ARTICLE IN PRESS YJCSS:3167

Journal of Computer and System Sciences ••• (••••) •••–•••

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

Towards a characterization of constant-factor approximable finite-valued CSPs [☆,☆☆]

Víctor Dalmau [a,*], Andrei Krokhin [b], Rajsekar Manokaran [c]

[a] Department of Information and Communication Technologies, Universitat Pompeu Fabra, Spain
[b] Department of Computer Science, Durham University, UK
[c] Computer Science and Engineering Department, IIT Madras, India

ARTICLE INFO

Article history:
Received 23 September 2016
Received in revised form 17 January 2018
Accepted 15 March 2018
Available online xxxx

Keywords:
Constraint satisfaction problem
Approximation algorithms
Universal algebra

ABSTRACT

We study the approximability of (Finite-)Valued Constraint Satisfaction Problems (VCSPs) with a fixed finite constraint language Γ consisting of finitary functions on a fixed finite domain. Ene et al. have shown that, under a mild technical condition, the basic LP relaxation is optimal for constant-factor approximation for VCSP(Γ) unless the Unique Games Conjecture fails. Using the algebraic approach to the CSP, we give new natural algebraic conditions for the finiteness of the integrality gap for the basic LP relaxation of VCSP(Γ) and show how this leads to efficient constant-factor approximation algorithms for several examples that cover all previously known cases that are NP-hard to solve to optimality but admit constant-factor approximation. Finally, we show that the absence of another algebraic condition leads to NP-hardness of constant-factor approximation. Thus, our results indicate where the boundary of constant-factor approximability for VCSPs lies.

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 [1–3]. Standard examples of CSPs include satisfiability of propositional formulas, graph coloring problems, and systems of linear equations. An instance of the CSP consists of a set of variables, a (not necessarily Boolean) domain of labels, 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 labels 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. [1–3]), 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) relation specifying the allowed combinations of labels. 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 [4,5], and (b) *language-based* where the con-

☆ This article is an extended version of a paper published in the proceedings of SODA'15.
☆☆ The first two authors were supported by UK EPSRC grant EP/J000078/01. The first author was also supported by MICCIN grant TIN2016-76573-C2-1P and Maria de Maeztu Units of Excellence Programme MDM-2015-0502. The third author was supported by ERC grant 226-203.
* Corresponding author.
E-mail addresses: victor.dalmau@upf.edu (V. Dalmau), Andrei.Krokhin@durham.ac.uk (A. Krokhin), rajsekar@gmail.com (R. Manokaran).

https://doi.org/10.1016/j.jcss.2018.03.003

Please cite this article in press as: V. Dalmau et al., Towards a characterization of constant-factor approximable finite-valued CSPs, J. Comput. Syst. Sci. (2018), https://doi.org/10.1016/j.jcss.2018.03.003

straint language Γ, i.e. the set of relations that can appear in constraints, is fixed (see, e.g. [6,1,7,3]); 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 [8], which completely characterize 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. [9–15], but many questions are still open.

Problems Max CSP and Min CSP can be generalized by replacing relations (that specify allowed combinations of labels) with functions that specify a value in $[0, 1]$ (measuring the desirability or the cost, respectively) for each tuple of labels. The goal would then be to find an assignment of labels that maximizes the total desirability (minimizes the total cost, respectively). The maximization version was studied in [16,17] under the name of Generalized CSP, or GCSP, (in fact, functions there can take values in $[-1, 1]$), while the minimization version is known as (Finite-)Valued CSP [14]. In General-Valued CSP, functions can also take the infinite value to indicate infeasible tuples [18,19,13], but we will not consider this case in this paper. In this paper we write VCSP to mean *finite-valued* CSP. We note that [20] write Min CSP to mean what we call VCSP in this paper. Naturally, both GCSP and VCSP can be parameterized by constraint languages Γ, now consisting of functions instead of relations.

The CSP has always played an important role in mapping the landscape of approximability of NP-hard optimization problems, see e.g. surveys [21,22]. For example, the famous PCP theorem has an equivalent reformulation in terms of inapproximability of a certain Max CSP(Γ), see [23]; moreover, Dinur's combinatorial proof of this theorem [24] deals entirely with CSPs. The first optimal inapproximability results [25] 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. [26–28]). The approximability of Boolean CSPs has been thoroughly investigated (see, e.g. [29,1,30,31,25,27,21,32]). Much work around the Unique Games Conjecture (UGC) directly concerns CSPs [21]. 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 $\mathrm{CSP}(\Gamma_k)$, 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 [21,32]. Raghavendra proved [17] that one SDP-based algorithm provides optimal approximation for all problems GCSP(Γ) assuming the UGC. In this paper, we investigate problems VCSP(Γ) and Min CSP(Γ) on 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. The same also holds for GCSP. Clearly, for Min CSP(Γ) to admit a constant-factor approximation algorithm, CSP(Γ) must be polynomial-time solvable.

The approximability of problems VCSP(Γ) has been studied, mostly for Min CSPs in the Boolean case (i.e., with domain $\{0, 1\}$, such CSPs are sometimes called "generalized satisfiability" problems), see [29,1]. We need a few concepts from propositional logic. A clause is *Horn* if it contains at most one positive literal, and *negative* if 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 [1], and they (and the corresponding dual Horn problems) are essentially the only such Boolean Min CSPs unless the UGC fails [33]. For Min CSP(2-HORN), which is identical to Min CSP(2-IHBS), a 2-approximation (LP-based) algorithm is described in [31], which is optimal assuming the UGC, whereas it is NP-hard to constant-factor approximate Min CSP(3-HORN) [30]. If $\neq_2$ is the Boolean relation $\{(0, 1), (1, 0)\}$, then 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 $O(\sqrt{\log n})$ [29], and it follows from [32] that neither problem belongs to APX unless the UGC is false. The UGC is known to imply the optimality of the basic LP relaxation for any VCSP(Γ) such that Γ contains the (characteristic function of the) equality relation [20], extending the line of similar results for natural LP and SDP relaxations for various optimization CSPs [34,35,17].

An approximation algorithm for any VCSP(Γ) was also given in the 2013 conference version of [20] (that was claimed to match the LP integrality gap), but its analysis was later found to be faulty and this part was retracted in the 2015 update of [20]. The SDP rounding algorithm for GCSPs from [36] is discussed in detail in the book [37], where it is pointed out that the same algorithm does not work for VCSPs.

Constant-factor approximation algorithms for Min CSP are closely related to certain *robust algorithms* for CSP that attracted much attention recently [10,33,38,39]. 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 $f(0) = 0$. CSPs admitting a robust algorithm (with some function f) were completely characterized in [10]; 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 posed in [33]. It is easy to see that, for any Γ, CSP(Γ) admits a robust algorithm with