and (complex) algebraic numbers, respectively. Let $\mathbb{Z}_\geq$, $\mathbb{Q}_\geq$ and $\overline{\mathbb{Q}}_\geq$ denote the *nonnegative* numbers in $\mathbb{Z}$, $\mathbb{Q}$ and $\overline{\mathbb{Q}}$, respectively. The *positive* integers $\mathbb{Z}_\geq \setminus \{0\}$ will be denoted by $\mathbb{N}$, and the positive algebraic numbers $\overline{\mathbb{Q}}_\geq \setminus \{0\}$ by $\overline{\mathbb{Q}}_>$. Also $\mathbb{B}$ will denote $\{0,1\}$ and, if $n \in \mathbb{N}$, then $[n]$ will denote $\{1, 2, \dots, n\}$.

Let $D = \{0, 1, \dots, q-1\}$ ($q \in \mathbb{N}$), which we call the *domain*, and $\mathbb{K} \subseteq \mathbb{A}$, which we call the *codomain*. Let

$$\mathfrak{F}_r(D,\mathbb{K}) \;=\; \{f\colon D^r \to \mathbb{K}\}\,, \qquad \mathfrak{F}(D,\mathbb{K}) \;=\; \bigcup_{r\geq 1}\mathfrak{F}_r(D,\mathbb{K})\,,$$

denote the sets of functions of all *arities* from D to $\mathbb{K}$. We will write $r = r(f)$ for the arity of $f \in \mathfrak{F}(D,\mathbb{K})$. If $r(f) = 1$, f is called a *unary* function and, if $r(f) = 2$, it is a *binary* function.

A problem $\#\mathsf{CSP}(\mathcal{F})$ is parameterised by a finite set $\mathcal{F} \subset \mathfrak{F}(D,\mathbb{K})$ for some D and $\mathbb{K}$. An *instance* I of $\#\mathsf{CSP}(\mathcal{F})$ consists of a finite set of *variables* V and a finite set of *constraints* $\mathcal{C}$. A constraint $\kappa = \langle \mathbf{v}_\kappa, f_\kappa\rangle \in \mathcal{C}$ consists of a function $f_\kappa \in \mathcal{F}$ (of arity $r_\kappa = r(f_\kappa)$) and a *scope*, a sequence $\mathbf{v}_\kappa = (v_{\kappa,1},\ldots,v_{\kappa,r_\kappa})$ of variables from V, which need not be distinct. A *configuration* σ for the instance I is a function $\sigma\colon V \to D$. If $\mathbf{v} = (v_1,\ldots,v_r)$, we will write $\sigma(\mathbf{v})$ for $\big(\sigma(v_1),\ldots,\sigma(v_r)\big)$. Then the *weight* of the configuration σ is given by

$$\mathsf{w}(\sigma) = \prod_{\kappa\in\mathcal{C}} f_\kappa\big(\sigma(\mathbf{v}_\kappa)\big)\,.$$

Finally, the *partition function* $Z_{\mathcal{F}}(I)$ is given, for an instance I, by

$$Z_{\mathcal{F}}(I) = \sum_{\sigma\colon V\to D} \mathsf{w}(\sigma).$$

Then $\#\mathsf{CSP}(\mathcal{F})$ denotes the problem of computing the function $Z_{\mathcal{F}}$. We will write

$$\#\mathsf{CSP}_q[\mathbb{K}] = \{\#\mathsf{CSP}(\mathcal{F}) : \mathcal{F} \subset \mathfrak{F}(D,\mathbb{K}),\ |D| = q\}\,,\quad \#\mathsf{CSP}[\mathbb{K}] = \bigcup_{q=2}^{\infty} \#\mathsf{CSP}_q[\mathbb{K}].$$

The case $q = 1$ is clearly trivial, so we omit it from the definition of $\#\mathsf{CSP}[\mathbb{K}]$. The case $q = 2$ is called *Boolean* $\#\mathsf{CSP}[\mathbb{K}]$.

If Γ is a set of *relations*, as in [1, 2], we regard it as a set of functions $\mathcal{F}(\Gamma) \subset \mathfrak{F}(D,\mathbb{B})$, so $\#\mathsf{CSP}$ means $\#\mathsf{CSP}[\mathbb{B}]$. If $R \in \Gamma$ is r-ary, we define $f(R) \in \mathcal{F}$ so that, for each $\mathbf{a} \in D^r$, $f(\mathbf{a}) = 1$ if $\mathbf{a} \in R$, otherwise $f(\mathbf{a}) = 0$. Then we write $\#\mathsf{CSP}(\Gamma)$ rather than $\#\mathsf{CSP}(\mathcal{F}(\Gamma))$, and Z_Γ rather than $Z_{\mathcal{F}(\Gamma)}$.

We consider here only *non-uniform* $\#\mathsf{CSP}$, where D and $\mathcal{F}$ are considered to be objects of constant size. Thus it is only the variable set V, and the constraint set $\mathcal{C}$, that determine the size of an instance.

Various other restrictions on $\#\mathsf{CSP}[\mathbb{K}]$ have been considered in the literature, often in combination. For example, we may insist that $|\mathcal{F}| = m$, for some $m \in \mathbb{N}$, particularly $m = 1$, e.g. [5]. We may insist that no function has arity greater than r, for some $r \in \mathbb{N}$, particularly $r = 2$, e.g. [4]. We may insist that no variable occurs more than k times in an instance, e.g. [10]. We may insist that the functions in $\mathcal{F}$ possess some particular property, such as symmetry, e.g. [13]. We do not consider these restrictions in any detail here. However, we will make use of the following restricted version of #$\mathsf{CSP}[\mathbb{K}]$ in Section 4.

A unary function which must be applied *exactly once* to each variable $v \in V$ will be called a *vertex weighting*, and its function values *vertex weights*. Thus, if $\lambda\colon D \to \mathbb{K}$ is a vertex weighting, any instance I must contain exactly one constraint of the form $\langle (v), \lambda\rangle$ for each $v \in V$. Observe that, for general #CSP, it is unnecessary to allow multiple weightings $\lambda_1, \lambda_2, \ldots, \lambda_m$, since these can be combined into one equivalent vertex weighting $\lambda = \lambda_1\lambda_2\cdots\lambda_m$. (However, this does not necessarily remain true if the instances are restricted in some fashion.)

Our definition of vertex weights conforms to the use of similar terminology elsewhere, for example in [15]. We will denote the problem with $\mathcal{F} \subset \mathfrak{F}(D, \mathbb{K})$ and vertex weighting $\lambda\colon D \to \mathbb{K}$ by #$\mathsf{CSP}(\mathcal{F}; \lambda)$. The problem #$\mathsf{CSP}(\mathcal{F}; \lambda)$ is a restriction on the inputs to an associated #$\mathsf{CSP}[\mathbb{K}]$ problem, #$\mathsf{CSP}(\mathcal{F} \cup \{\lambda\})$. In an instance of #$\mathsf{CSP}(\mathcal{F} \cup \{\lambda\})$, $\langle (v), \lambda\rangle$ can appear any number of times, including zero, for each $v \in V$; in an instance of #$\mathsf{CSP}(\mathcal{F}; \lambda)$, $\langle (v), \lambda\rangle$ appears precisely once.

We will also consider *approximate* evaluation of $Z_{\mathcal{F}}$, meaning *relative* approximation. Thus, given $\epsilon > 0$ we wish to compute an estimate $\widehat{Z}_{\mathcal{F}}(I)$ of $Z_{\mathcal{F}}(I)$, for all I, such that

$$|\widehat{Z}_{\mathcal{F}}(I) - Z_{\mathcal{F}}(I)| \;\le\; \epsilon |Z_{\mathcal{F}}(I)|. \tag{1}$$

For *randomised* approximation, we require only that this holds with sufficient probability. See [9], for example, for further details. Observe that definition (1) applies equally if $Z_{\mathcal{F}}$ can take negative or complex values, though we consider only nonnegative real weights, here.

1.2. Weighted reductions

Let Σ be a finite alphabet, and let $F\colon \Sigma^* \to \mathbb{A}$. We are interested in evaluating F only for strings x that encode instances I of some computational problem.

However, we will make F into a total function by setting $F(x) = 0$ if $x \in \Sigma^*$ does not encode an instance. In particular, $F(\varepsilon) = 0$ for the empty string ε.

Definition 1. Let $F_1, F_2 \colon \Sigma^* \to \mathbb{A}$. A *weighted* reduction from F_1 to F_2 is a pair of FP-computable functions $\phi\colon \Sigma^* \to \overline{\mathbb{Q}}_>$, $\psi\colon \Sigma^* \to \Sigma^*$ such that $F_1(x) = \phi(x)F_2(\psi(x))$ for all $x \in \Sigma^*$.

In constructing a weighted reduction, we can clearly restrict attention to strings x that encode instances. Otherwise, we will simply take $\phi(x) = 1$, and $\psi(x) = \varepsilon$, the empty string.

Weighted reductions generalise the "*simulates*" concept defined in [11]. Parsimonious reductions [19] are contained as the special case $\phi(x) = 1$ for all $x \in \Sigma^*$. Weighted reduction relaxes the definition of parsimonious reduction by allowing a positive "weight" $\phi(x)$ for each $x \in \Sigma^*$. The generalisation is valuable in two respects. First, it preserves relative approximation of the functions F_1 and F_2 and, hence retains the most useful property of parsimonious reductions. If $\widehat{F}_2(x)$ is an approximation to $F_2(x)$ with relative error ϵ, it follows easily that $\phi(x)\widehat{F}_2(x)$ is an approximation to $F_1(x)$ with relative error ϵ. Weighted reduction is, in fact, a simple type of *AP-reduction*, as defined in [9].

Second, weighted reductions allow us to relax the cumbersome condition $F_1, F_2 \to \mathbb{Z}_\geq$, required by parsimonious reductions, so we can work with the natural classes of functions. All reductions used in this paper will be weighted reductions.

We write $F_1 \leq_{\mathrm{w}} F_2$ to indicate the existence of a weighted reduction from F_1 to F_2. If $F_1 \leq_{\mathrm{w}} F_2$ and $F_2 \leq_{\mathrm{w}} F_1$, we say that the functions are *equivalent* (under weighted reductions), and we write $F_1 \equiv_{\mathrm{w}} F_2$. Thus, if $F_1 \equiv_{\mathrm{w}} F_2$, then F_1 and F_2 will have the same computational complexity for both exact and approximate computation. This would not be true for approximate computation if we were to use the weaker notion of *Turing* reducibility, as is usual for exact computation in the class $\#\mathsf{P}$ [20].

If $\mathcal{F}_1$ and $\mathcal{F}_2$ are two classes of functions such that, for all $F_1 \in \mathcal{F}_1$, there is an $F_2 \in \mathcal{F}_2$ such that $F_1 \equiv_{\mathrm{w}} F_2$, and conversely, for all $F_2 \in \mathcal{F}_2$, there is an $F_1 \in \mathcal{F}_1$ such that $F_2 \equiv_{\mathrm{w}} F_1$, we will write $\mathcal{F}_1 \equiv_{\mathrm{w}} \mathcal{F}_2$.

The reason for making this definition in terms of equivalence, rather than reduction, is that, when $\mathcal{F}_1$ has a classification into functions of different complexity, for example a *dichotomy*, then this classification is inherited by any $\mathcal{F}_2 \equiv_{\mathrm{w}} \mathcal{F}_1$. In our proofs below, we will always have $\mathcal{F}_2 \subset \mathcal{F}_1$, so proving that $\mathcal{F}_1 \equiv_{\mathrm{w}} \mathcal{F}_2$ will only require showing that, for all $F_1 \in \mathcal{F}_1$, there is an $F_2 \in \mathcal{F}_2$ such that $F_1 \equiv_{\mathrm{w}} F_2$.

2. Equivalence of $\#\mathsf{CSP}[\mathbb{Q}_{\geq}]$ and $\#\mathsf{CSP}$

Under weighted reductions, we may assume that all instances of $\#\mathsf{CSP}(\mathcal{F})$ have every $v \in V$ appearing in the scope of some constraint. Otherwise, suppose the variables in $V_0 \subseteq V$ do not appear in the instance I, and let $n_0 = |V_0|$. Let I' be identical to I except that $V' = V \setminus V_0$. Then $Z_{\mathcal{F}}(I) = |D|^{n_0} Z_{\mathcal{F}}(I')$, so there is an equivalent problem of the required type using the reversible reduction $\phi(I) = |D|^{n_0}$ and $\psi(I) = I'$. We will assume that this has been done, so all variables in V appear in the scope of some constraint in $\mathcal{C}$.

Note also that *repeated* constraints are irrelevant in $\#\mathsf{CSP}$, but not in $\#\mathsf{CSP}[\mathbb{K}]$ when $\mathbb{K} \neq \mathbb{B}$. We may assume that instances of $\#\mathsf{CSP}(\Gamma)$ do not have repeated constraints, since otherwise there is trivial equivalence with this case.

First, suppose $\mathcal{F} \subset \mathfrak{F}(D, \mathbb{Q})$. Then, by computing a common denominator $N \in \mathbb{N}$ for the ranges of the functions in $\mathcal{F}$, we can write $f'(\mathbf{a}) = Nf(\mathbf{a})$, for each $f \in \mathcal{F}$ and we have $f'(\mathbf{a}) \in \mathbb{Z}$ for all $\mathbf{a} \in D^{r(f)}$. Let $\mathcal{F}' = \{f' : f \in \mathcal{F}\}$.

Lemma 1. *If $\mathcal{F}'$ is obtained from $\mathcal{F}$ as above, then $\#\mathsf{CSP}(\mathcal{F}) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{F}')$.*

Proof. If I is an instance of $\#\mathsf{CSP}(\mathcal{F})$, and I' is the corresponding instance of $\#\mathsf{CSP}(\mathcal{F}')$, we have $Z_{\mathcal{F}}(I) = N^{-k} Z_{\mathcal{F}'}(I')$, where $k = |\mathcal{C}(I)|$. Thus, letting $\phi(I) = N^{-k}$ and $\psi(I) = I'$, there is a weighted reduction from $Z_{\mathcal{F}}(I)$ to $Z_{\mathcal{F}'}(I')$, and hence $\#\mathsf{CSP}(\mathcal{F}) \leq_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{F}')$. Reversing this reduction gives $\#\mathsf{CSP}(\mathcal{F}) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{F}')$.

Corollary 1. $\#\mathsf{CSP}[\mathbb{Q}] \equiv_{\mathrm{w}} \#\mathsf{CSP}[\mathbb{Z}]$.

Proof. Since $\mathbb{Z} \subset \mathbb{Q}$, this follows immediately from Lemma 1.

Now, given $\mathcal{F} \subset \mathfrak{F}(D, \mathbb{Z}_{\geq})$, we will construct a set of relations $\Gamma(\mathcal{F})$, with domain A, as follows. For each function $f \in \mathcal{F}$ and each $\mathbf{a} \in D^r$, where $r = r(f)$, we create a set $\mathcal{D}_{f,\mathbf{a}}$ of cardinality $f(\mathbf{a})$ such that these sets are all mutually disjoint, and also disjoint from D. Let

$$A = D \cup \bigcup_{f\in\mathcal{F}} \bigcup_{\mathbf{a}\in D^r} \mathcal{D}_{f,\mathbf{a}}, \quad \text{so} \quad |A| = |D| + \sum_{f\in\mathcal{F}} \sum_{\mathbf{a}\in D^r} f(\mathbf{a})\,.$$

Now, for every $f \in \mathcal{F}$, we construct a relation $R(f) \subseteq A^{r+1}$, as follows. For each r-tuple $\mathbf{a}$ with $f(\mathbf{a}) > 0$, we create an $(r+1)$-tuple $(\mathbf{a}, w) \in R(f)$ for every $w \in \mathcal{D}_{f,\mathbf{a}}$.

Observation 1. *All tuples $(\mathbf{a}, w) \in R(f)$ have $\mathbf{a} \in D^r$ and $w \notin D$.*

Observation 2. *For each $w \in A \setminus D$, there is a unique $f \in \mathcal{F}$ and $\mathbf{a} \in D^r$ such that $(\mathbf{a}, w) \in R(f)$.*

We use these observations to prove the following equivalence.

Lemma 2. *If $\Gamma = \Gamma(\mathcal{F})$, as defined as above, then $\#\mathsf{CSP}(\mathcal{F}) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\Gamma)$.*

Proof. Suppose I is an instance of $\#\mathsf{CSP}(\mathcal{F})$. For each constraint $\kappa = \langle \mathbf{v}_\kappa, f_\kappa \rangle$, we create the constraint $\kappa' = \langle (\mathbf{v}_\kappa, v_\kappa), R(f_\kappa) \rangle$ in an instance I' of $\#\mathsf{CSP}(\Gamma)$, where v_κ is a new variable. Thus I' has variable set $V' = V \cup \{v_\kappa : \kappa \in \mathcal{C}\}$. Now, each configuration $\sigma \colon V \to D$ in I can be identified with the set of configurations $\sigma' \colon V' \to A$ in I' that agree with σ over V. Thus $\sigma'(\mathbf{v}_\kappa) = \sigma(\mathbf{v}_\kappa)$ and $\sigma'(v_\kappa) \in \mathcal{D}_{f_\kappa, \sigma(\mathbf{v}_\kappa)}$. By Observation 2, these partition the set of all σ' having nonzero weight. Since there are exactly $f(\sigma(\mathbf{v}_\kappa))$ choices for $\sigma'(v_\kappa)$, we have $Z_{\mathcal{F}}(I) = Z_\Gamma(I')$. We take $\phi(I) = 1$, $\psi(I) = I'$ and hence we have $\#\mathsf{CSP}(\mathcal{F}) \leq_{\mathrm{w}} \#\mathsf{CSP}(\Gamma)$.

Conversely, let I be an instance of $\#\mathsf{CSP}(\Gamma)$, and let $\kappa = \langle (\mathbf{v}, v), R(f) \rangle$ be any constraint. If v also appears in the tuple $\mathbf{v}'$ of a constraint $\kappa' = \langle (\mathbf{v}', v'), R(f') \rangle$, then there can be no configuration $\sigma \colon V \to A$ with nonzero weight, by Observation 1. Thus $Z_\Gamma(I) = 0$, so we take $\phi(I) = 1$ and $\psi(I) = \varepsilon$. Now, if v appears other than in constraint κ, it must be in a constraint $\kappa' = \langle (\mathbf{v}', v), R(f') \rangle$. But then, from Observation 2, any $\sigma \colon V \to A$ has nonzero weight only if $\sigma(\mathbf{v}') = \sigma(\mathbf{v})$ and $f' = f$. Thus we may add the equalities $\mathbf{v}' = \mathbf{v}$ and delete the constraint κ'. Repeating this procedure, we construct an instance I_0 of $\#\mathsf{CSP}(\Gamma)$ such that $Z_\Gamma(I_0) = Z_\Gamma(I)$, and each constraint $\kappa = \langle (\mathbf{v}, v), R(f) \rangle$ in the constraint set $\mathcal{C}_0$ of the instance I_0 has a unique variable $v = v_\kappa$. Thus I_0 is precisely the instance of $\#\mathsf{CSP}(\Gamma)$ which would result from applying the construction in the first part of the proof to the instance I_0' of #$\mathsf{CSP}(\mathcal{F})$ with variables $V' = V \setminus \{v_\kappa : \kappa \in \mathcal{C}_0\}$ and constraints $\langle \mathbf{v}_\kappa, f_\kappa \rangle$ $(\kappa \in \mathcal{C}_0)$. It follows that $Z_\Gamma(I) = Z_\Gamma(I_0) = Z_\mathcal{F}(I')$. So we may take $\phi(I) = 1$, $\psi(I) = I'$ and hence we have $\#\mathsf{CSP}(\Gamma) \leq_\mathrm{w} \#\mathsf{CSP}(\mathcal{F})$.

Remark 1. The reader will note that the size of the resulting unweighted problem increases dramatically with the size of the weights. Since these weights are constants in the non-uniform model, this has no impact on the complexity. However, we make no claims for the practicality of the reduction.

Theorem 1. $\#\mathsf{CSP}[\mathbb{Q}_\geq] \equiv_\mathrm{w} \#\mathsf{CSP}$.

Proof. This follows directly from $\mathbb{B} \subset \mathbb{Q}_\geq$ and Lemma 2.

As noted above, Bulatov [1] has shown a dichotomy for #CSP into problems which are in FP and problems which are #P-complete (see also [16]). Combining this with Theorem 1, and an argument given in Section 1.3 of [11], we have the following.

Theorem 2 (Dichotomy). *Any problem in* $\#\mathsf{CSP}[\mathbb{Q}_\geq]$ *is either in* FP *or is complete for* $\mathsf{FP}^{\#\mathsf{P}}$.

Remark 2. The method of proof used in Theorem 1 clearly fails for irrational weights. However, since this paper was written, Cai, Chen and Lu [6] have proved a general dichotomy theorem for weights in $\overline{\mathbb{Q}}_\geq$.

Remark 3. Theorem 1 may have analogues for mixed-sign and complex weights. However, the above method of proof encounters technical problems with repeated constraints in these cases.

3. Reduction to a single function

Here we consider $\mathcal{F} \subset \mathfrak{F}(D, \overline{\mathbb{Q}}_\geq)$. We will show that #$\mathsf{CSP}(\mathcal{F})$ is equivalent to #$\mathsf{CSP}(\{g\})$ for a single function $g \in \mathfrak{F}(D, \overline{\mathbb{Q}}_\geq)$. We abbreviate #$\mathsf{CSP}(\{g\})$ to #$\mathsf{CSP}(g)$.

We may assume that no $f \in \mathcal{F}$ is identically zero. Otherwise, if $f(\mathbf{a}) = 0$ for all $\mathbf{a} \in D^{r(f)}$, then $Z_\mathcal{F}(I) = 0$ for any instance I of #$\mathsf{CSP}(\mathcal{F})$ where f

appears in a constraint. Then, letting $\mathcal{F}' = \mathcal{F} \setminus \{f\}$, $Z_{\mathcal{F}}(I) = Z_{\mathcal{F}'}(I)$, we have $\#\mathsf{CSP}(\mathcal{F}) \leq_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{F}')$.

Let

$$M(f) = \sum_{\mathbf{a} \in D^r} f(\mathbf{a}) \ > \ 0 . \qquad (f \in \mathcal{F})$$

Now, let $\ell = |\mathcal{F}|$, and let $\mathcal{F} = \{g_1, g_2, \ldots, g_\ell\}$, $r_j = r(g_j)$ and $M_j = M(g_j)$ $(j \in [\ell])$. Let $s = \sum_{j=1}^{\ell} r_j$ and define $g \colon D^s \to \overline{\mathbb{Q}}_{\geq}$ by

$$g(\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_\ell) = \prod_{j=1}^{\ell} g_j(\mathbf{a}_j) \qquad (\mathbf{a}_j \in D^{r_j};\ j \in [\ell]) .$$

If $\kappa = \langle \mathbf{v}_\kappa, f_\kappa \rangle$ is a constraint of an instance I of $\#\mathsf{CSP}(\mathcal{F})$, let i_κ be defined by $i_\kappa = j$ if $f_\kappa = g_j$.

Theorem 3. *For all $\mathcal{F} \subset \mathfrak{F}(D, \overline{\mathbb{Q}}_{\geq})$, there exists $g \in \mathfrak{F}(D, \overline{\mathbb{Q}}_{\geq})$ such that $\#\mathsf{CSP}(\mathcal{F}) \equiv_{\mathrm{w}} \#\mathsf{CSP}(g)$.*

PROOF. The required g is the function $g(\mathcal{F})$ constructed above. For any instance I of $\#\mathsf{CSP}(\mathcal{F})$, construct an instance $I' = \psi(I)$ of $\#\mathsf{CSP}(g)$ by padding each constraint $\kappa = \langle \mathbf{v}_\kappa, f_\kappa \rangle$ that has $f_\kappa = g_{i_\kappa}$, to give

$$\kappa' \ = \ \big\langle \big(\mathbf{u}_{1,\kappa}, \ldots, \mathbf{u}_{i_\kappa - 1,\kappa}, \mathbf{v}_\kappa, \mathbf{u}_{i_\kappa + 1,\kappa}, \ldots, \mathbf{u}_{\ell,\kappa}\big), g \big\rangle ,$$

where $\mathbf{u}_{j,\kappa}$ $(j \in [\ell], j \neq i_\kappa)$ is an r_j-tuple of new variables not in V, and disjoint for each $j \neq i_\kappa$ and $\kappa \in \mathcal{C}$. Thus I' has variable set V', with

$$|V'| \ = \ |V| + \sum_{\kappa \in \mathcal{C}} \sum_{j \neq i_\kappa} r_j \ \leq \ s|\mathcal{C}| .$$

Any $\sigma' \colon V' \to D$ decomposes into $\sigma \colon V \to D$ and $\sigma_{j,\kappa} \colon \mathbf{u}_{j,\kappa} \to D$ $(j \neq i_\kappa, \kappa \in \mathcal{C})$. Clearly,

$$\sum_{\sigma_{j,\kappa}} g_j\big(\sigma_{j,\kappa}(\mathbf{u}_j)\big) \ = \ \sum_{\mathbf{a} \in D^{r_j}} g_j(\mathbf{a}) \ = \ M_j \ > \ 0 .$$

Thus, it follows that

$$Z_g(I') \ = \ \chi(I) Z_{\mathcal{F}}(I), \quad \text{where} \ \ \chi(I) \ = \prod_{\kappa \in \mathcal{C}} \prod_{j \neq i_\kappa} M_j \ > \ 0 ,$$

which gives a weighted reduction from $\#\mathsf{CSP}(\mathcal{F})$ to $\#\mathsf{CSP}(g)$ with $\phi(I) = 1/\chi(I)$.

The reduction in the other direction is straightforward. Suppose $\kappa = \langle \mathbf{v}_\kappa, g \rangle$ is any constraint of an arbitrary instance I of $\#\mathsf{CSP}(g)$, where $\mathbf{v}_\kappa = (\mathbf{v}_{1,\kappa}, \ldots, \mathbf{v}_{\ell,\kappa})$, with $\mathbf{v}_{j,\kappa} \in V^{r_j}$ $(j \in [\ell])$. Create the instance $I' = \psi(I)$ of $\#\mathsf{CSP}(\mathcal{F})$ with constraints $\mathcal{C}' = \{\langle \mathbf{v}_{j,\kappa}, f_j \rangle : j \in [\ell],\ \kappa \in \mathcal{C}\}$. Clearly, $Z_g(I) = Z_{\mathcal{F}}(I')$, so we have a weighted reduction from $\#\mathsf{CSP}(g)$ to $\#\mathsf{CSP}(\mathcal{F})$ with $\phi(I) = 1$.

Remark 4. Theorem 3 does not appear to carry over to negative or complex weights. The proof above fails because we may have $M(f) = 0$, so $\chi(I) = 0$, and hence $\phi(I)$ will be undefined.

The important features of the equivalence of Theorem 3 are

(i) it does not change the domain D;

(ii) it preserves approximation, since the reductions are weighted;

(iii) it preserves relations, since $\mathbb{B}$ is closed under product;

(iv) it preserves the maximum number of occurrences (*degree*) of variables.

Thus, for most complexity studies, allowing multiple functions or relations in #CSP does not increase generality. Theorem 3 can be used to simplify proofs given, for example, in [1, 8, 10–12, 16].

4. Reduction to binary constraints

The proof of equivalence of $\#\mathsf{CSP}[\mathbb{Q}_{\geq}]$ and #CSP in Section 2 is probably the simplest, but not the only construction. We present a different proof here, which is of interest in its own right. An instance of $\#\mathsf{CSP}(\mathcal{F})$ is reduced to an instance $\#\mathsf{CSP}(\Gamma)$, where Γ is a set of *binary* relations. Thus any problem in #CSP can be stated as an equivalent problem concerning *digraphs*.

We give the proof in two parts. In the first part, we show equivalence of any problem in $\#\mathsf{CSP}[\overline{\mathbb{Q}}_{\geq}]$ with a problem having a vertex weighting and a set of binary *relations*. We will then show that this vertex-weighted problem is equivalent to an unweighted digraph problem.

Theorem 4. *If $\mathcal{F}$ is a finite subset of $\mathfrak{F}(D, \overline{\mathbb{Q}}_{\geq})$, then $\#\mathsf{CSP}(\mathcal{F}) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{B}; \lambda)$, where $\mathcal{B}$ is a finite set of binary relations and $\lambda\colon D \to \overline{\mathbb{Q}}_{\geq}$ is a vertex weighting.*

Proof. We may assume, by Theorem 3, that $\mathcal{F} = \{g\}$ with $g \in \mathfrak{F}_r(D, \overline{\mathbb{Q}}_{\geq})$, for some r. Thus, to specify a constraint, we need only give its scope. We also assume that every variable appears in some scope, as discussed in Section 2. Then $\#\mathsf{CSP}(\mathcal{B}; \lambda)$ is specified as follows.

(a) The domain $A = D^r$, so $\mathbf{a} = (a_1, a_2, \ldots, a_r) \in A$ for all $a_1, a_2, \ldots, a_r \in D$.

(b) For all $\mathbf{a} \in A$, $\lambda(\mathbf{a}) = g(\mathbf{a})$.

(c) For each $i, k \in [r]$, there is a $\beta_{ik} \in \mathcal{B}$ such that for all $\mathbf{a}, \mathbf{b} \in A$,

$$\beta_{ik}(\mathbf{a}, \mathbf{b}) \;=\; \begin{cases} 1, & \text{if } \mathbf{a}_i = \mathbf{b}_k; \\ 0, & \text{otherwise.} \end{cases}$$

Let I be any instance of #$\mathsf{CSP}(g)$, with variable set V and constraint set $\mathcal{C}$. We can construct an equivalence relation $\sim$ on $\mathcal{C} \times [r]$ such that $(\iota, i) \sim (\kappa, k)$ if, and only if, $\mathbf{v}_{\iota,i}$ and $\mathbf{v}_{\kappa,k}$ are the same variable $v \in V$. Thus $\sim$ has $|V|$ equivalence classes, each class corresponding to a variable in V.

We now construct an instance $I' = \psi(I)$ of #$\mathsf{CSP}(\mathcal{B}; \lambda)$, which has variable set V' and constraint set $\mathcal{C}'$, as follows.

(i) For each $\kappa \in \mathcal{C}$, we have a variable $\kappa \in V'$. Thus $V' = \mathcal{C}$.

(ii) For all $\kappa \in V'$, we have one constraint $\langle (\kappa), \lambda \rangle \in \mathcal{C}'$. Thus λ is a vertex weighting.

(iii) For all $\iota, \kappa \in \mathcal{C}$, we have a constraint $\langle (\iota, \kappa), \beta_{ik} \rangle \in \mathcal{C}'$ if $(\iota, i) \sim (\kappa, k)$.

Let $\sigma \colon V \to D$ be any configuration for I. Then $(\iota, i) \sim (\kappa, k)$ implies $\beta_{ik}(\sigma(\mathbf{v}_\iota), \sigma(\mathbf{v}_\kappa)) = 1$. In turn, this implies $\sigma(\mathbf{v}_{\iota,i}) = \sigma(\mathbf{v}_{\kappa,k})$, as is required by the variables $\mathbf{v}_{\iota,i}$ and $\mathbf{v}_{\kappa,k}$ being identical. Thus there is a bijection between the configurations σ of I having nonzero weight and the configurations σ' of I' having nonzero weight. Let us write $\sigma' = \xi(\sigma)$ for this bijection. Note that $\sigma' = \xi(\sigma)$ then satisfies $\beta_{ik}\big(\sigma'(\iota), \sigma'(\kappa)\big) = \beta_{ik}\big(\sigma(\mathbf{v}_\iota), \sigma(\mathbf{v}_\kappa)\big) = 1$ if $(\iota, i) \sim (\kappa, k)$. Thus, with $\sigma' = \xi(\sigma)$, we have

$$\mathsf{w}(\sigma') \;=\; \prod_{\kappa \in \mathcal{C}} \lambda\big(\sigma'(\kappa)\big) \prod_{(\iota,i)\sim(\kappa,k)} \beta_{ik}\big(\sigma'(\iota), \sigma'(\kappa)\big) \;=\; \prod_{\kappa \in \mathcal{C}} g\big(\sigma(\mathbf{v}_\kappa)\big) \;=\; \mathsf{w}(\sigma),$$

so the bijection ξ is weight-preserving. Thus $Z_{\mathcal{B};\lambda}(I) = Z_g(I')$, and we have a weighted reduction from #$\mathsf{CSP}(g)$ to #$\mathsf{CSP}(\mathcal{B}; \lambda)$, with $\phi(I) = 1$.

Conversely, suppose I is any instance of #$\mathsf{CSP}(\mathcal{B}; \lambda)$ with variable set V and constraint set $\mathcal{C}$. We construct an instance $I' = \psi(I)$ of #$\mathsf{CSP}(g)$, with variable set V' and constraint set $\mathcal{C}'$, as follows. Note that A and the β_{ik} are not arbitrary, but have been derived as in (a) and (c) above. Thus, in particular, we can easily deduce the value of r. We now create a relation $\sim$ on the set

$V^* = V \times [r]$, as follows. For ease of notation, we will write $(u,i) \in V^*$ as u_i. For $u, v \in V$, let $u_i \sim v_k$ if there is a constraint $\langle (u,v), \beta_{ik}\rangle \in \mathcal{C}$.

Now, suppose σ is any configuration of I. Then, for any $v \in V$, we have $\sigma(v) = (a_1, \ldots, a_r) \in D^r$, from (a) above. Let us write $\sigma_i(v) = a_i$ $(i \in [r])$. Now, define $\sigma' \colon V^* \to D$ from σ by $\sigma'(v_i) = \sigma_i(v)$ for all $v \in V^*$, $i \in [r]$. We will write $\sigma' = \zeta(\sigma)$ for this function. If, for any $u, v \in V$, we have $u_i \sim v_k$, then we must have $\beta_{ik}(\sigma(u), \sigma(v)) = 1$. From (c) above, this implies that $\sigma_i(u) = \sigma_k(v)$, and hence $\sigma'(u_i) = \sigma'(v_k)$, where $\sigma' = \zeta(\sigma)$. Thus we can extend the relation $\sim$, as follows. We have $u_i \sim v_k$ if $\sigma'(u_i) = \sigma'(v_k)$ for all $\sigma' = \zeta(\sigma)$, where σ is a configuration of I. Clearly, $\sim$ is now an equivalence relation, which "identifies" the variables u_i and v_k. More precisely, the variable set V' of I' will be the set of equivalence classes $V^*/{\sim}$. For any $v_k \in V^*$, we write $\bar{v}_k$ for its equivalence class. Let $\sigma' \colon V^* \to D$ be such that $\sigma' = \zeta(\sigma)$ for some $\sigma \colon V \to D^r$. Then we can define $\bar{\sigma} \colon V' \to D$ by $\bar{\sigma}(\bar{v}_k) = \sigma(v_k)$ for all $v_k \in \bar{v}_k$. Thus we have constructed a bijection between the configurations σ of I having nonzero weight and the configurations $\bar{\sigma}$ of I' having nonzero weight. We will write $\bar{\sigma} = \xi(\sigma)$ for this bijection.

Now, I' will have constraint set

$$\mathcal{C}' = \{\langle \bar{\mathbf{v}}, g\rangle : \bar{\mathbf{v}} = (\bar{v}_1, \ldots, \bar{v}_r),\ v \in V\}\ .$$

Then, with $\bar{\sigma} = \xi(\sigma)$, we have

$$\mathsf{w}(\bar{\sigma}) \;=\; \prod_{v \in V'} g\big(\bar{\sigma}(\bar{\mathbf{v}})\big) = \prod_{v \in V} \lambda\big(\sigma(v)\big) \prod_{u_i \sim v_k} \beta_{ik}\big(\sigma(u), \sigma(v)\big) \;=\; \mathsf{w}(\sigma),$$

so the bijection ξ is weight-preserving. Thus $Z_g(I') = Z_{\mathcal{B};\lambda}(I)$, and we have a weighted reduction from $\#\mathsf{CSP}(\mathcal{B};\lambda)$ to $\#\mathsf{CSP}(g)$ with $\phi(I) = 1$.

Remark 5. In fact, Theorem 4 holds, more generally, for $\mathcal{F} \subset \mathfrak{F}(D, \mathbb{A})$. The proof above needs modification, however, since we cannot apply Theorem 3. Instead, we use binary relations $\beta^{f,h}_{i,j}$ for each $f, h \in \mathcal{F}$, $i \in [r(f)]$, $j \in [r(h)]$, and domain $A = \bigcup_{f \in \mathcal{F}} \{\mathbf{a}_f : \mathbf{a} \in D^r,\ r = r(f)\}$. We omit the details, since we currently have no application for this generalisation. The proof of Theorem 5 below is valid only for $\mathcal{F} \subset \mathfrak{F}(D, \mathbb{Q}_{\geq})$.

We can use this to give a different proof of the equivalence of $\#\mathsf{CSP}[\mathbb{Q}_{\geq}]$ and $\#\mathsf{CSP}$.

Theorem 5. *Let $\mathcal{B}$ be a set of binary relations, and let $\lambda\colon D \to \mathbb{Q}_{\geq}$ be a vertex weighting. Then $\#\mathsf{CSP}(\mathcal{B};\lambda) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\Gamma)$, where Γ is a set of binary relations.*

Proof. We will use the equivalence proved in Lemma 1. Thus we may take $\lambda\colon A \to \mathbb{Z}_{\geq}$. Then we use a construction similar to that of Section 2. Note that, if $\lambda(a) = 0$ for any $a \in A$, we can delete a from A. All configurations with $\sigma(v) = a$ for any $v \in V$ have zero weight and do not contribute to the partition function. Thus we may assume $\lambda(a) > 0$ for all $a \in A$. Then let

$$B_a \;=\; \{\,(a,i) : \, i \in [\lambda(a)]\,\} \quad (a \in A), \qquad B \;=\; \bigcup_{a\in A} B_a\,,$$

where we will again write (a,i) as a_i. Then Γ will comprise a set of binary relations γ on the domain B such that, for each $\beta \in \mathcal{B}$, there is a $\gamma(\beta)$ defined by

$$\gamma(\beta) \;=\; \{(a_i,b_j) : (a,b) \in \beta,\, i \in [\lambda(a)],\, j \in [\lambda(b)]\}\ .$$

Clearly, this gives a bijection between $\mathcal{B}$ and Γ, so we may also write $\beta = \beta(\gamma)$.

Now, let I be any instance of $\#\mathsf{CSP}(\mathcal{B};\lambda)$ with variable set V and constraint set $\mathcal{C}$. Then $I' = \psi(I)$ will have variable set $V' = V$ and constraint set

$$\mathcal{C}' \;=\; \{\langle(u,v),\gamma\rangle : \gamma = \gamma(\beta),\, \langle(u,v),\beta\rangle \in \mathcal{C}\}\ .$$

Let σ' be any satisfying configuration of I'. This can be mapped to a configuration σ of I satisfying all its binary constraints by $\sigma(v) = a$ if $\sigma'(v) = a_i$ for some $i \in [\lambda(a)]$. Let us write $\sigma = \eta(\sigma')$ for this function. Then,

$$\begin{aligned}\sum_{\sigma'\in\eta^{-1}(\sigma)} \mathsf{w}(\sigma') \;&=\; \left|\eta^{-1}(\sigma)\right| \;=\; \Big|\prod_{v\in V}\{\sigma'(v) : \sigma'(v) \in B_{\sigma(v)}\}\Big| \\ &=\; \prod_{v\in V}\left|B_{\sigma(v)}\right| \;=\; \prod_{v\in V}\lambda(\sigma(v)) \;=\; \mathsf{w}(\sigma).\end{aligned}$$

Thus $Z_{\mathcal{B};\lambda}(I) = Z_\Gamma(I')$, so and $\phi(I) = 1$, and we have shown $\#\mathsf{CSP}(\mathcal{B};\lambda) \leq_{\mathrm{w}} \#\mathsf{CSP}(\Gamma)$.

Conversely, if I is any instance of $\#\mathsf{CSP}(\Gamma)$ with variable set V' and constraint set $\mathcal{C}'$, we create an instance $I' = \psi(I)$ with variable set $V = V'$ and constraint set

$$\mathcal{C} \;=\; \{\langle(u,v),\beta\rangle : \langle(u,v),\gamma\rangle \in \mathcal{C}',\, \beta = \beta(\gamma)\}\ .$$

Reversing the above calculation yields $Z_{\mathcal{B};\lambda}(I') = Z_\Gamma(I)$, so $\phi(I) = 1$ and $\#\mathsf{CSP}(\Gamma) \leq_{\mathrm{w}} \#\mathsf{CSP}(\mathcal{B};\lambda)$. Hence $\#\mathsf{CSP}(\mathcal{B};\lambda) \equiv_{\mathrm{w}} \#\mathsf{CSP}(\Gamma)$.

Remark 6. Observe that this proof does not really require that the relations in $\mathcal{B}$ are all binary. We have made this restriction only for notational simplicity, and because it is the case needed for the following application.

Combining Theorems 4 and 5, we have an alternative proof of the results implied by Theorem 1. That is, $\#\mathsf{CSP}[\mathbb{Q}_{\geq}] \equiv_{\mathrm{w}} \#\mathsf{CSP}$ and a dichotomy theorem for $\#\mathsf{CSP}[\mathbb{Q}_{\geq}]$.

Remark 7. Theorems 4 and 5 determine a canonical form for $\#\mathsf{CSP}[\mathbb{Q}_{\geq}]$. The general problem is a set of k digraphs, $H_1, H_2, \ldots, H_k$ on the same vertex set D. An instance is a set of k digraphs, $G_1, G_2, \ldots, G_k$ on the same vertex set V. A satisfying configuration is a labelling of V‘ < with D that induces a homomorphism from G_i to H_i for all $i \in [k]$. Cai and Chen [4], have recently given a *decidable* dichotomy theorem for the case $k = 1$ of this problem.

Remark 8. The digraphs $H_1, H_2, \ldots, H_k$ in the canonical problem of Remark 7 can be taken to be *directed acyclic graphs* (*DAGs*), though possibly with loops. A decidable dichotomy theorem for the case $k = 1$ of this problem (without loops) was given by Dyer, Goldberg and Paterson [14]. The simplification can be justified as follows. Suppose we impose an arbitrary linear order on A. By the symmetries $\beta_{ij}(\mathbf{u}, \mathbf{v}) = \beta_{ji}(\mathbf{v}, \mathbf{u})$ in the proof of Theorem 4, we need only include $(\mathbf{u}, \mathbf{v})$ in the relation β_{ij} if $\mathbf{u} \leq \mathbf{v}$. Thus each β_{ij} describes a DAG, perhaps having loops on its vertices.

References

[1] A. Bulatov, The complexity of the counting constraint satisfaction problem, in *Proc. 35th International Colloquium on Automata, Languages and Programming (Part 1)*, 2008, pp. 646–661.

[2] A. Bulatov and V. Dalmau, Towards a dichotomy theorem for the counting constraint satisfaction problem, *Information and Computation* **205** (2007), 651–678.

[3] A. Bulatov, M. Dyer, L.A. Goldberg, M. Jalsenius and D. Richerby, The complexity of weighted Boolean #CSP with mixed signs, *Theoretical Computer Science* **410** (2009), 3949–3961.

[4] J.-Y. Cai and X. Chen, A decidable dichotomy theorem on directed graph homomorphisms with non-negative weights, in *Proc. 51st Annual IEEE Symposium on Foundations of Computer Science*, 2010, pp. 437–446.

[5] J.-Y. Cai, X. Chen and P. Lu, Graph homomorphisms with complex values: a dichotomy theorem, in *Proc. 37th International Colloquium on Automata, Languages and Programming (Part 1)*, 2010, pp. 275–286.

[6] J.-Y. Cai, X. Chen and P. Lu, Non-negatively weighted #CSP: An effective complexity dichotomy, in *Proc. 26th Annual IEEE Conference on Computational Complexity*, 2011, pp. 45–54.

[7] J.-Y. Cai, P. Lu and M. Xia, Holant problems and counting CSP, in *Proc. 41st Annual ACM Symposium on Theory of Computing*, 2009, pp. 715–724.

[8] N. Creignou and M. Hermann, Complexity of generalized satisfiability counting problems, *Information and Computation* **125** (1996), 1–12.

[9] M. Dyer, L.A. Goldberg, C. Greenhill and M. Jerrum, The relative complexity of approximate counting problems, *Algorithmica* **38** (2004), 471–500.

[10] M. Dyer, L.A. Goldberg, M. Jalsenius and D. Richerby, The complexity of approximating bounded-degree Boolean #CSP, in *Proc. 27th International Symposium on Theoretical Aspects of Computer Science*, 2010, pp. 323–334.

[11] M. Dyer, L.A. Goldberg and M. Jerrum, The complexity of weighted Boolean #CSP, *SIAM Journal on Computing* **38** (2009), 1970-1986.

[12] M. Dyer, L.A. Goldberg and M. Jerrum, An approximation trichotomy for Boolean #CSP, *Journal of Computer and System Sciences* **76** (2010), 267–277.

[13] M. Dyer, L.A. Goldberg and M. Jerrum, A complexity dichotomy for hypergraph partition functions, *Computational Complexity* **19** (2010), 605–633.

[14] M. Dyer, L.A. Goldberg, and M. Paterson, On counting homomorphisms to directed acyclic graphs, *Journal of the ACM* **54** (2007), 27:1–27:23.

[15] M. Dyer and C. Greenhill, The complexity of counting graph homomorphisms, *Random Structures and Algorithms* **17** (2000), 260–289.

[16] M. Dyer and D. Richerby, On the complexity of #CSP, in *Proc. 42nd Annual ACM Symposium on Theory of Computing*, 2010, pp. 725–734.

[17] M. Dyer and D. Richerby, The #CSP dichotomy is decidable, in *Proc. 28th International Symposium on Theoretical Aspects of Computer Science*, 2011, pp. 261-272.

[18] L.A. Goldberg, M. Grohe, M. Jerrum and M. Thurley, A complexity dichotomy for partition functions with mixed signs, in *Proc. 26th International Symposium on Theoretical Aspects of Computer Science*, 2009, pp. 493–504.

[19] J. Simon, *On some central problems in computational complexity*, PhD thesis, Cornell University, Ithaca, NY, USA, 1975.

[20] L. Valiant, The complexity of enumeration and reliability problems, *SIAM Journal on Computing* **8** (1979), 410–421.