Shostak Light

Harald Ganzinger

MPI Informatik, D-66123 Saarbrücken, Germany, hg@mpi-sb.mpg.de

Abstract. We represent the essential ingredients of Shostak's procedure at a high level of abstraction, and as a refinement of the Nelson-Oppen procedure. We analyze completeness issues of the method based on a general notion of theories. We also formalize a notion of σ-models and show that on the basis of Shostak's procedure we cannot distinguish a theory from its approximation represented by the class of its σ-models.

1 Introduction

Shostak (1984) introduced a procedure that decides the universal fragment of the theory of equality. This congruence closure procedure can be combined with decision procedures for other theories, provided they are what Shostak called "canonizable" and "solvable". Shostak's procedure is at the core of several theorem proving systems, including PVS (Owre, Rushby & Shankar 1992), STeP (Manna et al. 1995) and SVC (Barrett, Dill & Levitt 1996). Previous papers have often suffered from a too technical description of the procedure. Consequently completeness of those formulations of Shostak's procedure has always been difficult to prove. Kapur (2002) compiles a list of technical problems with some of these papers.

More recently several papers have helped in advancing the status of this matter. Tiwari (2000) described Shostak's procedure at an abstract level of inference rules extending the inference system for congruence closure given by Bachmair & Tiwari (2000) (also see Kapur 1997). In Tiwari's presentation Shostak's method appears as a special case of the Nelson/Oppen method (Nelson & Oppen 1979) and is proved complete for equational theories. Rueß & Shankar (2001) presented a more implementation-oriented version of Shostak's procedure eliminating certain sources of incompleteness in Shostak's original formulation. However because of the lack of a more abstract specification, the proofs in the latter paper are somewhat hard to verify. Also, Rueß & Shankar (2001) only treat the validity problem for Horn clauses, and their completeness proof involves a specific notion of σ-models. Barrett, Dill & Stump (2002) describe a procedure without treating free function symbols, concentrating on the relation between convexity, a prerequisite for the completeness of Shostak's method, and stable infiniteness. They observe that convexity implies stable infiniteness for first-order theories without trivial models so that solvers for different theories can be combined with the Nelson-Oppen approach.

The present paper attempts at achieving two goals. One goal is to provide a formal presentation of Shostak's procedure intended to be useful as an abstract layer with respect to which more concrete implementations can be verified and variations of the procedure can be developed. Secondly we want to adopt a semantic view where built-in theories are arbitrary classes of structures, not necessarily first-order, and then investigate completeness issues from that general point of view.

As in (Tiwari 2000) and in (Bjørner 1998) we shall present Shostak's procedure (modeled by an inference system $\mathcal{S}$) as a refinement of an inference system $\mathcal{NO}$ modeling a non-branching variant of the Nelson-Oppen procedure. Our view is similar to the one adopted by Bjørner (1998) in that we relate both Shostak's and the Nelson-Oppen method to the general framework of constraint programming and constraint theorem proving: The solvers assumed for Shostak's procedure transform constraints into solved form which in turn can be used to simplify other constraints by eliminating variables. In the Nelson-Oppen procedure constraints are only tested for satisfiability but never solved. So the main difference is that of satisfiability checking for constraints vs. actually computing their solutions. In refutational theorem proving constraint solving is not needed for completeness, and for theories where complete sets of unifiers are large (or infinite) constraint solving is not advisable anyway (Huet 1972, Nieuwenhuis & Rubio 1995). To keep matters simple, in our presentation we do not model any specific efficient version of congruence closure computation. For these issues the reader is referred to (Bachmair, Tiwari & Vigneron 2002) and (Kapur 1997).

As theories in this paper are not restricted a priori we will be able to derive precise characterizations for completeness. Our completeness proofs are semantic and do not require any reasoning about the combinatorics of congruences and canonical term algebras. We show that convexity of a theory is necessary and sufficient for the completeness of $\mathcal{NO}$ and, hence, of $\mathcal{S}$. That convexity is necessary is immediate when one wants to apply Shostak's procedure to the validity problem of equational clauses with more than one positive literal. What we prove here is that even if, as in (Rueß & Shankar 2001), the procedure is only applied to Horn clauses, in the presence of additional free function symbols convexity is indispensable for completeness.

In Section 5, we shall relate convexity to the concept of σ-models. Shostak's (1984) definition of σ-models is somewhat loose. The notion defined in (Rueß & Shankar 2001) turns out to be too restrictive. For the more liberal definition that we shall provide the class of σ-models of a solvable theory represents a convex theory, and hence either Shostak's procedure is incomplete, or else we cannot distinguish between the theory and its σ-models by deciding clausal validity problems.

In Section 6 we briefly take a closer look at the special case of the Nelson-Oppen procedure for a single built-in theory plus free function symbols. In this case the procedure turns out to be complete for any, not necessarily stably infinite, theory. That is we show that if clausal validity is decidable for a theory it remains decidable upon adding free function symbols. Refining that procedure by employing a solver would give one a version of Shostak's procedure complete also for non-convex theories.

2 Basic Concepts

We employ the usual logical notions and notation. Specifically we consider first-order signatures of function symbols and assume that $\approx$ denotes formal equality, a logical symbol present implicitly in any signature. If Σ is a signature, a *Σ-term [Σ-formula]* is built from function symbols in Σ and from variables. When we write $\forall X F$, we assume that X is some superset of the set of free variables appearing in F. We consider equality $\approx$ as syntactically symmetric so that $u \approx v$ also matches $v \approx u$. Negated equations $\neg(s \approx t)$ are also written as $s \not\approx t$. We shall sometimes use oriented equations as rewrite rules $s \Rightarrow t$. The semantics of a rewrite rule is that of an equation, but rewrite rules are oriented, that is, not considered symmetric syntactically. Sets of equations and disequations are semantically viewed as the conjunction of their elements.

For us a *Σ-theory* $\mathcal{M}$ is simply a class of Σ-structures, the *models* of the theory, not necessarily first-order. We are interested in deciding the validity problem for clauses for such theories. The *word problem* for $\mathcal{M}$ is to decide whether or not $\mathcal{M} \models \forall X(s \approx t)$ for Σ-equations $s \approx t$. If $\mathcal{M} \models \forall X(s \approx t)$ we call s and t *equal modulo* $\mathcal{M}$, and call them *different* modulo $\mathcal{M}$, otherwise. The *uniform word problem*, also called the *validity problem for Horn clauses*, is the problem of deciding implications $\mathcal{M} \models \forall X(\Gamma \to A)$ for finite sets of Σ-equations Γ and for $A = \bot$ or $A = s \approx t$ a Σ-equation. The *clausal validity problem* in $\mathcal{M}$ is the problem of deciding $\mathcal{M} \models \forall X(A_1 \wedge \ldots \wedge A_n \to B_1 \vee \ldots \vee B_m)$ for arbitrary clauses over Σ-equations A_i and B_j.

A theory $\mathcal{M}$ is called *convex* if for any finite set Γ of Σ-equations and for Σ-equations A_i, $1 \leq i \leq n$, whenever $\mathcal{M} \models \forall X(\Gamma \to A_1 \vee \ldots \vee A_n)$, then there exists an index j such that $\mathcal{M} \models \forall X(\Gamma \to A_j)$. For convex theories, any clausal validity problem can be reduced to a linear number of validity problems for Horn clauses. Clausal validity problems are often presented as unsatisfiability problems for sets of equational literals since $\mathcal{M} \models \forall X(\Gamma \to A_1 \vee \ldots \vee A_n)$ if, and only if, $\exists X(\Gamma \wedge \neg A_1 \wedge \ldots \wedge \neg A_n)$ is unsatisfiable in $\mathcal{M}$.

In the simple case, both the Nelson/Oppen and Shostak's method deal with two disjoint signatures Δ and Φ of *defined function symbols* and of *free function symbols*, respectively, where the semantics of the defined symbols are given by a Δ-theory $\mathcal{T}$.[1] The theory models are considered in contexts where additional free functions from Φ exist. To that end, by $\mathcal{T}^{\Phi}$ we denote the class of $\Delta \cup \Phi$-structures I such that the restriction of I to Δ (one simply ignores the interpretations of the function symbols from Φ) is in $\mathcal{T}$. Both the Nelson/Oppen method and Shostak's method are designed to extend given decision procedures for the clausal validity problem in the theory $\mathcal{T}$ to a decision procedure for the validity of clauses in $\mathcal{T}^{\Phi}$.

[1] In the general case of the Nelson-Oppen method we may have more than one theory over disjoint signatures, possibly including a theory of free functions. The original definition of Shostak's procedure in (Shostak 1984) was given for a single built-in theory, and since then several authors including Bjørner (1998), Kapur (2002), Barrett et al. (2002), and Shankar & Rueß (2002) have described variants to be applied to the combination of solvable theories.

Contradiction

$$\frac{E \parallel D}{\bot}$$

if $\mathcal{T} \models \forall X(E \to \bot)$.

Compose

$$\frac{E \parallel D \cup \{f(s_1, \ldots, s_n) \approx s,\ f(s'_1, \ldots, s'_n) \approx s'\}}{E \cup \{s \approx s'\} \parallel D \cup \{f(s_1, \ldots, s_n) \approx s\}}$$

if $\mathcal{T} \models \forall X(E \to s_i \approx s'_i)$, for $1 \le i \le n$.

Fig. 1. Inference system $\mathcal{NO}$ modeling a non-branching Nelson-Oppen procedure

3 A Non-Branching Nelson/Oppen Procedure

Let us assume that we have a theory $\mathcal{T}$ for which the clausal validity problem is decidable and that we want to decide the clausal validity problem in $\mathcal{T}^{\Phi}$. One possibility is to employ a non-branching version of the the Nelson/Oppen method. When a theory clause $A_1 \wedge \ldots \wedge A_n \to B_1 \vee \ldots \vee B_m$ is valid without $A_1 \wedge \ldots \wedge A_n$ entailing one of the disjuncts B_i we are not going to non-deterministically backtrack over the m disjuncts. Also we do not consider the free theory of Φ as another built-in theory, but rather deal with it explicitly using a specific rule for congruence closure. The system $\mathcal{NO}$ given in Figure 1, where rules may be applied in any order, models that particular version of the Nelson/Oppen procedure.

The inference rules manipulate configurations of the form $E \parallel D$ and are intended to decide the satisfiability of $\exists X(E \wedge D)$ in $\mathcal{T}^{\Phi}$, with X the set of variables appearing in E or D. Our format is such that E contains equations and disequations over Δ, the *constraints*. (We use the letters s, t, u, v, and w to denote Δ-terms.) D is a set of *function definitions* $F \approx u$ for free function symbols. Here, F denotes terms of the form $f(s_1, \ldots, s_n)$, with f in Φ and with Δ-terms s_i as arguments. Since $\mathcal{NO}$ only deals with this restricted ("purified") syntactic format of constraints and function definitions, we have to assume that the initially given problem is presented in this form. This is no essential restriction as the satisfiability problems arising from clausal validity problems over $\Delta \cup \Phi$ can be purified with the help of auxiliary variables in linear time.

The rule Contradiction derives $\bot$ if the set of constraints is unsatisfiable in $\mathcal{T}$. Compose computes overlaps between two function definitions, provided their arguments (which are Δ-terms) are equal for every solution of the constraints in E. Note that the formulas $E \to \bot$ and $E \to s_i \approx s'_i$ are equivalent to clauses and hence their validity is decided by the theory module.

We shall write $E \parallel D \vdash_{\mathcal{NO}} E' \parallel D'$ whenever the first configuration can be transformed into the second by application of a rule in $\mathcal{NO}$. An $\mathcal{NO}$-derivation is a sequence of configurations $\kappa_0 \vdash_{\mathcal{NO}} \kappa_1 \vdash_{\mathcal{NO}} \ldots$. A configuration to which no inference rule applies is called *terminal* in $\mathcal{NO}$ or *irreducible* by $\mathcal{NO}$.

Proposition 1. *The inference system is sound. More specifically, (i) whenever* $E \parallel D \vdash E' \parallel D'$ *then* $\mathcal{T}^{\Phi} \models \forall X(E \wedge D \leftrightarrow E' \wedge D')$*; and (ii) if* $E \parallel D \vdash \bot$ *then* $E \cup D$ *is unsatisfiable in* $\mathcal{T}^{\Phi}$

Proposition 2. *The derivation relation* $\vdash_{\mathcal{NO}}$ *is well-founded.*

Theorem 1. $\mathcal{NO}$ *is complete for a theory* $\mathcal{T}$ *if* $\mathcal{T}$ *is convex.*

Proof. We assume that $\mathcal{T}$ is convex and show that whenever the procedure terminates with final state $E \parallel D$ then $E \wedge D$ is satisfiable in $\mathcal{T}^{\Phi}$. For this we need to identify a suitable $\mathcal{T}$-model I satisfying E and extend it by definitions for the free function symbols so that D is also satisfied. Consider the set M of Δ-terms that appear either in a disequation in E, or as an argument of a free function symbol on the left side of a function definition in D. Call two terms s and t in M equivalent if $\mathcal{T} \models \forall X(E \rightarrow s \approx t)$. Define N such that it contains exactly one representative of each equivalence class of M. Suppose $N = \{u_1, \ldots, u_m\}$. If $E \cup \{u_i \not\approx u_j \mid i \neq j\}$ were unsatisfiable in $\mathcal{T}$ then either $m \leq 1$ and E is unsatisfiable, or else $m > 1$ and $\mathcal{T} \models \forall X(E \rightarrow \bigvee_{i \neq j} u_i \approx u_j)$. In the first case Contradiction would have derived $\bot$ which it did not. In the second case, by the convexity of $\mathcal{T}$, again either E is unsatisfiable which it is not, or else $\mathcal{T} \models \forall X(E_+ \rightarrow u_i \approx u_j)$ for some pair $i \neq j$, where E_+ is the subset of positive equations in E. The latter situation would contradict the way N was constructed. We have shown that there exists a structure I in $\mathcal{T}$ and a variable assignment $\alpha : X \rightarrow I$ satisfying E and where the terms u_i denote pairwise different values in I.

Now extend I by interpretations for the free function symbols as follows: If f is a free function symbol and $f(s_1, \ldots, s_n) \approx u$ is a function definition in D, evaluate the s_i as well as u in I, α, yielding values a_i and c, respectively, and define $f_I(a_1, \ldots, a_n)$ to be c. Define f_I arbitrarily at all other argument tuples of the domain of I. We have to show that f is well-defined. A potential ambiguity may arise from the presence of two definitions $f(s_1, \ldots, s_n) \approx s$ and $f(s'_1, \ldots, s'_n) \approx s'$ in D, should it be the case that $I, \alpha \models s_i \approx s'_i$, for $1 \leq i \leq n$. However, if these two function definitions were present then there would exist an index j such that $\mathcal{T} \not\models \forall X(E \rightarrow s_j \approx s'_j)$, for otherwise Compose would have eliminated one of the two definitions. By construction N contains different terms u and u' equivalent to s_j and s'_j, respectively. Since we picked I and α such that different terms in N denote different values in I, it cannot be possible that $I, \alpha \models s_j \approx s'_j$. □

In the general branching version of the Nelson-Oppen procedure with an arbitrary number of theory components a property weaker than convexity, called stable infiniteness, suffices to obtain completeness. For a detailed proof of this fact the reader is referred to (Tinelli & Harandi 1996). For theories having only non-trivial models—these are structures with more than one element—convexity is sufficient for the completeness also of the general version of the Nelson-Oppen procedure. This is a consequence of the results in (Barrett et al. 2002). Above we gave a direct and simple model construction proof that neither relies on the general completeness result nor on the relation between convexity and stable infiniteness.

If a theory has non-trivial models only, convexity is also necessary for $\mathcal{NO}$ to be complete even for merely deciding the validity of Horn clauses.

Theorem 2. *If $\mathcal{T}$ is a non-convex theory of non-trivial structures then there exists a Horn clause of the form $E, D \rightarrow s \approx t$ valid in $\mathcal{T}^{\Phi}$ such that $E \cup \{s \not\approx t\} \parallel D$ is irreducible by $\mathcal{NO}$.*

Proof. Suppose $\mathcal{T}$ is not convex. Then there exists a set of Δ-equations E and a set of $n \geq 2$ Δ-equations $s_i \approx t_i$, $1 \leq i \leq n$, such that the clause $C = \forall X(E \rightarrow \bigvee_i s_i \approx t_i)$ is valid in $\mathcal{T}$ but $\mathcal{T} \not\models \forall X(E \rightarrow s_i \approx t_i)$, for any i. Let Φ contain n different monadic function symbols f_i, and define D to be the set of function definitions containing the equations $f_i(s_i) \approx x$, and $f_i(t_i) \approx y$, with x and y two different variables not occurring in E, s_i, and t_i. We show that $\mathcal{T} \models \forall X(E \wedge D \rightarrow x \approx y)$. Suppose that I is in $\mathcal{T}$ and α a variable assigment such that $I, \alpha \models E \wedge D$. As C is valid in I there exists an index i such that $I, \alpha \models s_i \approx t_i$. Moreover, as the function definitions are satisfied in I, α we infer that $I, \alpha \models f_i(t_i) \approx y$ and $I, \alpha \models f_i(s_i) \approx x$ and, hence, $I, \alpha \models x \approx y$. On the other hand no inference in $\mathcal{NO}$ applies to the configuration $E \cup \{x \not\approx y\} \parallel D$. Compose cannot be applied as $\mathcal{T} \not\models \forall X(E \rightarrow s_i \approx t_i)$, for every i. For the same reason E must be satisfiable in $\mathcal{T}$. As $\mathcal{T}$ has only non-trivial models, Contradiction does not apply to $E \cup \{x \not\approx y\}$. $\square$

The system $\mathcal{NO}$ only formalizes the bare bones of a variant of the Nelson/Oppen procedure. In practice one may want to add additional (sound) inference rules to increase the efficiency of the method. There is a uniform method of doing this in a way such that completeness is maintained. Call a set of additional inference rules on configurations *admissible* if they are sound and if termination is maintained. Completeness can not be lost by adding additional inference rules. However one can also safely delete instances of inference rules as long as it is guaranteed that configurations reducible by deleted inference rules can also be reduced by some other inference rules. In the next section we are going to model Shostak's method as a refinement in this sense of the inference system $\mathcal{NO}$.

4 Shostak Light

Shostak's procedure assumes the presence of a (unitary) unification algorithm for $\mathcal{T}$. More specifically it is assumed that there exists an effectively computable function solve such that, for any $\mathcal{T}$-equation $s \approx t$:

(A) $\mathsf{solve}(s \approx t) = \bot$ if, and only if, $\mathcal{T} \models \forall X(s \not\approx t)$;
(B) $\mathsf{solve}(s \approx t) = \emptyset$ if, and only if, $\mathcal{T} \models \forall X(s \approx t)$; and otherwise
(C) $\mathsf{solve}(s \approx t) = \{x_1 \Rightarrow u_1, \ldots, x_n \Rightarrow u_n\}$ is a finite set of rewrite rules over Δ such that

(i) the x_i are pairwise different variables occurring in $s \approx t$;
(ii) the x_i do not occur in the u_j; and
(iii) $\mathcal{T} \models \forall X[(s \approx t) \leftrightarrow \exists Y(x_1 \approx u_1 \wedge \ldots \wedge x_n \approx u_n)]$, where Y is the set of variables occurring in one of the u_j but not in $s \approx t$, and $X \cap Y = \emptyset$.

If a function solve with these properties exists we call the theory *solvable*. $\mathsf{solve}(s \approx t)$, if different from $\perp$, may be viewed as a (possibly empty) substitution $\sigma = [u_1/x_1, \ldots, u_n/x_n]$, written as a set of rewrite rules $\{x_1 \Rightarrow u_1, \ldots, x_n \Rightarrow u_n\}$, that solves the $\mathcal{T}$-equation $s \approx t$.

Solutions can be parameterized by new variables, those in Y. It is assumed that in each calling context for solve, the variables in Y are fresh. Where this needs to be formalized we shall write $\mathsf{solve}_Z(s \approx t) = S$, assuming that then the extra variables appearing in S are not in Z.

Example 1. Let $\mathbb{Q}$ be the single-model theory consisting of the rational numbers with linear arithmetic. In the signature of $\mathbb{Q}$ we have all rational numbers as constants, the binary addition operator $+$, and, for each rational number q, a unary operator $q \cdot \_$ multiplying its argument by q. Equations over $\mathbb{Q}$ can be solved by Gaussian elimination, and it is well-known that the theory is convex.

Example 2. Let $\mathbb{Z}/(3)$ be the one-model theory of the three-element field obtained by considering the remainders from division by 3. Let the signature consist of the constants 0 and 1, and the binary addition $+$. Clearly, $\mathbb{Z}/(3)$ is solvable. For example, $a + a + 1 = b + b$ is solved by $a \Rightarrow 1 + b$. However $\mathbb{Z}/(3)$ is not convex as witnessed by the disjunction $\forall x(x \approx 0 \vee x \approx 1 \vee x \approx 1 + 1)$.

Due to (B), solvers effectively decide the word problem for $\mathcal{T}$.[2] Further, if a theory is solvable we can also effectively decide the *uniform* word problem for $\mathcal{T}$. In fact, for deciding $\mathcal{T} \models \forall X(\Gamma \rightarrow A)$ we iteratively apply solve to the equations in Γ. If this yields $\perp$, the implication is valid in $\mathcal{T}$. Otherwise for the implication to be valid A has to be an equation $s \approx t$, and we obtain a substitution σ that is equivalent to Γ such that $\mathcal{T} \models \forall X(\Gamma \rightarrow s \approx t)$ if, and only if, $\mathcal{T} \models \forall X, Y(s \approx t)\sigma)$ (with Y the extra variables in the codomain of σ), if and only if, $\mathsf{solve}((s \approx t)\sigma) = \emptyset$.

The Nelson/Oppen method is based on being able to decide the [un-]solvability of certain theory constraints. When one has a solver available one can do more and additionally replace constraints by their solved forms (the unifiers). Since

[2] Most presentations of Shostak's method do not require property (B) for solve, but assume the presence of a canonizer so that the word problem can be decided by comparing canonical forms. We present Shostak's procedure without a canonizer. The word problem is all we need to be able to decide, and we may leave it to the implementation of the solver as to whether solutions computed will always be in canonical form. For increasing the efficiency of an actual implementation, the presence of a canonizer might be helpful, but keeping terms always canonical may not be the most efficient strategy. Formalizing normalization strategies involving a canonizer only requires to add more reduction inference rules to our inference system below. We shall discuss this in more detail at the end of this section.

Contradiction

$$\frac{U \cup \{s \approx t\}, R \parallel D}{\perp} \quad \text{if } \mathsf{solve}(s \approx t) = \perp$$

$$\frac{U \cup \{s \not\approx t\}, R \parallel D}{\perp} \quad \text{if } \mathsf{solve}(s \approx t) = \emptyset$$

Solve

$$\frac{U \cup \{s \approx t\}, R \parallel D}{U, R \cup S \parallel D}$$

where
(i) $S = \mathsf{solve}_X(s \approx t) \neq \perp$, with X the set of variables appearing in the antecedent,
(ii) both s and t are irreducible by R.

Reduce

$$\frac{U, R \cup \{x \Rightarrow t\} \parallel D \cup \{F[x] \approx s\}}{U, R \cup \{x \Rightarrow t\} \parallel D \cup \{F[t] \approx s\}}$$

$$\frac{U \cup \{L[x]\}, R \cup \{x \Rightarrow t\} \parallel D}{U \cup \{L[t]\}, R \cup \{x \Rightarrow t\} \parallel D}$$

Compose

$$\frac{U, R \parallel D \cup \{f(s_1, \ldots, s_n) \approx s,\ f(s'_1, \ldots, s'_n) \approx s'\}}{U \cup \{s \approx s'\}, R \parallel D \cup \{f(s_1, \ldots, s_n) \approx s\}}$$

if $\mathsf{solve}(s_i \approx s'_i) = \emptyset$, for $1 \leq i \leq n$.

Fig. 2. Inference system $\mathcal{S}$ modeling Shostak's procedure

solvable theories are required to have unique most general solutions for solvable constraints, no backtracking occurs. Also, applying solutions to unsolved constraints and to function definitions effectively eliminates some of the variables and in this sense simplifies the satisfiability problem. So Shostak is to Nelson/Oppen what theorem proving and CLP with computation of unifiers for built-in theories is to constraint theorem proving and CLP with constraint propagation and constraint satisfiability checking.

The figure 2 presents the inference system $\mathcal{S}$ where again rules may be applied in any order. $\mathcal{S}$ refines $\mathcal{NO}$ in that the constraints E in $\mathcal{NO}$ are now represented by the union of two constraints U and R. In other words, $\mathcal{S}$-configurations $U, R \parallel D$ correspond to $\mathcal{NO}$-configurations $U \cup R \parallel D$. In the refined format, U is the subset of disequations and of "unsolved" positive equations, whereas R is a positive constraint in *solved form*, a substitution derived from previous constraint solving steps. The Contradiction and Compose rules are instances of the Contradiction and Compose rules, respectively, of $\mathcal{NO}$. Configurations reducible by instances of Contradiction and Compose in $\mathcal{NO}$ that are not dealt with by Contradiction and Compose in $\mathcal{S}$ can be reduced by instances of Solve or Reduce (cf. Proposition 8).

Solve solves Δ-equations $s \approx t$. Soundness of this rule is a consequence of the soundness of the solver, cf. Proposition 4 below. More precisely, we only solve normalized equations in which both s and t are irreducible by R. The reduce inferences are designed to compute those normal forms. The solved equation is deleted from U and its solution S is added to the solved form R. The rules added to R upon Solve are all of the form $x \Rightarrow w$, and are called *variable definitions*. By Propositions 3 and 5 R always contains at most one definition for a variable and is terminating. Sets of constraints R with these properties we call *solved forms*.

Reduce expands variables in the F-terms in D as well as in the Δ-terms in U by their definitions. In most presentations of Shostak's method one would apply both the Contradiction and the Compose rules only to irreducible terms s, t, s_i and s_i', respectively. Since our results will be applicable to all fair (that is, maximal) strategies of inference rule application, soundness and completeness also follows for any more refined strategy of substitution application.

We shall write $U, R \parallel D \vdash_{\mathcal{S}} U', R' \parallel D'$ whenever the first configuration can be transformed into the second by application of a rule in $\mathcal{S}$. An $\mathcal{S}$-derivation is a sequence of configurations $\kappa_0 \vdash_{\mathcal{S}} \kappa_1 \vdash_{\mathcal{S}} \ldots$ with κ_0 a configuration of the form $U, \emptyset \parallel D$. A configuration to which no inference rule applies is called *terminal* in $\mathcal{S}$ or *irreducible* by $\mathcal{S}$. A derivation is called *maximal* if its end configuration is terminal.

Proposition 3. *Any rule set R appearing in an $\mathcal{S}$-derivation contains at most one definition for any variable.*

Proof. The property is trivially true initially. When adding a rule set S to R in Solve, if R contains a definition $x' \Rightarrow t'$, S cannot contain a rule for x'. Otherwise x' would have to occur in $s \approx t$, and the equation being solved at this step would not be irreducible with respect to R. □

Proposition 4. *The inference system is sound. More specifically, (i) whenever $U, R \parallel D \vdash_{\mathcal{S}} U', R' \parallel D'$ then $\mathcal{T}^{\Phi} \models \exists X(U \wedge R \wedge D) \rightarrow \exists X, Y(U' \wedge R' \wedge D')$ and $\mathcal{T}^{\Phi} \models \forall X, Y(U' \wedge R' \wedge D' \rightarrow U \wedge R \wedge D)$, with Y the variables in $U', R' \parallel D'$ but not in $U, R \parallel D$; and (ii) if $U, R \parallel D \vdash_{\mathcal{S}} \bot$ then $U \cup R \cup D$ is unsatisfiable in $\mathcal{T}^{\Phi}$.*

In the rewrite systems R, variables are considered as constants which can not be substituted by other terms. In this sense the systems R induce terminating rewrite relations.

Proposition 5. *If $U, R \parallel D \vdash_{\mathcal{S}} U', R' \parallel D'$ and if R is terminating then R' is terminating.*

Proof. Let us, for a configuration $U, R \parallel D$ with variables in X, define $x \succ^X y$ if, and only if, y occurs on the right side of a definition for x in R. R is terminating if, and only if, $\succ^X$ is a well-founded partial ordering on X. (For the "if" part, use a lexicographic path ordering over some precedence $>^X$ for which $\Phi >^X X >^X \Delta$, and which coincides with $\succ^X$ on X to show termination of R.)

We now show that if $\succ^X$ is a well-founded partial ordering on X and if $U, R \parallel D \vdash_{\mathcal{S}} U', R' \parallel D'$ then $\succ^{X'}$ is a well-founded partial ordering on X', the set of variables in the new configuration. The only non-trivial case is when the derivation is by Solve where the new variable definitions S are added to R. However only equations $s \approx t$ irreducible by R are solved, so that no variable appearing in s or t is reducible by R. Therefore any variable occurring on the right side of a rule in S is irreducible by R. Also, according to the definition of a solver, right sides of rules in S are irreducible by S. Consequently, $\succ^{X'}$ is well-founded. □

Proposition 6. *The inference system $\mathcal{S}$ is terminating.*

Proof. We need to describe a well-founded ordering $\succ$ on configurations with terminating rewrite systems R for which all inference rules are strictly monotone. Define $\succ$ such that $\perp$ is minimal. Moreover if $\kappa = U, R \parallel D$ and $\kappa' = U', R' \parallel D'$ are two configurations with X and X', respectively, the set of variables occurring in κ and κ', let $\kappa \succ \kappa'$ whenever
(i) $|D| > |D'|$; or else
(ii) $|D| = |D'|$, and $U \supset U'$; or else
(iii) $|D| = |D'|$, $R = R'$, and $U \Rightarrow_R U'$; or else
(iv) $|D| = |D'|$, $R = R'$, $U = U'$, and $D \Rightarrow_R D'$.
This ordering is well-founded. For if in a sequence $\kappa_0 \succ \kappa_1 \succ \ldots$ the number of function definitions does not decrease and no equations are deleted from U no new rules can be introduced, and therefore $R_i = R_{i+1}$. As the rewrite relations in configurations are all terminating any such sequence must be terminating. Clearly, the rules in $\mathcal{S}$ are strictly decreasing with respect to $\succ$. □

The proposition in particular shows that the number of new variables introduced during a derivation must be finite, irrespective of the way a solver introduces them.

Proposition 7. *(i) If R is a solved form and if s and t are irreducible by R then $\mathcal{T} \models \forall X(R \rightarrow s \approx t)$ if, and only if, $\mathcal{T} \models \forall X(s \approx t)$.*

(ii) Let $\mathcal{T}$ be convex. If R is a solved form, U a set of Δ-disequations satisfiable in $\mathcal{T}$ and irreducible by R, and if s and t are irreducible by R then $\mathcal{T} \models \forall X(U, R \rightarrow s \approx t)$ if, and only if, $\mathcal{T} \models \forall X(s \approx t)$.

Proposition 8. *Let $\mathcal{T}$ be a convex theory. If $U, R \parallel D$ is a terminal configuration of $\mathcal{S}$ then $U \cup R \parallel D$ is a terminal configuration of $\mathcal{NO}$.*

Proof. If no inference in $\mathcal{S}$ can be applied to $U, R \parallel D$ then U contains only negative equations and is satisfiable in $\mathcal{T}$, R is a solved form (cf. propositions 3 and 5), and any term appearing in U or in an F-term in D is irreducible by R. We first show that Compose in $\mathcal{NO}$ cannot be applied to $U \cup R \parallel D$. Otherwise, there would be two definitions $f(s_1, \ldots, s_n) \approx s$ and $f(s'_1, \ldots, s'_n) \approx s'$ in D such that $\mathcal{T} \models \forall X(U, R \rightarrow s_i \approx s'_i)$, for $1 \leq i \leq n$. From (ii) in Proposition 7 we conclude that $\mathcal{T} \models \forall X(s_i \approx s'_i)$, for $1 \leq i \leq n$. Therefore, $\mathsf{solve}(s_i \approx s'_i) = \emptyset$ and Compose would also be applicable in $\mathcal{S}$, which is a contradiction.

Showing that also the Contradiction inference in $\mathcal{NO}$ is not applicable to $U \cup R \parallel D$ is essentially similar. □

To summarize, we have shown that for convex theories $\mathcal{S}$ is a refinement of $\mathcal{NO}$:
(i) All new instances of inference rules are sound. (Proposition 4)
(ii) There is a well-founded refinement of the ordering on $\mathcal{NO}$-configurations such that the new inference rules are strictly monotone (Proposition 6).
(iii) If $U, R \parallel D$ is a terminal configuration for $\mathcal{S}$ then $U \cup R \parallel D$ is a terminal configuration for $\mathcal{NO}$ so that configurations reduced by $\mathcal{NO}$-rules not present anymore in $\mathcal{S}$ can be reduced by other rules in $\mathcal{S}$.
Convexity of $\mathcal{T}$ was required for showing (iii). As a consequence we obtain completeness of $\mathcal{S}$ for convex, solvable theories.

Theorem 3. *$\mathcal{S}$ is complete for any solvable convex theory $\mathcal{T}$.*

For theories without trivial models, convexity is also a necessary requirement for the completeness of Shostak's procedure. The proof can be given essentially as for Theorem 2. Another possibility is to exploit one more correspondence between derivations in $\mathcal{NO}$ and $\mathcal{S}$.

Lemma 1. *If $E \parallel D$ is a terminal configuration for $\mathcal{NO}$ then all maximal derivations in $\mathcal{S}$ from configurations $U, R \parallel D$, where $E = U \cup R$, end in a configuration different from $\perp$.*

Proof. If $E \parallel D$ is irreducible by $\mathcal{NO}$, Contradiction is not applicable to $U, R \parallel D$. Also Compose is not applicable in $U, R \parallel D$ as otherwise Compose in $\mathcal{NO}$ would be applicable to $E \parallel D$. Therefore only Solve and Reduce can be applied to $U, R \parallel D$. Observe that $\mathcal{T} \models \forall X(U \wedge R \leftrightarrow \exists Y(U' \wedge R'))$ with Y the new variables in the configuration obtained from any such inference. Moreover, if $F \approx s$ is a function definition for a free symbol f in D' then there exists a corresponding definition $G \approx t$ in D for f such that $\mathcal{T} \models \forall X(R \rightarrow s \approx t)$, and $\mathcal{T} \models \forall X(R \rightarrow u \approx v)$ for any two terms u and v appearing at corresponding argument positions in F and G, respectively. Therefore Contradiction and Compose can also not be applied in $U', R' \parallel D'$ as otherwise the respective rule in $\mathcal{NO}$ would be applicable to $E \parallel D$. The Lemma now follows by induction. □

Theorem 4. *If $\mathcal{T}$ is a solvable non-convex theory of non-trivial structures then there exists a Horn clause valid in $\mathcal{T}^{\Phi}$ such that $\mathcal{S}$ fails to derive $\perp$ on the corresponding unsatisfiability problem.*

Proof. We apply Theorem 2 to obtain a Horn clause $E, D \rightarrow s \approx t$ that is valid in $\mathcal{T}^{\Phi}$ and for which $E \cup \{s \not\approx t\} \parallel D$ is irreducible by $\mathcal{NO}$. Now apply the previous Lemma. □

So far we have not modeled the concept of canonizers. We briefly sketch how to accommodate canonizers in $\mathcal{S}$. A *canonizer* for a theory $\mathcal{T}$ is a ground[3] rewrite system C on $T_{\Delta}(X)$ where the right side of every rule is irreducible by C and does not contain any variable that does not already appear on the left side. Moreover, each rule in C must be universally valid in $\mathcal{T}$. (Usually canonizers are

[3] The rules may contain variables from X but they are considered as constants.

assumed to have further properties of which we, however, do not make any use here.) Since right sides of rules are reduced, canonizers are terminating rewrite systems. However, $\mathsf{C} \cup R$, where R is a solved form appearing in an $\mathcal{S}$-deduction, in general will not be terminating. Therefore, when extending the Reduce rules to a canonizer one needs to decide upon a terminating strategy for interleaving C-steps and R-steps. One example of a terminating reduction relation would be $\Rightarrow_R^{||} \cup \Rightarrow_{\mathsf{C}}$, if $\Rightarrow_R^{||}$ denotes one step of parallel replacement of *all* R-redexes in a term. Then the termination proof (Proposition 6) remains the same with $\Rightarrow_R$ replaced by $\Rightarrow_R^{||} \cup \Rightarrow_{\mathsf{C}}$.

5 σ-Models

Shostak's original paper as well as (Rueß & Shankar 2001) employ a notion of σ-models relative to which they state completeness. Shostak's definition is somewhat imprecise. According to (Rueß & Shankar 2001), a σ-model is a Δ-structure satisfying all equations $\forall X(s \approx t)$ for which s and t are equal modulo $\mathcal{T}$, and all disequations $s \not\approx t$ such that s and t are *ground* terms that are different modulo $\mathcal{T}$. They call a theory solvable if the class of these σ-models is solvable. This definition of σ-models and solvable theories appears to be too restrictive as it does not capture many intuitively solvable theories. As an example, consider the theory of lists over car, cdr and $cons$ satisfying the rules $\forall x, y(car(cons(x, y)) \Rightarrow x)$, $\forall x, y(cdr(cons(x, y)) \Rightarrow y)$, and $\forall x(cons(car(x), cdr(x)) \Rightarrow x)$ and also the disequations $\forall X(x \not\approx t)$, whenever t is irreducible by the list rules and contains an occurrence of x as an argument of an occurrence of $cons$ in t. Shostak (1984) shows that these lists form a solvable theory. The σ-models of lists, however, contain (non-trivial) structures L in which $l = cdr(l)$ for some element l in L. Therefore, for any solver, $\mathsf{solve}(x \approx cdr(x)) \neq \bot$, and as a consequence of this fact no solver can exist for the theory of σ-models of lists. In fact, $\mathsf{solve}(x \approx cdr(x))$ would have to be a rule $x \Rightarrow t$, with x not in t, and thus $\forall Y(t \approx cdr(t))$ would have to be a consequence of the list rules which it is not. (To see this assume, wolog, that t is irreducible by the list rules. If t does not start with a $cons$, also $cdr(t)$ is irreducible. Otherwise $t = cons(t_1, t_2)$ and t_2 is the canonical form of $cdr(t)$ and different from t.)

The definition of σ-models in (Rueß & Shankar 2001) is solely based on the properties of the canonizer σ (hence the name). Our subsequent definition will be based on the solver (so we should rather speak of solve-models), and as a consequence of this we can avoid the shortcomings illustrated by the list example. Given $\mathcal{T}$, we define $\sigma(\mathcal{T})$, the class of *σ-models of $\mathcal{T}$* (with respect to solve), to be the class of Δ-structures for which solve is sound. This is the class of structures satisfying (i) $\sigma(\mathcal{T}) \models \forall X(s \not\approx t)$, whenever $\mathsf{solve}(s \approx t) = \bot$, and (ii) $\sigma(\mathcal{T}) \models \forall X[(s \approx t) \leftrightarrow \exists Y(x_1 \approx u_1 \wedge \ldots \wedge x_n \approx u_n)]$, whenever $\mathsf{solve}(s \approx t) = \{x_1 \Rightarrow u_1, \ldots, x_n \Rightarrow u_n\}$, where X is the set of parameters in $s \approx t$ and Y is the set of new parameters in the solution. Hence if solve is a solver for $\mathcal{T}$ then it is also a solver for $\sigma(\mathcal{T})$. It is easy to see that $\mathcal{T} \subseteq \sigma(\mathcal{T})$. Therefore, any solver for $\sigma(\mathcal{T})$ is also a solver for $\mathcal{T}$. Also, $\sigma(\sigma(\mathcal{T})) = \sigma(\mathcal{T})$.

Proposition 9. *$\mathcal{S}$ is sound with respect to $\sigma(\mathcal{T})$.*

As equational theories, the σ-theories of (Rueß & Shankar 2001) are closed under products and, therefore, are convex. Our σ-theories are also convex.

Proposition 10. *If $\mathcal{T}$ is a solvable theory then $\sigma(\mathcal{T})$ is convex.*

Proof. σ-models are axiomatized by the first-order conditions (i) and (ii) above. It is not difficult to see that theories axiomatized by formulas of this kind are closed under products, and hence are convex. □

This proves the completeness of the method with respect to σ-models and, thus, extends the results in (Rueß & Shankar 2001) on a more abstract level to a more liberal notion of σ-models. In general $\mathcal{T}$ is a proper subset of $\sigma(\mathcal{T})$. Examples are $\mathcal{T} = \{\mathbb{Q}\}$ or $\mathcal{T} = \{\mathbb{Z}/(3)\}$ as defined above. Yet, if $\mathcal{T}$ is convex then we cannot distinguish $\mathcal{T}$ from $\sigma(\mathcal{T})$ with respect to clausal tautology problems.

Theorem 5. *If $\mathcal{T}$ is convex and solvable and if E is a finite set of equations and disequations over $\Delta \cup \Phi$, then E is satisfiable in $\mathcal{T}^{\Phi}$ if, and only if, E is satisfiable in $\sigma(\mathcal{T})^{\Phi}$.*

Proof. We have shown that $\mathcal{S}$ is sound with respect to both $\mathcal{T}$ and $\sigma(\mathcal{T})$. Since both $\mathcal{T}$ and $\sigma(\mathcal{T})$ are convex, $\mathcal{S}$ is also complete with respect to both $\mathcal{T}$ and $\sigma(\mathcal{T})$. The result of running $\mathcal{S}$ on E will, therefore, establish [un-]satisfiability of E both with respect to $\mathcal{T}$ and $\sigma(\mathcal{T})$. □

This result can be viewed as a justification for the semantic concept of σ-models as defined here. Deciding satisfiability with respect to its σ-models is all one can get for a solvable theory.

6 Branching Nelson-Oppen

From the model construction in the proof of Theorem 1 one sees what is lacking for making $\mathcal{NO}$ complete also for non-convex theories. One needs to non-deterministically branch on all possible ways two function definitions rules might be inconsistent with the constraints. Hence, a branching version of the procedure can be defined by adding this inference rule to $\mathcal{NO}$

$$\frac{E \parallel D}{E \cup \{s \approx t\} \parallel D[s/t] \quad | \quad E \cup \{s \not\approx t\} \parallel D}$$

whenever there are two definitions $f(s_1, \ldots, s_n) \approx s$ and $f(s'_1, \ldots, s'_n) \approx s'$ in D such that $s = s_i \neq s'_i = t$, for some index i, and for no index j the disequation $s_j \not\approx s'_j$ is in E. (By $D[s/t]$ we denote the result of substituting all occurrences of t as an argument of a free function symbol in D by s.)

In the extended system derivations are trees of inference rule applications with the new rule introducing a branching into two sub-derivations. Clearly, the system remains terminating. Terminal configurations in a derivation are either

$\perp$ or are such that if $f(s_1, \ldots, s_n) \approx s$ and $f(s'_1, \ldots, s'_n) \approx s'$ are two different function definitions in D for the same f then there exists an index i such that $s_i \not\approx s'_i$ is in E. Configurations of the first kind are unsatisfiable whereas those of the second kind are satisfiable. In fact since the configuration is terminal, E is satisfiable in $\mathcal{T}$, and extending any model I of E by definitions for the free f satisfying D is no problem as the argument tuples for any two definition rules for any f are different in I. As a consequence we obtain soundness and completeness of the branching version of the Nelson-Oppen procedure. We assume that a run producing a derivation returns "valid" if all leaves in the tree are $\perp$, and "not valid", otherwise.

Theorem 6. *Branching $\mathcal{NO}$ is sound and complete for the clausal validity problem in $\mathcal{T}^{\Phi}$, for any theory $\mathcal{T}$ where clausal validity is decidable.*

This result is not in contradiction with previous results in the literature. For one we only have a single theory built-in and the free function symbols are explicitly dealt with by the Compose rule. Only when combining more than one built-in theory stable infiniteness of the theories is needed. Secondly, the example 2.2 in (Baader & Tinelli 1997) which appears like a counterexample to our theorem allows negative equations to also contain free function symbols. Our preprocessing of satisfiability problems purifies such disequations by introducing new variables the disequality of which might lead to a contradiction in $\mathcal{T}$ that the procedure in (Baader & Tinelli 1997) might fail to infer.

7 Conclusion

We have modeled a version of Shostak's procedure at a high-level of abstraction and as a refinement of a similarly high-level model of a Nelson-Oppen-like procedure. On the semantic side theories were arbitrary sets of structures. Among others we have show that completeness for Horn clause validity problems is equivalent with the convexity of the theory. We have given a definition of σ-models based on the properties of solvers and have shown that these σ-models represent a tight approximation of solvable theories. Hence one may argue that the concept of a solver is more fundamental in Shostak's procedure than the concept of a canonizer (that we have not formalized here). Our completeness result for branching $\mathcal{NO}$ indicates how to obtain a Shostak procedure for non-convex solvable theories. We expect that we can extend our modeling and proof techniques also to the interesting and natural combination procedure for Shostak theories presented in (Shankar & Rueß 2002).

Acknowledgments. I am grateful to Viorica Sofronie-Stokkermans, Uwe Waldmann, Natarajan Shankar, Harald Rueß, and Franz Baader for fruitful discussions on the subject of this paper. I also thank the referees for their detailed and constructive criticism on a much different initial version of this paper.

References

Baader, F. & Tinelli, C. (1997), A new approach for combining decision procedures for the word problem, and its connection to the nelson-oppen combination method, *in* W. McCune, ed., 'Automated Deduction – CADE-14, 14th International Conference on Automated Deduction', LNAI 1249, Springer-Verlag, Townsville, North Queensland, Australia, pp. 19–33.

Bachmair, L. & Tiwari, A. (2000), Abstract congruence closure and specializations, *in* D. McAllester, ed., 'Automated Deduction – CADE-17, 17th International Conference on Automated Deduction', LNAI 1831, Springer-Verlag, Pittsburgh, PA, USA, pp. 64–78.

Bachmair, L., Tiwari, A. & Vigneron, L. (2002), 'Abstract congruence closure', *J. Automated Reasoning* . To appear.

Barrett, C., Dill, D. & Levitt, J. (1996), Validity checking for combinations of theories with equality, *in* M. Srivas & A. Camilleri, eds, 'Formal Methods In Computer-Aided Design, Palo Alto/CA, USA', Vol. 1166, Springer-Verlag, pp. 187–201. citeseer.nj.nec.com/barrett96validity.html

Barrett, C., Dill, D. & Stump, A. (2002), A generalization of Shostak's method for combining decision procedures, *in* 'Proc. FroCos 2002', Springer-Verlag. to appear.

Bjørner, N. (1998), Integrating decision procedures for temporal verification, PhD thesis, Stanford University.

Huet, G. (1972), Constrained Resolution: A Complete Method for Higher Order Logic, PhD thesis, Case Western Reserve University.

Kapur, D. (1997), Shostak's congruence closure as completion, *in* H. Comon, ed., 'Rewriting Techniques and Applications', Lecture Notes in Computer Science, Springer, Sitges, Spain, pp. 23–37.

Kapur, D. (2002), A rewrite rule based framework for combining decision procedures, *in* 'Proc. FroCos 2002', Springer-Verlag. to appear.

Manna, Z., Anuchitanulu, A., Bjørner, N., Browne, A., Chang, E. S., Colón, M., de Alfaro, L., Devarajan, H., Kapur, A., Lee, J., Sipma, H. & Uribe, T. E. (1995), STeP: The Stanford Temporal Prover, *in* 'TAPSOFT', Vol. 915 of *Lecture Notes in Computer Science*, Springer-Verlag, pp. 793–794.

Nelson, G. & Oppen, D. C. (1979), 'Simplification by cooperating decision procedures', *ACM Transactions on Programming Languages and Systems* **2**(2), 245–257.

Nieuwenhuis, R. & Rubio, A. (1995), 'Theorem proving with ordering and equality constrained clauses', *J. Symbolic Computation* **19**(4), 321–352.

Owre, S., Rushby, J. M. & Shankar, N. (1992), PVS: A prototype verification system, *in* D. Kapur, ed., '11th International Conference on Automated Deduction', LNAI 607, Springer-Verlag, Saratoga Springs, New York, USA, pp. 748–752.

Rueß, H. & Shankar, N. (2001), Deconstructing Shostak, *in* 'Proceedings of the Sixteenth IEEE Symposium On Logic In Computer Science (LICS'01)', IEEE Computer Society Press, pp. 19–28.

Shankar, N. & Rueß, H. (2002), Combining Shostak theories, *in* 'Proc. RTA 2002', Lecture Notes in Computer Science, Springer-Verlag. to appear.

Shostak, R. E. (1984), 'Deciding combinations of theories', *J. Association for Computing Machinery* **31**(1), 1–12.

Tinelli, C. & Harandi, M. (1996), A new correctness proof of the Nelson-Oppen combination procedure, *in* '1st Int'l Workshop on Frontiers of Combining Systems (FroCoS'96)', Vol. 3 of *Applied Logic Series*, Kluwer Academic Publishers.

Tiwari, A. (2000), Decision procedures in automated deduction, PhD thesis, SUNY at Stony Brook.