Science of Computer Programming ••• (••••) •••–•••

Contents lists available at ScienceDirect

Science of Computer Programming

www.elsevier.com/locate/scico

Variant-based satisfiability in initial algebras

José Meseguer

Department of Computer Science, University of Illinois at Urbana-Champaign, United States

ARTICLE INFO

Article history:
Received 20 August 2016
Received in revised form 1 September 2017
Accepted 4 September 2017
Available online xxxx

A Fuensanta, por "La Tregua," donde surgieron estas ideas

Keywords:
Finite variant property (FVP)
Constructor variant
Constructor unifier
Folding variant narrowing
Satisfiability in initial algebras

ABSTRACT

Although different satisfiability decision procedures can be combined by algorithms such as those of Nelson–Oppen or Shostak, current tools typically can only support a finite number of theories to use in such combinations. To make SMT solving more widely applicable one needs *theory-generic* satisfiability algorithms allowing a potentially infinite number of decidable theories to be *user-definable*, instead of needing to be built in by tool implementers. This work studies how *folding variant narrowing*, a generic unification algorithm that offers good extensibility in unification theory, can be extended to a generic *variant-based satisfiability* algorithm for the initial algebras of user-specified input theories when such theories satisfy Comon and Delaune's *finite variant property* (FVP) and some extra conditions. Several, increasingly larger infinite classes of theories whose initial algebras enjoy decidable variant-based satisfiability are identified and illustrated with examples. A method based on *descent maps* to bring other theories into these classes and to improve the generic algorithm's efficiency is also proposed.

1. Introduction

The use of decision procedures for theories axiomatizing data structures and functions commonly occurring in software and hardware systems is currently one of the most effective methods at the heart of state-of-the-art theorem provers and model checkers. It offers the promise, and often even the reality, of scaling up such verification efforts to handle large systems used in industrial practice. In the area of decision procedures two important phases stand out. The first is the discovery in the late 70's and early 80's of *combination methods* by Nelson and Oppen [80] and Shostak [85] to achieve satisfiability in combinations of decidable theories. The second is the marriage of SAT-solving technology with satisfiability procedures for decidable theories, an approach pioneered independently by a number of different groups [5,47,7,14,79,48] and distilled in the influential DPLL(T) architecture [82]. This approach has been key to the success of SMT solving, as witnessed by a vast literature on the subject.

However, one important challenge is the lack of *extensibility* of current SMT tools. This may seem somewhat paradoxical to say, since obviously the Nelson–Oppen (NO) combination method [80,83] offers unlimited extensibility by theory combinations under some conditions on the combined theories. This is true enough, but:

1. One needs to have algorithms and implementations for each of the theories supported by the SMT solver, which requires a non-trivial effort and in any case limits at any given time each SMT solver to support a *finite* (and in practice not very large) library of theories that it can handle.

E-mail address: meseguer@illinois.edu.

http://dx.doi.org/10.1016/j.scico.2017.09.001
0167-6423/

Please cite this article in press as: J. Meseguer, Variant-based satisfiability in initial algebras, Sci. Comput. Program. (2017), http://dx.doi.org/10.1016/j.scico.2017.09.001

2. What we need are *theory-generic*—i.e., not for a single theory, but for an infinite class of theories—satisfiability decision procedures, so that the input theories are easily *user-definable*. In this way, an SMT solver's repertoire of individual decidable theories becomes potentially *infinite* and easily specifiable by the tool's *users*, as opposed to its implementers.

Achieving extensibility in this more ambitious sense can have large payoffs for SMT solving technology, because it can widely extend both its *scope* and its *effectiveness*. In formal verification practice this would allow automating larger fragments of the verification effort, both in theorem proving and in model checking, and therefore scaling up to effectively handle larger problems.

This paper is all about making SMT solving extensible in the just-mentioned sense by what I call *variant-based satisfiability* methods. The best way for me to explain the key ideas is to place them in the context of a recent sea change in *unification theory* that has been quietly taking place thanks to *variant-based unification* [43,44], inspired by the Comon and Delaune notion of variant [32].

Unification theory is not just a *neighboring area* of SMT solving, but actually a *subfield*. Specifically, the subfield obtained by: (i) considering theories of the form $th(T_{\Sigma/E}(X))$, associated to equational theories (Σ, E), where $th(T_{\Sigma/E}(X))$ denotes the theory of the free (Σ, E)-algebra $T_{\Sigma/E}(X)$ on countably many variables X, and (ii) restricting ourselves to *positive* quantifier-free (QF) formulas of the form[1] $\varphi \equiv \bigvee_i \bigwedge G_i$, with each $\bigwedge G_i$ a conjunction of equations. A finitary E-unification algorithm then gives us a *decision procedure* for satisfiability of such formulas φ not only in the *free* (Σ, E)-algebra $T_{\Sigma/E}(X)$, but also in the *initial* (Σ, E)-algebra $T_{\Sigma/E}$ when all sorts of $T_{\Sigma/E}$ are non-empty.

Unification theory is not only a subfield of SMT solving but what might be called a *microcosm*, where many problems and challenges of SMT solving already show up, including the extensibility problem. For example, the Nelson–Oppen (NO) combination algorithm [80,83] is mirrored by algorithms for combining unification procedures, such as those of Baader and Schulz [8] and Boudet [20] (see [10] for a unified treatment of both NO and the Baader–Schulz algorithms). Also, as for SMT solving, extensibility *is* a problem for the exact same reasons: although combination methods exist, E-unification algorithms require substantial implementation efforts and a tool can only support so many of them.

One important advantage of unification theory is that it has had for a long time *generic* E-unification *semi-algorithms*, namely, *narrowing-based* [89,46,62,63] and *transformation-based* [49,90] ones. But one important drawback of these semi-algorithms is that, since E-unification for arbitrary E is undecidable, in general they only provide a *semi-decision* procedure, which is useless for *deciding* unifiability, i.e., satisfiability of formulas $\varphi \equiv \bigvee_i \bigwedge G_i$ in both the free (Σ, E)-algebra $T_{\Sigma/E}(X)$ and the initial (Σ, E)-algebra $T_{\Sigma/E}$, *unless* they can be proved *terminating* for a given equational theory E. For theories E whose equations can be oriented as convergent rewrite rules R, some termination results for narrowing-based unification, mostly centered on special input theories for the *basic narrowing* strategy [62], do exist for some quite restrictive classes of rules R (see [1,2], and references there, for a comprehensive and up-to-date treatment). Instead, the more general case of termination for narrowing-based unification for equational theories $E \uplus B$ for which the equations E can be oriented as convergent rules R *modulo* axioms B having a finitary B-unification algorithm, has been a real *terra incognita* until very recently, because *negative* results, like the impossibility of using basic narrowing when B is a set of associative-commutative (AC) axioms [32], seemed to dash any hopes not just of termination, but even of efficient implementation. Many of these limitations have now disappeared thanks to the *folding variant narrowing* algorithm [43,44]. Let me summarize the current state of the art after [44]:

1. When B has a finitary unification algorithm, folding variant narrowing with convergent oriented equations E modulo B will terminate on any input term (including unification problems expressed in an extended signature) iff $E \uplus B$ has the *finite variant property*[2] (FVP) in the sharpened sense of Definition 5.
2. No other complete narrowing strategy can terminate more often than folding variant narrowing; in particular, basic narrowing (when applicable, e.g., $B = \emptyset$) terminates strictly less often.
3. FVP is a semi-decidable property, and when it actually holds can be easily checked by folding variant narrowing, assuming convergence of the oriented equations E modulo B and a finitary B-unification algorithm [24].
4. Folding variant narrowing and variant-based unification for theories $E \uplus B$, where B can be any combination of associativity, commutativity and identity axioms are well supported by Maude [26] in its latest 2.7 version [37].

There are by now papers, e.g., [32,42,41], many cryptographic protocol specifications, e.g., [42,96,56,23,84], and several verification tools, e.g., [42,23,84], demonstrating that FVP equational theories are omni-present in cryptographic protocol verification and that variant-based unification and narrowing are very general and effective formal reasoning methods to verify such protocols. In this paper I give many examples showing that, in a similar way, QF satisfiability in initial algebras of FVP theories is decidable under reasonable conditions.

The key question addressed in this paper should now be obvious: can the good properties of variant-based unification as a *theory-generic*, finitary $E \uplus B$-unification algorithm for FVP theories be extended to a, likewise theory-generic, variant-based

[1] Here and anywhere else the symbol $\equiv$ will be used to denote syntactic equality.

[2] u is an E, B-*variant* of a term t in the Comon–Delaune sense [32] (see Definition 5 for a sharper definition) if u is the E, B-canonical form of a substitution instance, $t\theta$, of t. Therefore, the variants of t are intuitively the "irreducible patterns" to which t can be symbolically evaluated by the oriented equations E modulo B. $E \uplus B$ has the *finite variant property* (FVP) if any t has a *finite* set of *most general* E, B-variants.

Please cite this article in press as: J. Meseguer, Variant-based satisfiability in initial algebras, Sci. Comput. Program. (2017), http://dx.doi.org/10.1016/j.scico.2017.09.001