Towards a Characterization of Constant-Factor Approximable Min CSPs*

Víctor Dalmau† Andrei Krokhin‡ Rajsekar Manokaran§

Abstract

We study the approximability of Minimum Constraint Satisfaction Problems (Min CSPs) with a fixed finite constraint language Γ on an arbitrary finite domain. The goal in such a problem is to minimize the number of unsatisfied constraints in a given instance of CSP(Γ). A recent result of Ene *et al.* says that, under the mild technical condition that Γ contains the equality relation, the basic LP relaxation is optimal for constant-factor approximation for Min CSP(Γ) unless the Unique Games Conjecture fails. Using the algebraic approach to the CSP, we introduce a new natural algebraic condition, stable probability distributions on symmetric polymorphisms of a constraint language, and show that the presence of such distributions on polymorphisms of each arity is necessary and sufficient for the finiteness of the integrality gap for the basic LP relaxation of Min CSP(Γ). We also show how stable distributions on symmetric polymorphisms can in principle be used to round solutions of the basic LP relaxation, and how, for several examples that cover all previously known cases, this leads to efficient constant-factor approximation algorithms for Min CSP(Γ). Finally, we show that the absence of another condition, which is implied by stable distributions, leads to NP-hardness of constant-factor approximation.

1 Introduction

The constraint satisfaction problem (CSP) provides a framework in which it is possible to express, in a natural way, many combinatorial problems encountered in computer science and AI [11, 12, 19]. Standard examples of CSPs include satisfiability of propositional formulas, graph colouring problems, and systems of linear equations. An instance of the CSP consists of a set of variables, a (not necessarily Boolean) domain of values, and a set of constraints on combinations of values that can be taken by certain subsets of variables. The aim is then to find an assignment of values to the variables that, in the decision version, satisfies all the constraints or, in the optimization version, maximizes (minimizes) the number of satisfied (unsatisfied, respectively) constraints.

Since the CSP is NP-hard in full generality, a major line of research in CSP tries to identify special cases that have desirable algorithmic properties (see, e.g. [12, 13]), the primary motivation being the general picture rather than specific applications. The two main ingredients of a constraint are: (a) variables to which it is applied, and (b) relations specifying the allowed combinations of values or the costs for all combinations. Therefore, the main types of restrictions on CSP are: (a) *structural* where the hypergraph formed by sets of variables appearing in individual constraints is restricted [20, 35], and (b) *language-based* where the constraint language Γ, i.e. the set of relations that can appear in constraints, is fixed (see, e.g. [9, 11, 12, 19]); the corresponding decision/maximization/minimization problems are denoted by CSP(Γ), Max CSP(Γ), and Min CSP(Γ), respectively. The ultimate sort of results in this direction are dichotomy results, pioneered by [38], which completely characterise the restrictions with a given desirable property modulo some complexity-theoretic assumptions. The language-based direction is considerably more active than the structural one, and there are many (partial and full) language-based complexity classification results, e.g. [5, 6, 8, 12, 15], but many central questions are still open.

The CSP has always played an important role in mapping the landscape of approximability of NP-hard optimization problems. For example, the famous PCP theorem has an equivalent reformulation in terms of inapproximability of a certain Max CSP(Γ), see [2]; moreover, Dinur's combinatorial proof of this theorem [16] deals entirely with CSPs. The first optimal inapproximability results [23] by Håstad were about problems Max CSP(Γ), and they led to the study of a new hardness notion called approximation resistance (see, e.g. [3, 24, 28]). The approximability of Boolean CSPs has been thoroughly investigated (see, e.g. [1, 12, 21, 22, 23, 24, 25, 26]). Much work around the Unique Games Conjecture (UGC) directly concerns CSPs [25]. This conjecture states that, for any $\epsilon > 0$, there is a large enough number $k = k(\epsilon)$ such that it NP-hard to tell ϵ-satisfiable from $(1-\epsilon)$-satisfiable instances of CSP(Γ_k), where Γ_k consists of all graphs of bijections on a k-element set. Many approximation algorithms for classical optimization problems have been shown optimal assuming the UGC [25, 26]. Raghavendra proved [37] that one SDP-based algorithm provides optimal approximation for all problems Max CSP(Γ) assuming the UGC. In this paper, we investigate problems Min CSP(Γ) on

*The first two authors are supported by UK EPSRC grant EP/J000078/01. The first author is also supported by MICCIN grant TIN2013-48031-C4-1.

†University Pompeu Fabra, Barcelona, Spain.

‡Durham University, UK.

§KTH, Stockholm, Sweden.

an arbitrary finite domain that belong to APX, i.e. admit a (polynomial-time) constant-factor approximation algorithm, proving some results that strongly indicate where the boundary of this property lies.

Related Work. Note that each problem Max CSP(Γ) trivially admits a constant-factor approximation algorithm because a random assignment of values to the variables is guaranteed to satisfy a constant fraction of constraints; this can be derandomized by the standard method of conditional probabilities. Clearly, for Min CSP(Γ) to admit such an algorithm, CSP(Γ) has to be polynomial-time solvable.

The approximability of problems Min CSP(Γ) has been studied, mostly in the Boolean case (such CSPs are sometimes called "generalized satisfiability" problems), see [1, 12]. We need a few concepts from propositional logic. A clause is *Horn* if it contains at most one positive literal, and *negative* it contains only negative literals. Let k-HORN be the constraint language over the Boolean domain that contains all Horn clauses with at most k variables. For $k \geq 2$, let k-IHBS be the subset of k-HORN that consists of all clauses that are negative or have at most 2 variables. It is known that, for each $k \geq 2$, Min CSP(k-IHBS) belongs to APX [12], and they (and the corresponding dual Horn problems) are essentially the only such Boolean Min CSPs unless the UGC fails [14]. For Min CSP(2-IHBS), which the same as Min CSP(2-HORN), a 2-approximation (LP-based) algorithm is described in [22], which is optimal assuming UGC. It is NP-hard to constant-factor approximate Min CSP(3-HORN) [21]. Let $\neq_2$ be the boolean relation $\{(0,1),(1,0)\}$, Min CSP($\{\neq_2\}$) is known as MinUnCut. Min CSP(Γ) where Γ consists of 2-clauses is known as Min 2CNF Deletion. The best currently known approximation algorithms for MinUnCut and Min 2CNF Deletion have approximation ratio $O(\sqrt{\log n})$ [1], and it follows from [26] that neither problem belongs to APX unless the UG conjecture is false. The UGC is known to imply the optimality of the basic LP relaxation for any Min CSP(Γ) such that Γ contains the equality relation (in fact, even for the more general Valued CSP) [18], extending the line of similar results for natural LP and SDP relaxations for various optimization CSPs [30, 34, 37]. An approximation algorithm for any Min CSP(Γ) is also given in [18] (that was claimed to match the LP integrality gap), but its analysis was later found to be faulty [41].

Constant-factor approximation algorithms for Min CSP are closely related to certain *robust agorithms* for CSP that attracted much attention recently [6, 14, 32]. Call an algorithm for CSP(Γ) *robust* if, for every $\epsilon > 0$ and every $(1-\epsilon)$-satisfiable instance of CSP(Γ) (i.e. at most an ϵ-fraction of constraints can be removed to make the instance satisfiable), it outputs a $(1 - f(\epsilon))$-satisfying assignment (i.e. that fails to satisfy at most a $f(\epsilon)$-fraction of constraints) where f is a function such that $f(\epsilon) \to 0$ as $\epsilon \to 0$ and $f(0) = 0$. Note that the running time of the algorithm should not depend on ϵ (which is unknown when the algorithm is run). CSPs admitting a robust algorithm (with some function f) were completely characterised in [6]; when such an algorithm exists, one can always choose $f(\epsilon) = O(\log\log(1/\epsilon)/\log(1/\epsilon))$ for the randomized algorithm and $f(\epsilon) = O(\log\log(1/\epsilon)/\sqrt{\log(1/\epsilon)})$ for the derandomized version. A robust algorithm is said to have *linear loss* if the function f can be chosen so that $f(\epsilon) = O(\epsilon)$. The problem of characterizing CSPs that admit a robust algorithm with linear loss was mentioned in [14]. It is easy to see that, for any Γ, CSP(Γ) admits a robust algorithm with linear loss if and only if Min CSP(Γ) has a constant-factor approximation algorithm. We will use this fact when referring to results in [14].

Many complexity classification results for CSP have been made possible by the introduction of the universal-algebraic approach (see [9, 11]), which extracts algebraic structure from a given constraint language Γ (via operations called *polymorphisms* of Γ) and uses it to analyze problem instances. The universal-algebraic framework to study robust algorithms with a given loss was presented in [14], this approach was also used in [6, 32]. In this paper, we apply this framework with some old and some new algebraic conditions to study problems Min CSP(Γ). Our algebraic conditions use symmetric operations, which appear naturally when LP-based algorithms are used for CSPs; others recent examples are [32, 40].

Contributions. Most of our results assume that Γ contains the equality relation. We characterise problems Min CSP(Γ) for which the basic LP relaxation has finite integrality gap. Our characterization uses the algebraic approach to CSP that has been extremely fruitful in proving complexity classification results for CSPs. The key notion in this approach is that of a *polymorphism* which is, roughly, an operation preserving relations in Γ (see [9, 11]). The characterizing condition is in terms of stable probability distributions on symmetric polymorphisms of Γ. This condition can in principle be used to design efficient constant-factor approximation algorithms, provided one can efficiently sample from these distributions. We show that this is possible for some examples that cover all cases where such algorithms were previously known to exist.

We also strengthen our UG-hardness result to NP-hardness for a class of Min CSPs. A near-unanimity polymorphism is a type of polymorphism well known in the algebraic theory of CSP [7, 11, 19], and its presence follows from the existence of those stable distributions. We show Min CSP(Γ) is NP-hard to constant-factor approximate if Γ has no near-unanimity polymorphism.

2 Preliminaries

Let A be a finite set. A k-tuple $\mathbf{a} = (a_1, \ldots, a_k)$ is any element of A^k. A k-ary relation on A is a subset of A^k. We shall use arity(R) to denote the arity of relation R.

An *instance* of the CSP is a triple $I = (V, A, \mathscr{C})$ with V a finite set of *variables*, A a finite set called *domain*, and $\mathscr{C}$ a finite list of *constraints*. Each constraint in $\mathscr{C}$ is a pair $C = (\mathbf{v}, R)$ where $\mathbf{v} = (v_1, \ldots, v_k)$ is a tuple of variables of length k, called the *scope* of C, and R an k-ary relation on D, called the *constraint relation* of C. The *arity* of a constraint C, arity(C), is the arity of its constraint relation. When considering optimization problems, we will assume that each constraint has a weight $w_C \in \mathbb{Q}_{>0}$. It is known (see, e.g. Lemma 7.2 in [12]) that allowing weights in Min CSP(Γ) does not affect membership in APX.

Very often we will say that a constraint C belongs to instance I when, strictly speaking, we should be saying that appears in the constraint list $\mathscr{C}$ of I. Also, we might sometimes write $(v_1, \ldots, v_k, R)$ instead of $((v_1, \ldots, v_k), R)$. A *constraint language* is any *finite* set Γ of relations on A. The problem CSP(Γ) consists of all instances of the CSP where all the constraint relations are from Γ. An assignment for I is a mapping $s : V \to A$. We say that s *satisfies* a constraint $(\mathbf{v}, R)$ if $s(\mathbf{v}) \in R$ (where s is applied component-wise).

The *decision problem* for CSP(Γ) asks whether an input instance I of CSP(Γ) has a solution, i.e., an assignment satisfying all constraints. The natural *optimization problems* for CSP(Γ), Max CSP(Γ) and Min CSP(Γ), ask to find an assignment that maximizes the total weight of satisfied constraints or minimizes the total weight of unsatisfied constraints, respectively.

2.1 Basic linear program Many approximation algorithms for optimization CSPs use the basic (aka standard) linear programming (LP) relaxation [14, 32, 40]. We use a natural adaption of this LP to Min CSPs.

For any instance $I = (V, A, \mathscr{C})$ of Min CSP(Γ), there is an equivalent canonical 0-1 integer program. It has variables $p_v(a)$ for every $v \in V$, $a \in A$, as well as variables $p_C(\mathbf{a})$ for every constraint $C = (\mathbf{v}, R)$ and every tuple $\mathbf{a} \in A^{\text{arity}(R)}$. The interpretation of $p_v(a) = 1$ is that variable v is assigned value a; the interpretation of $p_C(\mathbf{a}) = 1$ is that $\mathbf{v}$ is assigned (component-wise) tuple $\mathbf{a}$. More formally, the program ILP is the following:

$$\text{minimize:} \quad \sum_{C=(\mathbf{v},R)\in\mathscr{C}} w_C \cdot (1 - p_C(R))$$

subject to:

$$(2.1) \quad p_v(A) = 1 \qquad \text{for } v \in V;$$

$$(2.2) \quad p_C(A^{j-1} \times \{a\} \times A^{\text{arity}(C)-j}) = p_{\mathbf{v}_j}(a) \quad \text{for } C = (\mathbf{v}, R) \in \mathscr{C}, 1 \le j \le \text{arity}(C), a \in A.$$

Here, for every $v \in V$ and $S \subseteq A$, $p_v(S)$ is a shorthand for $\sum_{a\in S} p_v(a)$ and for every C and every $T \subseteq A^{\text{arity}(C)}$, $p_C(T)$ is a shorthand for $\sum_{\mathbf{a}\in T} p_C(\mathbf{a})$.

If we relax this ILP by allowing the variables to take values in the range $[0, 1]$ instead of $\{0, 1\}$, we obtain the *basic linear programming relaxation* for I, which we denote by BLP(I). As Γ is fixed, an optimal solution to BLP(I) can be computed in time polynomial in $|I|$.

For an instance I of Min CSP(Γ), we denote by $\mathsf{Opt}(I)$ the value of an optimal solution to I, and by $\mathsf{Opt}_{\mathsf{LP}}(I)$ the value of an optimal solution to BLP(I).

For any finite set X, we shall denote by $\Delta(X)$ the set of all probability distributions on X. Furthemore, for any $n \in \mathbb{N}$, we shall denote by $\Delta_n(X)$ the subset of $\Delta(X)$ containing every $q \in \Delta(X)$ such that $q(x) \cdot n$ is an integer for every $x \in X$. To simplify notation we shall write Δ_n and Δ as a shorthand of $\Delta_n(A)$ and $\Delta(A)$ respectively. If $p \in \Delta(A^r)$ and $p_1, \ldots, p_r \in \Delta(A)$ will say that *the marginals of p are $p_1, \ldots, p_r$* to indicate that for every $1 \le i \le r$, and every $a \in A$, $p(A^{i-1} \times \{a\} \times A^{r-i}) = p_i(a)$.

Restriction (2.1) of BLP(I) expresses the fact that, for each $v \in V$, $p_v \in \Delta(A)$. Also, (2.1) and (2.2) together express the fact that, for each constraint $C = (\mathbf{v}, R)$, of arity k, we have $p_C \in \Delta(A^k)$ and that the marginals of the p_C distribution are consistent with the p_v distributions.

Recall that the *integrality gap* of BLP for Min CSP(Γ) is defined as

$$\sup_I \frac{\mathsf{Opt}(I)}{\mathsf{Opt}_{\mathsf{LP}}(I)}$$

where the supremum is taken over all instances I of Min CSP(Γ).

Theorem 2.1. ([18]) *Let Γ be a constraint language such that* $\text{eq}_A \in \Gamma$ *and let α_{gap} be the integality gap of BLP for* Min CSP(Γ)*. For every real number $\beta < \alpha_{gap}$, it is NP-hard to approximate* Min CSP(Γ) *to within a factor β unless the UGC is false. In particular, if the integrality gap is infinite then there is no constant-factor approximation algorithm for* Min CSP(Γ) *unless the UGC is false.*

In fact, the result from [18] is more general because it holds for $[0, 1]$-valued CSPs, i.e. when each constraint is a function taking values in $[0, 1]$, rather than only in $\{0, 1\}$. The setting in [18] assumes that each variable in an instance comes with its own list of allowed images (i.e. a subset of A), but this assumption is not essential in their reduction from the UGC.

2.2 Algebra Most of the terminology introduced in this section is standard. See [9, 11] for more detail about the algebraic approach to the CSP. An n-ary *operation* on A is a map $f : A^n \to A$. Let us now define several types of operations that will be used in this paper. We usually define operations by identities, i.e. by equations where all variables are assumed to be universally quantified.

- An operation f is *idempotent* if it satisfies the identity $f(x, \ldots, x) = x$.
- An operation f is *symmetric* if $f(x_1, \ldots, x_n) = f(x_{\pi(1)}, \ldots, x_{\pi(n)})$ for each permutation π on $\{1, \ldots, n\}$.

Thus, a symmetric operation is one that depends only on the multiset of its arguments. Since there is an obvious one-to-one correspondence between $\Delta_n(A)$ and multisets of size n, n-ary symmetric operations on A can be naturally identified with functions from $\Delta_n(A)$ to A.

- An n-ary operation f on A is *totally symmetric* if $f(x_1, \ldots, x_n) = f(y_1, \ldots, y_n)$ whenever $\{x_1, \ldots, x_n\} = \{y_1, \ldots, y_n\}$.
- An n-ary ($n \geq 3$) operation f on A is called an NU (near-unanimity) operation if it satisfies the identities

$$f(y, x, x \ldots, x, x) = f(x, y, x \ldots x, x) = \cdots = \\ = f(x, x, x \ldots x, y) = x.$$

An n-ary operation f on A *preserves* (or is a *polymorphism* of) a k-ary relation R on A if for every n (not necessarily distinct) tuples $(a_1^i, \ldots, a_k^i) \in R$, $1 \leq i \leq n$, the tuple

$$(f(a_1^1, \ldots, a_1^n), \ldots, f(a_k^1, \ldots, a_k^n))$$

belongs to R as well. Given a set Γ of relations on A, we denote by $\mathrm{Pol}(\Gamma)$ the set of all operations f such that f preserves each relation in Γ. If $f \in \mathrm{Pol}(\Gamma)$ then Γ is said to be *invariant* under f. If R is a relation we might freely write $\mathrm{Pol}(R)$ to denote $\mathrm{Pol}(\{R\})$.

Example 1. *Let $A = \{0, 1\}$.*

1. *It is well known and easy to check that, for each $n \geq 1$, the n-ary (totally symmetric) operation $f(x_1, \ldots, x_n) = \bigwedge_{i=1}^n x_i$ is a polymorphism of* 3-HORN.
2. *It is well known and easy to check that, for each $k \geq 2$, constraint language k*-IHBS*, as defined in Section 1, has polymorphism $x \wedge (y \vee z)$, but the operation $x \vee y$ is not a polymorphism of k*-IHBS.

The complexity of constant-factor approximation for Min CSP(Γ) is completely determined by $\mathrm{Pol}(\Gamma)$, as the next theorem indicates.

Theorem 2.2. ([14]) *Let Γ and Γ' be constraint languages on A such that* $\mathrm{Pol}(\Gamma) \subseteq \mathrm{Pol}(\Gamma')$. *Assume, in addition, that Γ contains the equality relation. Then, if* Min CSP(Γ) *has a constant-factor approximation algorithm then so does* Min CSP(Γ').

Say that BLP *decides* CSP(Γ) if, for any instance I of CSP(Γ), I is satisfiable whenever $\mathsf{Opt}_{\mathsf{LP}}(I) = 0$.

Theorem 2.3. ([32]) *For any Γ, The following are equivalent:*

1. BLP *decides* CSP(Γ),
2. *Γ has symmetric polymorphisms of all arities.*

Note that symmetric polymorphisms provide a natural rounding for BLP(I), as follows. Let s be an optimal solution to BLP(I) in which all variables are assigned rational numbers such that, for some $n \in \mathbb{N}$, $p_v \in \Delta_n(A)$ for each variable v in I and $p_C \in \Delta_n(A^{arity(C)})$ for each constraint C in I. Then each v can be assigned the element $f(p_v)$ where f is an n-ary symmetric polymorphism of Γ. It is not hard to check (or see [32]) that if $\mathsf{Opt}_{\mathsf{LP}}(I) = 0$ then this assignment will satisfy all constraints in I.

It was claimed in [32] that the conditions in Theorem 2.3 are also equivalent to the condition of having totally symmetric polymorphisms of all arities, but a flaw was later discovered in the proof of this claim, and indeed a counterexample (see Section 3.2) was found by G. Kun [31].

3 Results

We will formulate most of our results for constraint languages Γ that contain the equality relation eq. We make this restriction because some of the reductions in this paper and some papers that we use are currently known to work only with this restriction. We conjecture that this restriction is not essential, that is, for any Γ, Min CSP(Γ) admits a constant-factor approximation algorithm if and only if Min CSP($\Gamma \cup \{\mathrm{eq}\}$) does so (though the optimal constants may differ).

As mentioned before, for any Γ, CSP(Γ) admits a robust algorithm with linear loss if and only if Min CSP(Γ) has a constant-factor approximation algorithm. Hence, we can use results from Section 3 of [14] and assume, without loss of generality, that Γ contains the equality relation and all unary singletons, i.e., relations $\{a\}$, $a \in A$. Note that the latter condition implies that all polymorphisms of Γ are idempotent.

3.1 Finite integrality gaps Theorem 2.1 provides evidence that the BLP is optimal to design constant-factor approximation algorithms for Min CSP(Γ). In this subsection, we characterize problems Min CSP(Γ) for which BLP has a finite integrality gap.

For $p, q \in \Delta$, let $\mathrm{dist}(p, q) = \max_{a \in A} |p(a) - q(a)|$. For a tuple $\mathbf{a} \in A^n$, let $d_{\mathbf{a}} \in \Delta_n$ be such that each element $x \in A$ appears in $\mathbf{a}$ exactly $n \cdot d_{\mathbf{a}}(x)$ times. For tuples $\mathbf{a}, \mathbf{b} \in A^n$, define $\mathrm{dist}(\mathbf{a}, \mathbf{b}) = \mathrm{dist}(d_{\mathbf{a}}, d_{\mathbf{b}})$.

An n-ary *fractional operation* ϕ on A is any probability distribution on the set of n-ary operations on A. For every real number $c \geq 0$, call ϕ *c-stable* if, for all $\mathbf{a}, \mathbf{b} \in A^n$, we have $\Pr_{g \sim \phi}\{g(\mathbf{a}) \neq g(\mathbf{b})\} \leq c \cdot \mathrm{dist}(\mathbf{a}, \mathbf{b})$.

Theorem 3.1. *For any Γ containing eq_A, the following are equivalent:*

1. *The integrality gap of BLP for* Min CSP(Γ) *is finite.*
2. *There is $c \geq 0$ such that, for each $n \in \mathbb{N}$, there is an*

n-ary c-stable fractional operation ϕ_n on A whose support consists of symmetric polymorphisms of Γ.

Example 2. *Recall Example 1. It is known and not hard to check that the operation f_n is the only n-ary symmetric polymorphism of* 3-HORN. *By choosing* $\mathbf{a} = (1, 1, \ldots, 1)$ *and* $\mathbf{b} = (0, 1, \ldots, 1)$*, it follows easily that there is no $c \geq 0$ such that* 3-HORN *has a c-stable distribution ϕ_n (as in Theorem 3.1) for each n. Hence, the integrality gap of* BLP *for* Min CSP(3-HORN) *is infinite.*

To prove Theorem 3.1 we need a few definitions and intermediate results.

Let I be any weighted instance in Min CSP(Γ) with variable set V. A *fractional assignment* for I is any probability distribution, ϕ, on the set of assignments for I. For a real number $c \geq 1$, we say that a fractional assignment ϕ for I is *c-bounded* if, for every constraint $C = (v_1, \ldots, v_r, R)$ in I,

$$\Pr_{g \sim \phi}\{g(v_1), \ldots, g(v_r)) \notin R\} \leq c \cdot (1 - w_C)$$

where w_C is the weight in I of constraint C. We will apply it only to instances where $w_C \in [0, 1]$.

For every relation $R \in \Gamma$ of arity, say r, and every $p_1, \ldots, p_r \in \Delta$ define $\text{loss}(p_1, \ldots, p_r, R) \in [0, 1]$ to be $\min_p(1 - p(R))$ where p ranges over all the probability distributions on A^r with marginals $p_1, \ldots, p_r$.

In a technical sense, function loss 'encodes' the contribution of each constraint in optimal solutions of BLP. This is formalized in the following observation.

Observation 1. *Let I be any instance of* Min CSP(Γ) *and let $C = (v_1, \ldots, v_r, R)$ be any of its constraints. Then* $1 - p_C(R) = \text{loss}(p_{v_1}, \ldots, p_{v_r}, R)$ *holds in any optimal solution of* BLP(I).

For every $n \in \mathbb{N}$, the n-th *universal instance* for Γ, $U_n(\Gamma)$, is the instance with variable set Δ_n containing for every relation R of arity, say r, in Γ, and every $p_1, \ldots, p_r \in \Delta_n$, constraint $(p_1, \ldots, p_r, R)$ with weight $1 - \text{loss}(p_1, \ldots, p_r, R)$. We write simply U_n if Γ is clear from the context.

The following is a variant of Farkas' lemma (obtained easily from Corollary 7.1f in [39]) that we will use in our proofs.

Lemma 3.1. *(Farkas' Lemma) Let M be a $m \times n$ matrix, $b \in \mathbb{R}^m$. Then exactly one of the following two statements is true:*

- *There is an $x \in (\mathbb{R}_{\geq 0})^n$ with $||x||_1 = 1$ ($||x||_1$ denotes the 1-norm of x) such that $Mx \geq b$.*
- *There is an $y \in (\mathbb{R}_{\geq 0})^m$ with $||y||_1 = 1$ such that $yb > yM$.*

Theorem 3.1 follows directly from Lemmas 3.2 and 3.3 below.

Lemma 3.2. *For every constraint language Γ and $c \geq 1$, the following are equivalent:*

1. *The integrality gap of BLP for* Min CSP(Γ) *is at most c.*
2. *For each $n \in \mathbb{N}$, there is a c-bounded fractional assignment for U_n.*

Proof. This proof is an adaptation of the proof of Theorem 4.2 in [40], and it also works for valued CSPs.

($2 \Rightarrow 1$) Let $I = (V, A, \mathscr{C})$ be any instance of Min CSP(Γ), and let $p_v (v \in V)$, $p_C (C \in \mathscr{C})$ by any optimal solution of BLP(I). We can assume that there exists $n \in \mathbb{N}$ such that $p_v \in \Delta_n$ for every $v \in V$. For every assignment g for U_n, let s_g be the assignment for I defined as $s_g(v) = g(p_v), v \in V$.

Since (2) holds, it follows from Observation 1 and the definition of c-boundedness that, for every constraint $C = (v_1, \ldots, v_r, R)$ in I, we have

$$\Pr_{g \sim \phi}\{(s_g(v_1), \ldots, s_g(v_r)) \notin R\} \leq c \cdot (1 - p_C(R))$$

It follows that the expected value of s_g is at most $c \cdot \mathsf{Opt}_{\mathsf{LP}}(I)$. Consequently, there exists some s_g with value at most $c \cdot \mathsf{Opt}_{\mathsf{LP}}(I)$.

($1 \Rightarrow 2$) We shall prove the contrapositive. Assume that for some $n \in \mathbb{N}$, there is no c-bounded fractional assignment for U_n. We shall write a system of linear inequalities that expresses the existence of a c-bounded fractional assignment for U_n and then apply Lemma 3.1 to this system. To this end, we introduce a variable x_g for every assignment g for U_n. The system contains, for every constraint $C = (p_1, \ldots, p_r, R)$ in U_n, the inequality:

$$\sum_{g \in G_n} x_g \cdot \mathbf{1}[(g(p_1), \ldots, g(p_r)) \notin R] \leq c \cdot \text{loss}(C)$$

where G_n is the set of all assignments for U_n and $\mathbf{1}[(g(p_1), \ldots, g(p_r)) \notin R]$ is 1 if $g(p_1), \ldots, g(p_r)) \notin R$ and 0 otherwise. Note that the system does not include equations for $x_g \geq 0$ and $\sum_{g \in G_n} x_g = 1$ since this is already built-in in the version of Farkas' lemma that we use.

Since there is no c-bounded fractional assignment for U_n it follows from Farkas' Lemma that the system containing for every $g \in G_n$ inequality

$$\sum_{C=(p_1,\ldots,p_r,R)\in U_n} y_C \cdot c \cdot \text{loss}(C) < \tag{3.3}$$

$$< \sum_{C=R(p_1,\ldots,p_r,R)\in U_n} y_C \cdot \mathbf{1}[(g(p_1), \ldots, g(p_r)) \notin R]$$

has a solution where every variable y_C takes non-negative values and it holds that $\sum_C y_C = 1$. We can also assume the value of every variable in the solution is rational, since so are all the coefficients in the system.

Now consider instance $I = (V, A, \mathscr{C})$ where $V = \Delta_n$ and $\mathscr{C}$ contains, for every relation $R \in \Gamma$ of arity, say r, and every $p_1, \ldots, p_r \in \Delta_n$, constraint $C = (v_1, \ldots, v_r, R)$ with weight y_C.

We shall construct a solution $p_v(v \in \Delta_n), p_C(C \in \mathscr{C})$ of BLP(I). For every $v \in \Delta_n$, set p_v to v (note that v is a distribution on A). For every $C \in \mathscr{C}$ set p_C to the distribution q with $1-q(C) = \text{loss}(C)$. Hence, the objective value of the solution of BLP(I) thus constructed is $\sum_{C\in U_n} y_C \cdot \text{loss}(C)$, which is c times smaller than the left side of inequality (3.3). Furthermore, the total weight of falsified constrains by any assignment g for I is precisely the right side of inequality (3.3). It follows that the gap of instance I is larger than c.

For every set X, one can associate to every $p \in \Delta_n(X)$ the multiset p' such every element $x \in X$ occurs in p' exactly $p(x) \cdot n$ times. In a similar way, one obtains a one-to-one correspondence between the assignments (resp. fractional assignments) for U_n and the symmetric operations (resp. fractional operations with support consisting of symmetric operations).

Lemma 3.3. *For every constraint language Γ containing the equality relation, the following are equivalent:*

1. *There is $c \geq 1$, such that for each $n \in \mathbb{N}$, there is a c-bounded fractional assignment for $U_n(\Gamma)$.*
2. *There is $c \geq 0$ such that, for each $n \in \mathbb{N}$, there is an n-ary c-stable fractional operation on A whose support consists of symmetric polymorphisms of Γ.*

Proof. In this proof it is convenient to distinguish formally between a multiset y (resp. operations, fractional operation) and its associated distribution (resp. assignment, fractional assignment) that, whenever X and n are clear from the context, we shall denote by y'. The following observation will be useful.

Observation 2. *For any assignment g for U_n and any distribution $p \in \Delta_n(A^r)$, $(g(p_1), \ldots, g(p_r)) = g'(p')$ where $p_1, \ldots, p_r$ are the marginals of p and $g'(p')$ denotes the r-ary tuple obtained applying the symmetric n-ary operation g' (corresponding to g) to the (n) tuples in p' component-wise.*

(1) ⇒ (2) Assume that ϕ is a c-bounded fractional assignment for U_n. We claim that for every mapping g in the support of ϕ, g' is, in fact, a polymorphism of Γ. Indeed, let R be any relation of arity, say r, in Γ, let $t_1, \ldots, t_n \in R$. We want to show that $g'(t_1, \ldots, t_n) \in R$ where $g'(t_1, \ldots, t_n)$ denotes the r-ary tuple obtained applying g' to $t_1, \ldots, t_n$ component-wise.

Let $p \in \Delta_n(A^r)$ be the distribution associated to multiset $p' = [t_1, \ldots, t_n]$ and consider constraint $C = (p_1, \ldots, p_r, R)$ on U_n where $p_1, \ldots, p_r$ are the marginals of p. By the choice of p we have $p(R) = 1$. Since ϕ is c-bounded it follows that $\Pr_{g\in\phi}\{g(p_1), \ldots, g(p_r)\} \notin R) \leq c \cdot \text{loss}(C) \leq 1 - p(R) = 0$. Hence, $(g(p_1), \ldots, g(p_r)) \in R$ for every g in the support of ϕ. It follows from Observation 2 that $g'(t_1, \ldots, t_n) = (g(p_1), \ldots, g(p_r))$ and we are done.

We have just seen that the support of the fractional n-ary operation, ϕ', associated to ϕ consists of polymorphisms of Γ. Since, by definition, the support of ϕ' only contains symmetric operations, in order to complete the proof it suffices to show that ϕ' is $(c \cdot |A|)$-stable.

Let $p'_1, p'_2 \in A^n$ and consider constraint $C = (p_1, p_2, \text{eq}_A)$ in U_n where $p_1, p_2 \in \Delta_n$ are the distributions associated to p'_1 and p'_2 respectively and eq_A is the equality relation on A. It is not too difficult to find a distribution p on A^2 with marginals p_1 and p_2 such that $1 - p(\text{eq}) \leq |A| \cdot \text{dist}(p'_1, p'_2)$. A concrete example can be obtained as follows. For very $a \in A$, let $a_1 = \max\{p_1(a) - p_2(a), 0\}$, and $a_2 = \max\{p_2(a) - p_1(a), 0\}$. Also, let $d = \sum_a a_1 = \sum_a a_2$. Then we define p as follows:

$$p(a,b) = \begin{cases} \min\{p_1(a), p_2(b)\} & \text{if } a = b \\ \frac{a_1 \cdot b_2}{d} & \text{if } a \neq b \end{cases}$$

It is easy to verify that p satisfies the desired conditions. Finally, we have

$$\Pr_{g'\sim\phi'}\{g'(p'_1) \neq g'(p'_2)\} = \Pr_{g\sim\phi}\{(g(p_1), g(p_2)) \notin \text{eq}_A\} \leq$$
$$\leq c \cdot \text{loss}(C) \leq c \cdot |A| \cdot \text{dist}(p'_1, p'_2).$$

We note that this is the only part where the condition $\text{eq}_A \in \Gamma$ is required.

(2) ⇒ (1). For every $n \in \mathbb{N}$, let n' be a multiple of n to be fixed later, let ϕ' be a c-stable fractional polymorphism of arity n' whose support consists of symmetric operations, and let ϕ be its associated fractional assignment for $U_{n'}$. We shall prove later that, for every constraint $C = (p_1, \ldots, p_r, R)$ in U_n (note, not in $U_{n'}$), we have

$$\Pr_{g\sim\phi}\{(g(p_1), \ldots, g(p_r)) \notin R\} \leq 2 \cdot r \cdot c \cdot \text{loss}(C) \tag{3.4}$$

Consider now the fractional assignment ϕ^* on U_n where for every assignment f on U_n, $\phi^*(f) = \sum_g \phi(g)$ where g ranges over all assignments for $U_{n'}$ that *extend* f (that is, such that $f(p) = g(p)$ for every $p \in \Delta_n$). It follows from the definition ϕ^* that

$$\Pr_{f\sim\phi^*}\{(f(p_1), \ldots, f(p_r)) \notin R\} = \Pr_{g\sim\phi}\{(g(p_1), \ldots, g(p_r)) \notin R\}$$

for every constraint $(p_1, \ldots, p_r, R)$ in U_n. This gives a way to construct, for every $n \in \mathbb{N}$, a $(2 \cdot K \cdot c)$-bounded fractional assignment for U_n where K is the maximum arity of a relation in Γ.

To finish the proof it only remains to prove inequality (3.4) for any constraint $C = (p_1, \ldots, p_r, R)$ in U_n. Let

p be a distribution on A^r such that $1 - p(R) = \text{loss}(C)$ is achieved. We can assume that $\text{loss}(C) \leq 1/2$ since otherwise there is nothing to prove.

Note that we can assume that $p(t)$ is rational for every $t \in A^r$. Let q be the distribution on A^r defined as

$$q(t) = \begin{cases} p(t)/p(R) & t \in R \\ 0 & t \notin R \end{cases}$$

Consider constraint $(q_1, \ldots, q_r, R)$ where $q_1, \ldots, q_r$ are the marginals of q. Since the number of constraints in U_n is finite we can assume that n' has been picked such that $q \in \Delta_{n'}(A^r)$. We claim that $(g(q_1), \ldots, g(q_r)) \in R$ for any g in the support of ϕ. Indeed, if $q' = [t_1, \ldots, t_{n'}]$ is the multiset of tuples in A^r associated to q then by Observation 2 $(g(q_1), \ldots, g(q_r)) = g'(t_1, \ldots, t_{n'})$ and the latter tuple belongs to R because g' is a polymorphism of Γ.

We claim that $\text{dist}(p'_i, q'_i) \leq 2 \cdot \text{loss}(C)$ for every $1 \leq i \leq r$. Indeed, it follows from the definition of q and the assumption that $\text{loss}(C) \leq 1/2$ that

$$q_i(a) \in \left[\frac{p_i(a) - \text{loss}(C)}{1 - \text{loss}(C)}, \frac{p_i(a)}{1 - \text{loss}(C)}\right] \subseteq$$
$$\subseteq [p_i(a) - \text{loss}(C), p_i(a) + 2 \cdot \text{loss}(C)]$$

for every $a \in A$. We conclude that

$$\Pr_{g \in \phi}\{(g(p_1), \ldots, g(p_r)) \notin R\} \leq$$
$$\leq \Pr_{g \in \phi}\{\exists i \text{ such that } g(p_i) \neq g(q_i)\} \leq 2 \cdot r \cdot c \cdot \text{loss}(C).$$

3.2 Algorithms Any sequence ϕ_n, $n \in \mathbb{N}$, satisfying condition (2) of Theorem 3.1 can be used to obtain a (possibly efficient) randomized rounding procedure for BLP, as follows. As we explained after Theorem 2.3, if one has an optimal rational solution to BLP(I), one can use a symmetric operation of appropriate arity n to round this solution to obtain a solution for I. If the symmetric operation is drawn from a c-stable distribution ϕ_n on n-ary symmetric polymorphisms (such as in Theorem 3.1) then this procedure can be shown to give a constant-factor approximation for Min CSP(Γ) (this follows from the proof of direction (2) $\Rightarrow$ (1) of Theorem 3.1). However it is not entirely clear how to efficiently sample from ϕ_n. We shall now give two examples of sequences of stable distributions that are nice enough to admit efficiently sampling. The first of these examples covers all problems Min CSP(Γ) that were previously known to belong to APX.

Two classes of CSPs were introduced and studied in [10], one is a subclass of the other. We need two notions to define these classes. A *distributive lattice* $(L, \cap, \cup)$ is a (lattice representable by a) family L of subsets of a set closed under intersection $\cap$ and union $\cup$. We say that two constraint languages $\Gamma_1 = \{R_1^{(1)}, \ldots, R_m^{(1)}\}$ on domain A and $\Gamma_2 = \{R_1^{(2)}, \ldots, R_m^{(2)}\}$ on domain B, where the arities of corresponding relations match, are *homomorphically equivalent* if there are two mappings $f : A \to B$, $g : B \to A$ such that for all $1 \leq i \leq m$, $f(t_1) \in R_i^{(2)}$ for every $t_1 \in R_i^{(1)}$ and $g(t_2) \in R_i^{(1)}$ for every $t_2 \in R_i^{(2)}$. The smaller class consists of constraint languages Γ such that Γ is homomorphically equivalent to a constraint language Γ' on some set L (of subsets) that has polymorphisms $\cap$ and $\cup$ where $(L, \cap, \cup)$ is a distributive lattice. The larger class is defined similarly, but we require Γ' to have polymorphism $x \cap (y \cup z)$. Constraint languages k-IHBS (defined in Section 1) belong to the larger, but not to the smaller class (see Example 1). See [10] for other specific examples of CSPs contained in these classes. For the smaller class, Min CSP(Γ) was shown to belong to APX in [32]. This result was extended to the larger class in [14] (see Theorems 5.8 and 4.8 there). This larger class is essentially the only class of constraint languages Γ such that Min CSP(Γ) is currently known to be in APX.

We will now show how stable distributions on symmetric polymorphisms can be used to provide a constant-factor approximation algorithm for Min CSP(Γ) for every Γ in this class. Observe that if Γ and Γ' are homomorphically equivalent then Min CSP(Γ) and Min CSP(Γ') are essentially the same problem because there is an obvious one-to-one correspondence between instances of Min CSP(Γ_1) and Min CSP(Γ_2) (swapping $R_i^{(1)}$ and $R_i^{(2)}$ in all constraints) and the maps f and g allow one to move between solutions to corresponding instances without any loss of quality. So, we can assume that A consists of subsets of some set, and Γ has polymorphism $x \cap (y \cup z)$ where $(A, \cap, \cup)$ is a distributive lattice.

Throughout the section, K will denote the maximum arity of a relation in such Γ. For every $1 \leq h \leq n$, let $g_{h,n}(x_1, \ldots, x_n)$ be the n-ary symmetric operation on A defined as

$$\bigcup_{I \subseteq \{1,\ldots,n\}, |I|=h} \left(\bigcap_{i \in I} x_i\right)$$

Lemma 3.4. *For all $h, n \in \mathbb{N}$ with $\left(1 - \frac{1}{|A|^K}\right)n < h \leq n$, we have $g_{h,n} \in \text{Pol}(\Gamma)$.*

Proof. It is not difficult to see that $x \cap y$ is also a polymorphism of Γ. Indeed, for every relation R and every pair of tuples $t, t' \in R$, we have that $t \cap t' = t \cap (t' \cup t')$ and hence it belongs to R. We say that operation $x \cap y$ is obtained from $x \cap (y \cup z)$ by *composition*. Proceeding in this way, we shall show that R has polymorphism $f_{h,n}$ where $f_{h,n}(x_0, x_1, \ldots, x_n)$ is the $(1+n)$-ary operation defined as

$$x_0 \cap g_{h,n}(x_1, \ldots, x_n) = x_0 \cap \left(\bigcup_{I \subseteq \{1,\ldots,n\}, |I|=h} \left(\bigcap_{i \in I} x_i\right)\right)$$

First, we observe that the m-ary operation $x_1 \cap \cdots \cap x_m$ preserves R as it can be obtained from composition from $x \cap y$ by $x_1 \cap (x_2 \cap (x_3 \cap \cdots \cap (x_{m-1} \cap x_m) \cdots))$. In a bit more complicated fashion we can show that $x_0 \cap (x_1 \cup \cdots \cup x_n)$ preserves R. If $n = 3$ it follows that $x_0 \cap ((x_0 \cap (x_1 \cup x_2)) \cup x_3)$ is equal to $x_0 \cap (x_1 \cup x_2 \cup x_3)$ (recall that $\cup$ and $\cap$ are the set union and intersection respectively). The pattern generalizes easily to arbitrary values for n. Finally, one obtains $f_{h,n}$ by suitably composing $x_0 \cap (x_1 \cup \cdots \cup x_n)$ and $x_1 \cap \cdots \cap x_m$.

Let R be a relation in Γ of arity, say, r and let $t_1, \ldots, t_n$ be a list of (not necessarily distinct) tuples in R. By the pigeon-hole principle, there exists a tuple t appearing at least $\lceil n/|A|^r \rceil$ times in $t_1, \ldots, t_n$. It follows from the choice of h and t, that for every $I \subseteq \{1, \ldots, n\}$, with $|I| = h$, there exists $i \in I$ such that $t = t_i$. It then follows that $f_{h,n}(t, t_1, \ldots, t_n)$, which necessarily belongs to R, is precisely $g_{h,n}(t_1, \ldots, t_n)$

For every $n \in \mathbb{N}$, consider the n-ary fractional operation ϕ_n with support $\left\{g_{h,n} \mid \left(1 - \frac{1}{|A|^K}\right) n < h \leq n\right\}$ that distributes uniformly among the operations of its support.

Lemma 3.5. *There exists some $c \geq 0$ such that ϕ_n is c-stable for every $n \in \mathbb{N}$.*

Proof. Let $\mathbf{a} = (a_1, \ldots, a_n), \mathbf{b} = (b_1, \ldots, b_n) \in A^n$. Recall that from distributivity we assume that every element $a \in A$ is a subset of some set that we call S. Note that , according to the definition of $g_{h,n}$, an element $j \in S$ belongs to $g_{h,n}(\mathbf{a})$ if $|\mathbf{a}|_j \geq h$ where $|\mathbf{a}|_j$ is defined to be $|\{1 \leq i \leq n \mid j \in a_i\}|$. Consequently, if $g_{h,n}(\mathbf{a}) \neq g_{h,n}(\mathbf{b})$ then there exists some $j \in S$ such that $|\mathbf{a}|_j \leq h < |\mathbf{b}|_j$ or $|\mathbf{b}|_j \leq h < |\mathbf{a}|_j$. It follows that

$$\Pr_{g \in \phi_n} \{g(\mathbf{a}) \neq g(\mathbf{b})\} \leq \frac{1}{n/|A|^K} \sum_{j \in S} ||\mathbf{a}|_j - |\mathbf{b}|_j| \leq$$
$$\leq |A|^K \cdot |S| \cdot \mathrm{dist}(\mathbf{a}, \mathbf{b}).$$

With the help of the sequence ϕ_n, we can obtain a constant-factor approximation algorithm for Min CSP(Γ). A different proof of this result was given in [14].

Theorem 3.2. *If a constraint language Γ has polymorphism $x \cap (y \cup z)$ where $(A, \cap, \cup)$ is a distributive lattice then* Min CSP(Γ) *has a constant-factor approximation algorithm.*

Proof. Let $I = (V, A, \mathscr{C})$ be any instance of Min CSP(Γ) and let $p_v (v \in V)$, $p_C (C \in \mathscr{C})$ be an optimal solution of BLP(I) with objective value $\mathsf{Opt}_{\mathsf{LP}}(I)$. We can assume that there exists some $n \in \mathbb{N}$ such that all the probabilities in the solution are of the form n'/n where n' is a non-negative integer. Also we can assume that $\log(n)$ is polynomial in the size of instance I.

Consider an assignment s for I obtained in the following way: draw $g_{h,n}$ according to ϕ_n and assign $s(v) = g_{h,n}(p'_v)$ where p'_v is any tuple such that every $a \in A$ appears exactly $p_v(a) \cdot n$ times in p'_v. It can be shown (this is basically the proof of direction ($2 \Rightarrow 1$) of Theorem 3.1) that there exists some $c' \geq 1$ such that expected value of assignment s is $c' \cdot \mathsf{Opt}_{\mathsf{LP}}(I)$. In particular, c' can be taken to be $2Kc$ where c is the stability constant of ϕ_n.

We shall prove that there is a randomized polynomial-time algorithm that constructs s. Recall that we assume that every element $a \in A$ is a subset of some set that we call S. Hence, in order to compute $g_{h,n}(p'_v)$, it is only necessary to give an efficient procedure that decides, for every $j \in S$, whether $j \in g_{h,n}(p'_v)$. Note that, according to the definition of $g_{h,n}$, $j \in g_{h,n}(p'_v)$ iff the number, $|p'_v|_j$, of entries in tuple p'_v that contain j is at least h. This number can be easily computed from p_v as $|p'_v|_j = n \cdot \sum_{\{a \in A | j \in a\}} p_v(a)$.

We finish this subsection by introducing another constraint language Γ such that Min CSP(Γ) admits a constant-factor approximation algorithm. The interest of this result is in the fact that it is the first known example of a constraint language where Min CSP(Γ) has a constant-factor approximation algorithm but is not invariant under totally symmetric polymorphisms of all arities (i.e. Γ does not have the so-called width 1 property [19]). This constraint language has domain $A = \{-1, 0, +1\}$ and contains relations $R_+ = \{(a_1, a_2, a_3) \in A^3 \mid a_1 + a_2 + a_3 \geq 1\}$ and $R_- = \{(a_1, a_2, a_3) \in A^3 \mid a_1 + a_2 + a_3 \leq -1\}$. This is the example of G. Kun [31] that we mentioned after Theorem 2.3. It is easy to show that this constraint language has no totally symmetric polymorphism of arity 3.

However $\{R_+, R_-\}$ have many symmetric polymorphisms. In particular, it is not difficult to see that, for all $h, n \in \mathbb{N}$ with $h < \lfloor n/3 \rfloor$, operation

$$s_{h,n}(x_1, \ldots, x_n) = \begin{cases} 1 & \text{if } h < \sum_i x_i \\ 0 & \text{if } -h \leq \sum_i x_i \leq h \\ -1 & \text{if } \sum_i x_i < -h \end{cases}$$

preserves Γ. It is also easy to show that the n-ary fractional operation with support $\{s_{h,n} \mid h < \lfloor n/3 \rfloor\}$ that distributes uniformly among the operations of its support is 3-stable and that can be efficiently sampled. Consequently, Min CSP($\{R_+, R_-\}$) has a constant-factor approximation algorithm.

3.3 NP-hardness result We will now show that, modulo P≠NP, if Min CSP(Γ) admits a constant-factor-approximation algorithm then Γ must have a near-unanimity (NU) polymorphism. This identity is well known in universal algebra [4] and its application in CSP [7, 11, 19]. For example, every relation invariant under an n-ary NU operation is uniquely determined by its $(n-1)$-ary projections [4], and NU polymorphisms characterize CSPs of "bounded strict width" [19].

We can assume (proved in Lemma 3.7 of [14]) that Γ contains all unary singleton relations $\{a\}$, $a \in A$. This implies that polymorphisms of Γ are idempotent. It can be easily derived from c-stability that then Γ must have a near-unanimity polymorphism of some (large enough) arity. Indeed, for any n-ary fractional operation ϕ_n with support on symmetric polymorphisms of Γ and every pair $a, b \in A$, the mass of operations g in the support of ϕ_n such that $g(b, a, \dots, a) \neq g(a, a, \dots, a)$ $(= a)$ is at most $\frac{c}{n}$. Since c is constant, if we choose n large enough, some g in the support of ϕ_n will satisfy the near-unanimity identity.

As an intermediate step, we consider the variant of CSP(Γ) where some constraints in an instance can be designated as *hard*, meaning that they must be satisfied in any feasible solution, while the other constraints are soft and can be falsified. It makes sense to investigate approximation algorithms for this mixed version of CSP (see, e.g. [21]). In particular, the value of a feasible assignment for a instance of mixed Min CSP(Γ) is defined to be the number (or total weight) of soft constraints it violates. It is not difficult to see, and was mentioned in [21] that mixed Min CSP(Γ) has a constant-factor approximation algorithm if and only if the ordinary, not mixed, Min CSP(Γ) has such an algorithm.

The proof of our NP-hardness result makes use of a hardness approximation for the problem Max IS_k in which the goal is to find a maximum independent set in a given k-regular hypergraph. Recall that an independent set in a hypergraph is a subset of its vertices that does not include any of its hyperedges (entirely). For real numbers $0 \leq \alpha, \beta \leq 1$, say that an algorithm (α, β)-*distinguishes* Max IS_k if, given a k-regular hypergraph $H = (V, E)$, it correctly decides between the following two cases:

1. the size of the largest independent set of H is at least $\beta \cdot |V|$
2. the size of the largest independent set of H is at most $\alpha \cdot |V|$.

Note that it does not matter what the algorithm does for a hypergraph falling into neither of these cases.

Theorem 3.3. ([17]) *For any integer $k \geq 3$ and any real number $\epsilon > 0$, it is* NP*-hard to $(\epsilon, 1 - \frac{1}{k-1} - \epsilon)$-distinguish* Max IS_k.

Theorem 3.4. *Let Γ be a constraint language containing all unary singleton relations. If* Min CSP(Γ) *admits a constant-factor approximation algorithm then Γ has an NU polymorphism, unless* P = NP.

The key in proof is to show that, roughly, if Γ has no NU polymorphisms then Γ can simulate (pp-define, to be precise), for every $k \geq 3$, a k-ary relation R_k such that $R_k \cap \{a, b\}^k = \{a, b\}^k \setminus \{(a, \dots, a)\}$ for some distinct $a, b \in A$. This relation, used in hard constraints, can encode a k-uniform hypergraph, while soft unary constraints using relation $\{a\}$ simulate a choice of an independent set. To make this precise we will need a few definitions.

We say that R is pp-definable from Γ if there exists a (primitive positive) formula

$$\phi(x_1, \dots, x_k) \equiv \exists y_1, \dots, y_l \, \psi(x_1, \dots, x_k, y_1, \dots, y_l)$$

where ψ is a conjunction of atomic formulas with relations in Γ and eq_A such that for every $(a_1, \dots, a_k) \in A^k$

$$(a_1, \dots, a_k) \in R \text{ if and only if } \phi(a_1, \dots, a_k) \text{ holds.}$$

Note that in the definition of primitive positive formulas we are slightly abusing notation by identifying a relation with its relation symbol. It is shown in [14] that if Γ contains eq_A and R is pp-definable from Γ then the problems Min CSP(Γ) and Min CSP($\Gamma \cup \{R\}$) simultaneously belong or do not belong to APX.

An n-ary operation on A is called a *weak near-unanimity (WNU)* operation if it is idempotent and satisfies the identities

$$f(y, x, \dots, x) = f(x, y, \dots, x) = \dots = f(x, x, \dots, y).$$

Proof. (of Theorem 3.4) Assume, towards a contradiction, that Γ falsifies the statement of the Theorem.

The following lemma can be derived from a combination of several known results. We give a (more or less) direct proof for completeness.

Lemma 3.6. *For every $k \geq 1$, there is a k-ary relation, R, pp-definable from Γ, and $a, b \in A$ such that*

$$R \cap \{a, b\}^k = \{a, b\}^k \setminus \{(a, \dots, a)\}$$

Proof. It follows easily from [4] that if Pol(Γ) does not contain any NU operation, then for every $n \geq 3$ there is a relation $T \subseteq A^n$ pp-definable from Γ and a tuple $(a_1, \dots, a_n) \notin T$ such that for every $1 \leq i \leq n$ there exists $c_i \in A$ such that $(a_1, \dots, a_{i-1}, c_i, a_{i+1}, \dots, a_n) \in T$. Setting $n \geq (k+2)|A|^2$ it follows from the pigeonhole principle that there exists $a, c \in A$ and $I = \{i_1, \dots, i_{k+2}\} \subseteq \{1, \dots, n\}$ of size $k+2$ such that $a_i = a$ and $c_i = c$ for every $i \in I$. Consider relation S defined as

$$S = \{(x_{i_1}, \dots, x_{i_{k+2}}) \mid (x_1, \dots, x_n) \in T, \forall i \notin I (x_i = a_i)\}$$

Clearly, S is pp-definable using T and the unary singletons. It follows that S is pp-definable from Γ as well. We have that $(a, a, \dots, a) \notin S, t_1 = (c, a, \dots, a) \in S$, $t_2 = (a, c, \dots, a) \in S$, …, and $t_{k+2} = (a, a, \dots, c) \in S$. We can also assume that, in addition to the previous property, S is symmetric, meaning that if $(x_1, \dots, x_{k+2})$ belongs to S then so does any tuple obtained by permuting its entries. This is because we can always replace

S by the relation $\{(x_1, \ldots, x_{k+2}) \mid (x_{\sigma(1)}, \ldots, x_{\sigma(k+2)}) \in S$ for every permutation $\sigma\}$ which is pp-definable from S. Since, by assumption, Min CSP(Γ) admits a constant-factor approximation algorithm it follows from Theorem 9 of [14] that Γ has a certain property, called *bounded width* (or else P = NP). Theorem 2.8 in [29] states that this property implies that Pol(Γ) contains WNU polymorphisms g_3, g_4 of arity 3 and 4, respectively, such that $g_3(y, x, x) = g_4(y, x, x, x)$ holds for for every $x, y \in A$. The proof of Theorem 2.8 in [29] shows how to obtain g_n for $n = 3, 4$, but the proof generalizes immediately to show that, for each $n \geq 3$, Γ has an n-ary WNU polymorphism g_n, of arity n, and the identity $g_n(y, x, \ldots, x) = g_{n'}(y, x, \ldots, x)$ holds for all n, n'.

Let $b = g_n(c, a, \ldots, a)$ and let j be minimum with the property that S contains every tuple $t \in \{a, b\}^{k+2}$ with at least j b's. We claim that $1 \leq j \leq 3$. The lower bound follows from the fact that $(a, \ldots, a) \notin S$. For the upper bound, it follows from the fact every g_n is a WNU (and so idempotent), that every tuple $t \in \{a, b\}^{k+2}$ with $j(\geq 3)$ b's can be obtained by applying g_j component-wise to tuples $t_{i_1}, \ldots, t_{i_j}$ where $i_1, \ldots, i_j$ are the components in t that contain a b. Since S is symmetric then it does not contain any tuple in $\{a, b\}^{k+2}$ with less than j b's.

Finally, consider relation R defined as

$$R = \{(x_1, \ldots, x_k) \mid (\underbrace{b, \ldots, b}_{j-1}, \underbrace{a, \ldots, a}_{3-j}, x_1 \ldots, x_k) \in S\}$$

By a similar reasoning than before we infer that R is pp-definable from Γ. It follows from the definition that R, a and b satisfy the statement of the lemma.

Lemma 3.7. *For every $k \geq 1$, there is a linear algorithm that, for a given k-regular hypergraph $H = (V, E)$, returns an instance I of mixed* Min CSP(Γ) *such that the value of optimal solution for I is $1 - m/|V|$ where m is the set of the maximum independent set in H.*

Proof. Fix $k \geq 1$ and let R and a, b as in Lemma 3.6. Let $\exists y_1, \ldots, y_l\ \psi(x_1, \ldots, x_k, y_1, \ldots, y_l)$ be a primitive positive formula defining R from Γ. It is well known that ψ can be seen as an instance J of CSP(Γ). More precisely, define J to be the instance that has variables $x_1, \ldots, x_k, y_1, \ldots, y_l$ and contains for every atomic formula $S(v_1, \ldots, v_r)$ in ψ, the constraint $((v_1, \ldots, v_r), S)$. It follows that for any assignment $s : \{x_1, \ldots, x_k, y_1, \ldots, y_l\} \to A$, s is a solution of J if and only if $\psi(s(x_1), \ldots, s(x_k), s(y_1), \ldots, s(y_l))$ holds.

Consider the algorithm that, given a k-regular hypergraph, $H = (V, E)$, constructs an instance I of mixed Min CSP(Γ) as follows. The set of variables of I contains, in addition to all nodes in V, some other fresh variables to be introduced later. Then, for every hyperedge $E = \{v_1, \ldots, v_k\}$, add a copy of J where the variables have been renamed so that $x_1 = v_1, \ldots, x_k = v_k$ and $y_1, \ldots, y_n$ are different fresh variables (different for each hyperedge). All the constraints added so far are designated as hard. Finally, add for every $v \in V$ a soft constraint $(v, \{a\})$ requiring v to take value a.

Note that as k is fixed, this can be carried out in linear time. It follows from the construction of I that for every independent set, X, of H there is an assignment for I satisfying all hard constraints that maps every node in X to a and every node in $V \setminus X$ to b. This assignment violates exactly $|V| - |X|$ soft constraints. Conversely, for every assignment s in I, the set $X = \{v \in V \mid s(v) = a\}$ is an independent set of H.

We are finally in a position to obtain a contradiction. As discussed above, if Min CSP(Γ) admits a constant-factor approximation algorithm then so does its mixed variant. Let δ satisfy $0 < \delta/2 \leq 1 - c \cdot \delta$ where c is the approximation factor for the mixed Min CSP(Γ) algorithm. Lemma 3.7 immediately gives, for every k, a polynomial time algorithm that $(1 - c \cdot \delta, 1 - \delta)$-distinguishes Max IS_k but this task is NP-hard, as follows by setting $\epsilon = \delta/2$ and $k = 1 + 2/\delta$ in Theorem 3.3.

References

[1] A. Agarwal, M. Charikar, K. Makarychev, and Y. Makarychev. $O(\sqrt{log\ n})$ approximation algorithms for MinUnCut, Min 2CNF Deletion, and Directed Cut problems. In *STOC*, pages 573–581, 2005.

[2] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

[3] P. Austrin. *Conditional Inapproximability and Limited Independence*. PhD thesis, KTH, 2008.

[4] K.A. Baker and A.F. Pixley. Polynomial interpolation and the chinese remainder theorem. *Math. Z.*, 143:165–174, 1975.

[5] L. Barto and M. Kozik. Constraint satisfaction problems of bounded width. In *FOCS*, pages 595–603, 2009.

[6] L. Barto and M. Kozik. Robust satisfiability of constraint satisfaction problems. In *STOC*, pages 19–22, 2012.

[7] L. Barto, M. Kozik, and R. Willard. Near unanimity constraints have bounded pathwidth duality. In *LICS*, pages 125–134, 2012.

[8] A. Bulatov. A dichotomy theorem for constraint satisfaction problems on a 3-element set. *J. ACM*, 53(1):66–120, 2006.

[9] A. Bulatov, P. Jeavons, and A. Krokhin. Classifying complexity of constraints using finite algebras. *SIAM J. Comput.*, 34(3):720–742, 2005.

[10] C. Carvalho, V. Dalmau, and A. Krokhin. Two new homomorphism dualities and lattice operations. *J. Logic Comput.*, 21(6):1065–1092, 2011.

[11] D. Cohen and P. Jeavons. The complexity of constraint languages. In F. Rossi, P. van Beek, and T. Walsh, editors, *Handbook of Constraint Programming*, chapter 8. Elsevier, 2006.

[12] N. Creignou, S. Khanna, and M. Sudan. *Complexity Classifications of Boolean Constraint Satisfaction Problems*, volume 7 of *SIAM Monographs on Discrete Mathematics and Applications*. 2001.

[13] N. Creignou, Ph.G. Kolaitis, and H. Vollmer, editors. *Complexity of Constraints*, volume 5250 of *LNCS*. Springer, 2008.

[14] V. Dalmau and A. Krokhin. Robust satisfiability for CSPs: Hardness and algorithmic results. *ACM TOCT*, 5(4):Article 15, 2013.

[15] V. Deineko, P. Jonsson, M. Klasson, and A. Krokhin. The approximability of Max CSP with fixed-value constraints. *J. ACM*, 55(4):Article 16, 2008.

[16] I. Dinur. The PCP theorem by gap amplification. *J. ACM*, 54(3), 2007.

[17] I. Dinur, V. Guruswami, S. Khot, and O. Regev. A new multilayered PCP and the hardness of hypergraph vertex cover. *SIAM J. Comput.*, 34(5):1129–1146, 2005.

[18] A. Ene, J. Vondrak, and Y. Wu. Local distribution and the symmetry gap: Approximability of the multiway partitioning problems. In *SODA*, pages 306–325, 2013.

[19] T. Feder and M. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through Datalog and group theory. *SIAM Journal on Computing*, 28:57–104, 1998.

[20] G. Gottlob, G. Greco, and F. Scarcello. Tractable optimization problems through hypergraph-based structural restrictions. In *ICALP*, pages 16–30, 2009.

[21] V. Guruswami and E. Lee. Complexity of approximating CSP with balance/hard constraints. In *ITCS*, pages 439–448, 2014.

[22] V. Guruswami and Y. Zhou. Tight bounds on the approximability of Almost-satisfiable Horn SAT and Exact Hitting Set. *Theory of Computing*, 8:Article 11, 2012.

[23] J. Håstad. Some optimal inapproximability results. *J. ACM.*, 48:798–859, 2001.

[24] J. Håstad. Every 2-CSP allows nontrivial approximation. *Comput. Complexity*, 17(4):549–566, 2008.

[25] S. Khot. On the unique games conjecture. In *CCC*, pages 99–121, 2010.

[26] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for Max-Cut and other 2-variable CSPs? *SIAM J. Comput.*, 37(1):319–357, 2007.

[27] S. Khot and O. Regev. Vertex cover might be hard to approximate to within 2-ϵ. *J. Comput. System Sci.*, 74(3):335 – 349, 2008.

[28] S. Khot, M. Tulsiani, and P. Worah. A characterization of strong approximation resistance. In *STOC*, pages 634–643, 2014.

[29] M. Kozik, A. Krokhin, M. Valeriote, and R. Willard. Characterizations of several Maltsev conditions. *Algebra Univ.*, 2014. to appear.

[30] A. Kumar, R. Manokaran, M. Tulsiani, and N. Vishnoi. On LP-based approximability for strict CSPs. In *SODA*, pages 1560–1573, 2011.

[31] G. Kun. Personal communication.

[32] G. Kun, R. O'Donnell, T. Suguru, Y. Yoshida, and Y. Zhou. Linear programming, width-1 CSPs, and robust satisfaction. In *ITCS*, pages 484–495, 2012.

[33] R. Manokaran. *Approximability and mathematical relaxations*. PhD thesis, Princeton University, 2012.

[34] R. Manokaran, J. Naor, P. Raghavendra, and R. Schwartz. SDP gaps and UGC hardness for Multiway Cut, 0-Extension, and Metric Labeling. In *STOC*, pages 11–20, 2008.

[35] D. Marx. Tractable hypergraph properties for constraint satisfaction and conjunctive queries. *J. ACM*, 60(6):Article 42, 2013.

[36] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: Invariance and optimality. In *FOCS*, pages 21–30, 2005.

[37] P. Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *STOC*, pages 245–254, 2008.

[38] T.J. Schaefer. The complexity of satisfiability problems. In *STOC*, pages 216–226, 1978.

[39] A. Schrijver. *Theory of Linear and Integer Programming*. Wiley, 1986.

[40] J. Thapper and S. Živný. The power of linear programming for valued CSPs. In *FOCS*, pages 669–678, 2012.

[41] J. Vondrak. Personal communication.