Symmetry Breaking for Distributed Multi-Context Systems

Christian Drescher\textsuperscript{1}, Thomas Eiter\textsuperscript{2}, Michael Fink\textsuperscript{2}, Thomas Krennwallner\textsuperscript{2}, and Toby Walsh\textsuperscript{1}

\textsuperscript{1} NICTA and University of New South Wales
Locked Bag 6016, Sydney NSW 1466, Australia
\{christian.drescher, toby.walsh\}@nicta.com.au

\textsuperscript{2} Institut für Informationssysteme, Technische Universität Wien
Favoritenstraße 9-11, A-1040 Vienna, Austria
\{eiter, fink, tkren\}@kr.tuwien.ac.at

Abstract. Heterogeneous nonmonotonic multi-context systems (MCS) permit different logics to be used in different contexts, and link them via bridge rules. We investigate the role of symmetry detection and symmetry breaking in such systems to eliminate symmetric parts of the search space and, thereby, simplify the evaluation process. We propose a distributed algorithm that takes a local stance, i.e., computes independently the partial symmetries of a context and, in order to construct potential symmetries of the whole, combines them with those partial symmetries returned by neighbouring contexts. We prove the correctness of our methods. We instantiate such symmetry detection and symmetry breaking in a multi-context system with contexts that use answer set programs, and demonstrate computational benefit on some recently proposed benchmarks.

1 Introduction

Due to the increasing application of distributed systems, there has been recent interest in formalisms that accommodate several, distributed knowledge bases. Based on work by McCarthy \cite{14} and Giunchiglia \cite{11}, a powerful approach is multi-context systems (MCS; \cite{12}). Intuitively, an MCS consists of several heterogeneous theories (the contexts), which may use different logical languages and different inference systems, that are interlinked with a special type of rules that allow to add knowledge into a context depending on knowledge in other contexts. MCSs have applications in various areas such as argumentation, data integration, and multi-agent systems. In the latter, each context models the beliefs of an agent while the bridge rules model an agent’s perception of the environment. Among various proposals for MCS, the general MCS framework of Brewka and Eiter \cite{5} is of special interest, as it generalises previous approaches in contextual reasoning and allows for heterogeneous and nonmonotonic MCSs. Such a
system can have different, possibly nonmonotonic logics in the different contexts, e.g., answer set programs (ASP; [4]), and bridge rules can use default negation to deal with incomplete information.

Although there has been dramatic improvements [3] in the performance of distributed algorithms for evaluating Brewka and Eiter’s style nonmonotonic MCSs such as DMCS [7], many applications exhibit symmetries. For example, suppose context $C_1$ is an advanced database system which repairs inconsistencies (e.g., from key violations in database tables), and another context $C_2$ is accessing the repaired tables via bridge rules. A large (exponential) number of repairs may exist, each yielding a local model (i.e., belief set) of $C_1$; many of those models are symmetric, thus $C_2$’s bridge rules may fire for many symmetric repairs. This can frustrate an evaluation algorithm as it fruitlessly explores symmetric subspaces. Furthermore, communicating symmetric solutions from one context to another can impede further search. If symmetries can be identified, we can avoid redundant computation by pruning parts of the search space through symmetry breaking. However, symmetry breaking in MCSs has not been explored in any depth.

In order to deal with symmetry in MCSs, we must accomplish two tasks: (1) identifying symmetries and (2) breaking the identified symmetries. We make several fundamental and foundational contributions to the study of symmetry in MCS.

– First, we define the notion of symmetry for MCSs. This is subsequently specialized to local symmetries and partial symmetries that capture symmetry on parts of an MCS. Partial symmetries can be extended to a symmetry of the whole system under suitable conditions which are formalized in a corresponding notion of join.

– Second, we design a distributed algorithm to identify symmetries based on such partial symmetries. The method runs as background processes in the contexts and communicate with each other for exchanging partial symmetries. This algorithm computes symmetries of a general MCS based on the partial symmetries for each individual context. We demonstrate such symmetry detection for ASP contexts using automorphisms of a suitable coloured graph.

– Third, we break symmetries by extending the symmetry breaking methods of Crawford et al. [6] to distributed MCS. We construct symmetry-breaking constraints (SBCs) for a MCS that take into account beliefs imported from other contexts into account. These constraints ensure that an evaluation engine never visits two points in the search space that are symmetric. For contexts other than propositional logic, distributed SBCs have to be expressed appropriately. Again we illustrate this in the case of ASP contexts and develop a logic-program encoding for distributed symmetry breaking constraints.

– Finally, we experimentally evaluate our approach on MCSs with ASP contexts. In problems with large number of symmetries, we demonstrate the effectiveness of only breaking a subset of the symmetries. Results on MCS benchmarks that resemble context dependencies of realistic scenarios [3] show that symmetry breaking yields significant improvements in runtime and compression of the solution space.