Available online at www.sciencedirect.com
SCIENCE DIRECT
Electronic Notes in Theoretical Computer Science
ELSEVIER Electronic Notes in Theoretical Computer Science 133 (2005) 175–191
www.elsevier.com/locate/entcs

Obtaining Memory-Efficient Solutions to Boolean Equation Systems

Misa Keinänen[1,2]

Department of Computer Science and Engineering
Laboratory for Theoretical Computer Science
Helsinki University of Technology
P.O. Box 5400, FIN-02015 HUT, Finland

Abstract

This paper is concerned with memory-efficient solution techniques for Boolean fixed-point equations. We show how certain structures of fixed-point equation systems, often encountered in solving verification problems, can be exploited in order to substantially improve the performance of fixed-point computations. Also, we investigate the space complexity of the problem of solving Boolean equation systems, showing a NL-hardness result. A prototype of the proposed technique has been implemented and experimental results on a series of protocol verification benchmarks are reported.

Keywords: algorithm, boolean equation systems, memory efficient resolution, space complexity.

1 Introduction

Many verification tools and algorithms for finite-state concurrent systems need to solve fixed-point equations, such as those for checking behavioural equivalences and model checking μ-calculus [17]. However, solving general fixed-point equations is a difficult task for which no polynomial time algorithms are known (although the problem is in $NP \cap co{-}NP$ [11], and even in $UP \cap co{-}UP$ [16]).

[1] Petteri Kaski is thanked for valuable discussions. Jaco van de Pol provided useful comments on the specifications from [8,13]. The work was financially supported by Academy of Finland (project 53695), Emil Aaltonen foundation and Helsinki Graduate School in Computer Science and Engineering.

[2] Email: mkk@tcs.hut.fi

doi:10.1016/j.entcs.2004.08.064

Special techniques have been developed to overcome this and there are many practically relevant classes of fixed-point equation systems which do have very efficient algorithms [1,4,6,11,12,14,19]. But a remaining problem is the memory consumption of these techniques. Industrial scale verification problems lead typically to systems with millions of fixed-point equations, and these equations may become too large to fit the memory at the same time. Yet, so far no-one has investigated the space complexity of solving fixpoint equation systems, and very little attention has been paid to develop memory-efficient solution algorithms.

In this paper, we investigate memory-efficient techniques for solving Boolean equation systems [1,2,18,20,21,23,26]. Boolean equation systems give a useful framework to study fixed-point computations, because μ-calculus can be encoded in this setting (for example, see [1,5,21,23]). We will call attention to a fragment of Boolean equation systems, which we call *stratified* systems of equations. These are specific fixed-point equation systems where, the variables are interspersed in the equations so that the computation of the solutions becomes very simple, and can be optimized to be memory efficient. We present efficient techniques for solving such systems. The main novelty is that we exploit the particular structure of fixed-point equations to substantially improve the memory usage.

We also study the space complexity of solving general Boolean equation systems and prove a NL-hardness result which appears to be new. This helps to understand the theoretical minimum of space storage needed for solving Boolean equation systems.

It turns out that many verification problems of practical importance can be expressed with Boolean equation systems which fall in the stratified class. So a set of applications of our techniques is outlined. For the purposes of this paper, we have also implemented the algorithm and used it to tackle a series of protocol verification problems. The technique turned out to be successful on these initial applications.

The structure of the paper is as follows. In Section 2, we introduce Boolean equation systems and explain the key notions. In Section 3, we discuss state-of-the-art techniques for fixed-point computations and discuss their limitations. In Section 4, we describe how stratified Boolean equation systems can be solved with our techniques, and combine these ideas into a single effective procedure. In Section 5, we prove the NL-hardness of the problem of solving fixed-point equations. In Section 6, we illustrate that many verification problems can be encoded with equation systems in the stratified class. In Section 7 we present the application of our solution techniques to protocol verification problems and results on two sets of benchmarks. Section 8 concludes.