Some Positive Results for Boundedness of Multiple Recursive Rules

Ke Wang
Department of Information Systems and Computer Sciences
National University of Singapore
Lower Kent Ridge Road, Singapore 0511

Abstract. Following results are sketched in this extended abstract: (1) Datalog recursive programs where each rule has at most one subgoal called unit recursions are shown to be bounded, with an effective construction of equivalent non-recursive programs. (2) A generalized chain program, which allow IDB predicates of arbitrary arity and remove the uniqueness condition of chain variables, is bounded if and only if it is a unit recursion. (3) The characterization of uniform unboundedness for linear sirups in [NS] is extended to a substantial superclass called class $C^+$. (4) Boundedness for class $C^+$ with multiple exit rules is decidable in polynomial space. (5) Predicate boundedness is decidable in doubly exponential time for a large class of Datalog programs that properly contains all connected monadic programs. (6) For binary linear programs, program boundedness is decidable if each recursive predicate is defined by at most one recursive rule; predicate boundedness is also decidable if each recursive predicate is mutually recursive with one another.

1 Introduction

This abstract presents some positive results of the boundedness problem for logic programs with multiple rules and multiple recursive predicates. The boundedness problem is to answer whether a given recursive program is equivalent to a non-recursive program, i.e., whether the program is bounded. Detecting bounded programs is a powerful optimization technique as a bounded program needs only a fixed number of iterations in evaluation or can simply be replaced by a non-recursive program. Unfortunately, this problem is undecidable in many cases, which include, among others, programs with a single recursive rule [Ab], linear programs with one binary IDB predicate [Va], and programs with two linear recursive rules and one initialization rule [Va]. Because of the inherent difficulty of boundedness problem, the positive results in earlier work [HKMV, Io, Na, NS, Va] have been obtained mainly for programs of a single recursive rule except for monadic programs [CGKV], some strongly restricted chain rules [AP, BKBR, Gu] that correspond naturally to productions of a context-free grammar, as well as typed rules with a single predicate (not only a single IDB predicate) [S]. There is a lack of positive results for more general rules.

The following are the contributions in this paper.
- (Section 3) Datalog programs in which each recursive rule has at most one subgoal are bounded. Such programs are called \textit{unit recursions} in this paper. The result is also extended to a more general case, called \textit{pseudo-unit recursions}, where each recursive rule has at most one recursive subgoal and the variables in all non-recursive subgoals occur in the recursive subgoal. A construction of a non-recursive program that is equivalent to a unit recursion (resp., pseudo-unit recursion) is presented. The constructed non-recursive program may have many rules, but each rule is very simple and the depth of the program is very small, a feature desirable for parallel evaluation.

- (Section 4) Reduction of boundedness to finiteness of CFL for "chain rules" [AP, BKBR, Gu] is extended to more general programs, called \textit{generalized chain programs}. It is shown that a generalized chain program is bounded if and only if it is a unit recursion. In all "chain rules" studied previously in the literature, uniqueness of chain variables has been a crucial requirement for mapping rules to productions of a CFG. Our generalization is substantial in that IDB predicates of arbitrary arity are allowed and uniqueness of chain variables is no longer required.

- (Section 5) We extend the characterization of uniform unboundedness for linear sirups in [NS] to a superclass of the class $C$ defined there, which we call class $C^+$. For a linear sirup in $C^+$, the restriction that no linking variables are mapped to persistent variables, which is a crucial requirement in [NS], is removed. All linear sirups efficiently identified as in $C$ by methods in [NS] as well as more linear sirups can be efficiently identified as in $C^+$ by a method given in this paper.

We also extend the language (or automata) theoretic approach in [CGKV] for monadic programs to arbitrary Datalog programs. In particular,

- (Section 6) We show that boundedness (not just uniform boundedness) for linear sirups with a recursive rule in class $C^+$ and with multiple exit rules is decidable in polynomial space. Positive results of boundedness were obtained in [NS] only for a refinement of class $C$ with one strongly restricted exit rule.

- (Section 7) We show that predicate boundedness is decidable in doubly exponential time for a large class of arbitrary Datalog programs that properly contains all connected monadic programs [CGKV].

- (Section 8) We show that program boundedness is decidable for binary linear programs in which each recursive predicate is defined by at most one recursive rule and that predicate boundedness is also decidable if each recursive predicate is mutually recursive with one another. These results generalize the decidability of boundedness for linear binary sirups in [Va].

In the spirit of [CGKV], it follows immediately from our results that containment problem is decidable for programs considered in Sections 6,7 and 8.

Due to compactness of the presented materials, certain knowledge of work in [CGKV, Gu, Na, NS, Va] is helpful in reading the paper.