Theoretical Computer Science 715 (2018) 21–34

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Tractability conditions for numeric CSPs ☆

Peter Jonsson [a], Johan Thapper [b,*]

[a] *Department of Computer and Information Science, Linköpings universitet, SE-58183 Linköping, Sweden*
[b] *Laboratoire d'Informatique Gaspard-Monge, Université Paris-Est Marne-la-Vallée, 5 boulevard Descartes, F-77420 Champs-sur-Marne, France*

ARTICLE INFO

Article history:
Received 6 April 2017
Received in revised form 15 October 2017
Accepted 15 January 2018
Available online 1 February 2018
Communicated by L.M. Kirousis

Keywords:
Constraint satisfaction problem
Semialgebraic constraint
o-minimality
Algebraic variety
Computational complexity

ABSTRACT

The computational complexity of the constraint satisfaction problem (CSP) with semilinear relations over the reals has gained recent attraction. As a result, its complexity is known for all finite sets of semilinear relations containing the relation $R_+ = \{(x, y, z) \in \mathbb{R}^3 \mid x + y = z\}$. We consider larger and more expressive classes of relations such as *semialgebraic* and *o-minimal* relations. We present a general result for characterising computationally hard fragments and, under certain side conditions, this result implies that polynomial-time solvable fragments are only to be found within two limited families of sets of relations. In the setting of semialgebraic relation, our result takes on a simplified form and we provide a full complexity classification for constraint languages that consist of algebraic varieties. Full classifications like the one obtained here for algebraic varieties or the one for semilinear relations appear to be rare and we discuss several barriers for obtaining further such results. These barriers have strong connections with well-known open problems concerning the complexity of various restrictions of convex programming.

1. Introduction

1.1. The constraint satisfaction problem

The *constraint satisfaction problem* (CSP) is an important computational problem in many areas of computer science and mathematics. In this problem, we are given a set of variables that take their values from a (finite or infinite) *domain*. The assignments to the variables are further subjected to a set of *constraints*. A constraint is defined by requiring that a tuple of variables belongs to some specified relation. The question is whether the variables can be assigned values such that all constraints are satisfied. Since even the general finite-domain CSP is NP-hard, the complexity of CSPs is often studied by introducing an additional parameter, a set Γ of allowed relations, known as a *constraint language* (or *template*). This leads to a problem CSP(Γ) where the relations of all constraints in the input are required to come from Γ. This way of parameterizing constraint satisfaction problems has proved to be very fruitful for both finite and infinite domains. In the sequel, when we talk about a CSP, we will mean a problem CSP(Γ) for some fixed Γ.

The complexity of finite-domain CSPs has been extensively investigated, beginning with Schaefer [31]. This has lead to the development of a set of standard tools, including the powerful universal-algebraic approach [12]. Much of this effort has

☆ The first author was partially supported by the Swedish Research Council (VR) under grant 621-2012-3239.
* Corresponding author.
E-mail addresses: peter.jonsson@liu.se (P. Jonsson), thapper@u-pem.fr (J. Thapper).

https://doi.org/10.1016/j.tcs.2018.01.013
0304-3975/

been devoted to the *Feder–Vardi Dichotomy Conjecture* [14] which posits that every finite-domain CSP is either polynomial-time solvable or NP-complete. This conjecture has recently been resolved in the positive in two independent works [11, 35].

Infinite-domain CSPs, on the other hand, constitute a much more diverse set of problems: *Every* computational problem is polynomial-time equivalent to some infinite-domain CSP [4]. Obtaining a full understanding of their computational complexity is thus too ambitious. Instead, restricted classes of problems are studied. For example, one can consider CSPs over numeric domains, such as the reals, the rationals, or the integers, with constraints derived from arithmetic operations and the natural order on these domains. We refer the reader to [10] for a recent survey of such CSPs.

In this article, we study CSPs over the reals with constraints based on *o-minimal* and in particular *semialgebraic* relations. The point of departure of our investigation is recent progress on the project of classifying *semilinear* CSPs. We describe this next.

1.2. Semilinear CSPs

A relation is semilinear if it can be written as a finite union of finite intersections of open and closed half-spaces over, for instance, the reals, the rationals or the integers. Let $SL_X[Y]$ denote the set of semilinear relations with domain X and coefficients in Y. We will mainly consider $SL_{\mathbb{R}}[\mathbb{Q}]$; this set of relations equals the set of first-order definable relations over $\{+, \leq, \{1\}\}$ [15].

Characterising the polynomial-time solvable cases of $SL_{\mathbb{R}}[\mathbb{Q}]$ is a challenging task. The construction presented in [21, Section 6.3] proves the following: for every finite constraint language Γ over a finite domain, there exists a finite $\Gamma' \subseteq SL_{\mathbb{R}}[\mathbb{Q}]$ such that CSP(Γ) and CSP(Γ') are polynomial-time equivalent problems. Hence, the classification task of semilinear CSPs is inextricably linked to the classification task of finite-domain CSPs, and thereby to the Feder–Vardi dichotomy conjecture. We also observe that the complexity of every finite *temporal* constraint language would be determined as a by-product of a full classification of $SL_{\mathbb{R}}[\mathbb{Q}]$. A temporal constraint language is a constraint language that is first-order definable in $(\mathbb{Q}; <)$. The complexity of temporal constraint languages has been fully determined [7] and the polynomial-time solvable cases fall into nine different categories. The proof is complex and makes heavy use of the universal-algebraic approach.

One way of obtaining classes of semilinear constraint languages that are more manageable is to restrict attention to expansions of certain natural sets of relations. One such choice is the set $\Gamma_{lin} = \{R_+, \leq, \{1\}\}$, where $R_+ = \{(x, y, z) \in \mathbb{R}^3 \mid x + y = z\}$. Studying the computational complexity of expansions of Γ_{lin} is well motivated by the fact that CSP(Γ_{lin}) is polynomial-time many-one equivalent to the linear programming feasibility problem [5]. This direction has been pursued in a number of recent works [5,6,21,22]. A complete classification for semilinear expansions of Γ_{lin} was obtained in [5]. This was generalised to semilinear expansions of $\{R_+\}$ by Jonsson and Thapper [23]. Here, we describe an intermediate classification for semilinear expansions of $\{R_+, \{1\}\}$ [22].

A relation is *primitive positive* (pp) definable from a constraint language Γ if it can be expressed using existential quantification over conjunctions of atoms. The importance of pp-definability is explained by Lemma 4 below.

We say that a relation $R \subseteq \mathbb{R}^k$ is *essentially convex* if for all $p, q \in R$ there are only finitely many points on the line segment between p and q that are not in R. A BNU (for *bounded*, *non-constant*, and *unary*) is a bounded unary relation that contains more than one point.

Theorem 1 *(Jonsson and Thapper [22]). Let $\{R_+, \{1\}\} \subseteq \Gamma \subseteq SL_{\mathbb{R}}[\mathbb{Q}]$ be a finite constraint language. If*

1. *Γ contains a relation that is not essentially convex, and*
2. *Γ can primitive positively define a BNU relation,*

then CSP(Γ) is NP-hard. Otherwise, CSP(Γ) is tractable.

Our goal will be to extend Theorem 1 as far as possible to *semialgebraic* and *o-minimal* constraint languages. We say that a relation $R \subseteq \mathbb{R}^k$ is *semialgebraic* if it can be first-order defined in $\{+, \cdot, \leq\}$ with parameters in $\mathbb{R}$. The classical Tarski–Seidenberg theorem [33] implies that semialgebraic constraint languages have a clear geometric interpretation: every semialgebraic relation can be written as a finite union of solution sets of strict and non-strict polynomial inequalities. Semialgebraic relations appear in many different contexts within mathematics and computer science (cf. the textbook by Basu, Pollack, and Roy [3]).

A constraint language Γ that contains a total ordering of the domain $\mathbb{R}$ is called *o-minimal* if every first-order definable set (with parameters from $\mathbb{R}$) can be represented as a union of finitely many intervals and points. Structures that are o-minimal have been studied thoroughly in model theory (cf. van den Dries [34] and Macpherson [26]). A well-known class of relations that give rise to o-minimal but not semialgebraic constraint languages is the set of *Semi-Pfaffian* relations, cf. Khovanskii [24]. A concrete example of a semi-Pfaffian constraint language that is not semialgebraic is $\mathrm{SA}_{\mathbb{R}}[\mathbb{R}] \cup \{(x, y) \in \mathbb{R}^2 \mid y = e^x\}$.