Artificial Intelligence 229 (2015) 105–125

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Variable symmetry breaking in numerical constraint problems ☆

Alexandre Goldsztejn [a], Christophe Jermann [b,*], Vicente Ruiz de Angulo [c], Carme Torras [c]

[a] IRCCyN, CNRS, Nantes, France
[b] LINA, Université de Nantes/CNRS, Nantes, France
[c] Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Barcelona, Spain

ARTICLE INFO

Article history:
Received 10 December 2014
Received in revised form 6 August 2015
Accepted 16 August 2015
Available online 20 August 2015

Keywords:
Constraint programming
Symmetries
Numerical constraints
Variable symmetries

ABSTRACT

Symmetry breaking has been a hot topic of research in the past years, leading to many theoretical developments as well as strong scaling strategies for dealing with hard applications. Most of the research has however focused on discrete, combinatorial, problems, and only few considered also continuous, numerical, problems. While part of the theory applies in both contexts, numerical problems have specificities that make most of the technical developments inadequate.

In this paper, we present the RLEX constraints, partial symmetry-breaking inequalities corresponding to a relaxation of the famous LEX constraints extensively studied in the discrete case. They allow (partially) breaking any variable symmetry and can be generated in polynomial time. Contrarily to LEX constraints that are impractical in general (due to their overwhelming number) and inappropriate in the continuous context (due to their form), RLEX constraints can be efficiently handled natively by numerical constraint solvers. Moreover, we demonstrate their pruning power on continuous domains is almost as strong as that of LEX constraints, and they subsume several previous work on breaking specific symmetry classes for continuous problems. Their experimental behavior is assessed on a collection of standard numerical problems and the factors influencing their impact are studied. The results confirm RLEX constraints are a dependable counterpart to LEX constraints for numerical problems.

1. Introduction

Numerical constraint solvers are nowadays beginning to be competitive and even to outperform, in some cases, classical methods for solving systems of equations and inequalities over the reals. As a consequence, their application has raised interest in fields as diverse as neurophysiology and economics [2], biochemistry, crystallography, robotics [3] and, more generally, in those related to global optimization [4]. Symmetries naturally occur in many of these applications, and it is advisable to exploit them in order to reduce the search space and, thus, to increase the efficiency of the solvers.

☆ This paper is an extended version of [1] presented at the conference CP 2011.

* Corresponding author.
E-mail addresses: alexandre.goldsztejn@irccyn.ec-nantes.fr (A. Goldsztejn), christophe.jermann@univ-nantes.fr (C. Jermann), ruiz@iri.upc.edu (V. Ruiz de Angulo), torras@iri.upc.edu (C. Torras).

http://dx.doi.org/10.1016/j.artint.2015.08.006

Numerical solvers follow the Branch&Prune scheme, similarly to discrete constraint solvers: At each iteration, a sub-domain is selected, pruned according to the constraints, and then split into several sub-domains to be further explored. The main differences with discrete solvers are that (sub)domains being continuous only their boundaries are contracted, and that a sub-domain is declared to be a solution whenever it reaches some prescribed computational precision. Because of the resemblance of the solving processes, it is tempting to port symmetry breaking methods designed for discrete Constraint Satisfaction Problems (CSPs) to numerical ones.

Considerable work on symmetry breaking has been done for discrete CSPs in the last decades [5–7]. Two main symmetry-breaking strategies have been pursued: 1) to devise specialized search algorithms that avoid symmetric portions of the search space [8,9]; and 2) to add *symmetry-breaking constraints* (SBCs) that filter out redundant subspaces [10,11]. Contrarily, there exists very little work on symmetry breaking for numerical problems. For cyclic variables permutations, an approach divides the initial space into boxes and eliminates symmetric ones before the solving starts [12]. SBCs have also been proposed, but only for either specific problems [13,14] or specific symmetry classes [15–17], and often only partially breaking the considered symmetries.

In this paper, we propose the first general SBCs for numerical constraint problems that (partially) break any variable symmetry. These SBCs take the form of simple binary inequalities of the form $x_i \le x_j$ where x_i and x_j are two distinct variables of the problem and $i < j$. Thus, at most $\frac{n(n-1)}{2}$ inequalities are generated to deal with any symmetry. They can be generated in polynomial time knowing a generator of the symmetry group, using classical group theory algorithms. Moreover we demonstrate that these SBCs are suitable and optimal for numerical problems, i.e., they enclose tightly an asymmetric search subspace and thus have a better, or similar, pruning power than other SBCs.

The outline of the paper is as follows: Section 2 provides the necessary background on numerical problems and symmetries; Section 3 introduces our SBCs as a relaxation of the lexicographic-ordering based SBCs [18] widely used by the discrete CSP community, and it also establishes the good properties of this relaxation; Section 4 introduces the state-of-the-art and compares our SBCs to existing alternatives; Section 5 assesses the practical interest of our SBCs on a benchmark of standard problems and analyzes the factors influencing their impact. Section 6 concludes the paper with future research directions.

2. Principles of variable symmetry breaking

A CSP is defined as a triple $\langle x, d, c\rangle$, where $x = (x_1, \dots, x_n)$ is a list of variables, $d = (d_1, \dots, d_n)$ is a list of domains for the variables, and $c = (c_1, \dots, c_m)$ is a list of constraints. The focus of this paper is on numerical CSPs (NCSPs), whose variables are continuous, and thus domains are subsets of $\mathbb{R}$, typically represented as a set of intervals (box) $\mathbf{x}$. To conform mathematical notations in use for numerical problems, the same symbols are used for variables and their valuations, i.e., x will often denote a point in $\mathbb{R}^n$. For the same reason, we adopt a functional notation for the evaluation of a constraint $c_i : \mathbb{R}^n \to \{\text{true}, \text{false}\}$ and for the evaluation of the conjunction of the constraints $c : \mathbb{R}^n \to \{\text{true}, \text{false}\}$. Hence a solution of a NCSP is a point $x \in \mathbf{x}$ that satisfies $c(x)$, and its solution set is $\chi = \{x \in \mathbf{x} : c(x)\}$.

A bijective function $s : \mathbb{R}^n \to \mathbb{R}^n$ is a *symmetry* of a (N)CSP if it maps solutions to solutions,[1] i.e., for any $x \in \mathbf{x}$ such that $c(x) = \text{true}$, $s(x) \in \mathbf{x}$ and $c(s(x)) = \text{true}$. We say x and $s(x)$ are *symmetric points* and, in case they are solutions, *symmetric solutions*. The symmetries of a (N)CSP form a group for the composition law. This symmetry group is denoted Σ in the following. Though the identity function is forcibly part of Σ, since it is a trivial symmetry of any CSP, it is not considered in the following as it is irrelevant to break.

In this paper, we consider only symmetries that are permutations of variables. Let $\mathcal{S}_n$ be the set of all permutations of $\{1, \dots, n\}$. The image of i by a permutation σ is denoted by i^σ. Any permutation σ is completely defined by the image of each integer in $\{1, \dots, n\}$, and it is usually described as a vector $[1^\sigma, 2^\sigma, \dots, n^\sigma]$. A symmetry s is a *variable symmetry* iff there is a permutation $\sigma \in \mathcal{S}_n$ such that for any point $x \in \mathbf{x}$, $s(x) = (x_{1^\sigma}, \dots, x_{n^\sigma})$. We identify such symmetries with their associated permutations and denote both by σ in the following. Consequently, the group of variable symmetries of a CSP is isomorphic to a permutation subgroup of $\mathcal{S}_n$, which are both identified and denoted by Σ in the following. The application of a variable symmetry σ to a point x is denoted by x^σ, this notation being extended to sets (discrete or continuous) of points $X \subseteq \mathbf{x}$ by $X^\sigma = \{x^\sigma : x \in X\}$.

Example 1. The 3-cyclic roots problem consists in finding all $(x_1, x_2, x_3) \in \mathbb{R}^3$ satisfying $(x_1 + x_2 + x_3 = 0) \ \wedge \ (x_1x_2 + x_2x_3 + x_3x_1 = 0) \ \wedge \ (x_1x_2x_3 = 1)$. This problem has six variable symmetries (including identity):

$$\Sigma = \{[1,2,3], [1,3,2], [2,1,3], [2,3,1], [3,1,2], [3,2,1]\}, \tag{1}$$

i.e., $\Sigma = \mathcal{S}_3$. Indeed, all the variables are *syntactically* interchangeable within all the constraints by the commutativity laws of product and sum.

[1] Nothing is required for non-solution points, i.e., we consider only *solution symmetries* [19].